* Equisatisfiable formulas

* F is satisfiable if and only if G is satisfiable, then F & G are called equisatisfiable

$F = a \vee b$

$G = (a \vee \neg n) \wedge (b \vee n)$

$\sigma_1 \Vdash F$

$\sigma_1 : \langle a=1, b=0 \rangle$

$\sigma_2 \nVdash G$, $\sigma_3 \Vdash G$

$\sigma_2 : \langle a=1, b=0, n=0 \rangle$

$\sigma_3 : \langle a=1, b=0, n=1 \rangle$

F & G are equisatisfiable if the following holds:

1. Every satisfying assignment of F can be extended to the satisfying assignment of G.
   For every $\sigma \Vdash F$, $\exists \sigma'$ s.t. σ' extends σ to var(G) & $\sigma' \Vdash G$.

2. Every satisfying assignment of G can be projected on variables of F to get the satisfying assignment of F.

For every $\sigma' \Vdash G$, $\exists \sigma$ s.t. $\sigma = \sigma'_{\downarrow \{var(F)\}}$ & $\sigma \Vdash F$

* $F = p \vee (q \wedge r)$

$G = (p \vee t) \wedge (t \leftrightarrow (q \wedge r))$

$G' = (p \vee t) \wedge (t \rightarrow (q \wedge r))$

Q if F & G are equisatisfiable? YES

Q if F & G' are equisatisfiable?
YES

$F = p \vee \neg q$

$G = (p \vee \neg t) \wedge (t \rightarrow q)$

$G' = (p \vee \neg t) \wedge (t \leftrightarrow q)$

Q: if F is equisatisfiable to G ?

NO, $\mathcal{C}' : \langle p \mapsto 0, q \mapsto 1, t \mapsto 0 \rangle$

$\mathcal{C}' \models G$

$\mathcal{C} = \mathcal{C}' \downarrow_{var(F)} = \mathcal{C}' \downarrow_{\{p,q\}} = \langle p \mapsto 0, q \mapsto 1 \rangle$

$\mathcal{C} \not\models F.$

Q: F is equisatisfiable to G' ?

<u>YES</u>

K-SAT

CNF

$$F = C_1 \wedge C_2 \wedge C_3 \wedge \dots \wedge C_m$$

where

$$C_i = (l_1 \vee l_2 \vee \dots \vee l_k)$$

$$l_j = p \; ; \quad l_j = \neg p$$

where p is propositional variable.

2-SAT :

$$F = (x_1 \vee \neg x_2) \wedge (x_3 \vee x_4)$$

3-SAT

$$F = (x_1 \vee \neg x_2 \vee x_3) \wedge (x_3 \vee x_1 \vee x_4)$$

Q: Can you convert 4-SAT formula into a 3-SAT formula?

$F = (x_1 \vee x_2 \vee \neg x_3 \vee x_4) \wedge$
$(x_1 \vee x_3 \vee x_4 \vee \neg x_5)$

$\rightarrow (x_1 \vee x_2 \vee t_1) \wedge (t_1 \rightarrow (\neg x_3 \vee x_4))$
$\wedge (x_1 \vee x_3 \vee t_2) \wedge (t_2 \rightarrow (x_4 \vee \neg x_5))$

$\rightarrow (x_1 \vee x_2 \vee t_1) \wedge (\neg t_1 \vee \neg x_3 \vee x_4)$
$\wedge (x_1 \vee x_3 \vee t_2) \wedge (\neg t_2 \vee x_4 \vee \neg x_5)$

Q: Can you convert 3-SAT formula into 2-SAT?

<u>NO</u> :

<u>Try it out!</u> :)

Encoding of Graph coloring to SAT

Proper-coloring: An assignment of colors to the vertices of a graph so that no-two adjacent vertices have same color.

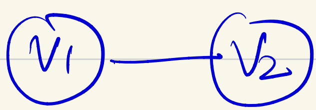
K-color: A proper-coloring involving a total # of k colors.

Q- Is the following graphs 2-colorable?

$G(V, E)$

$V = \{v_1, v_2\}$

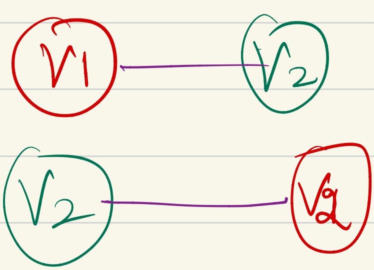
$E = \{(v_1, v_2)\}$



colors {Red, Green}

$k=2$

→ Yes



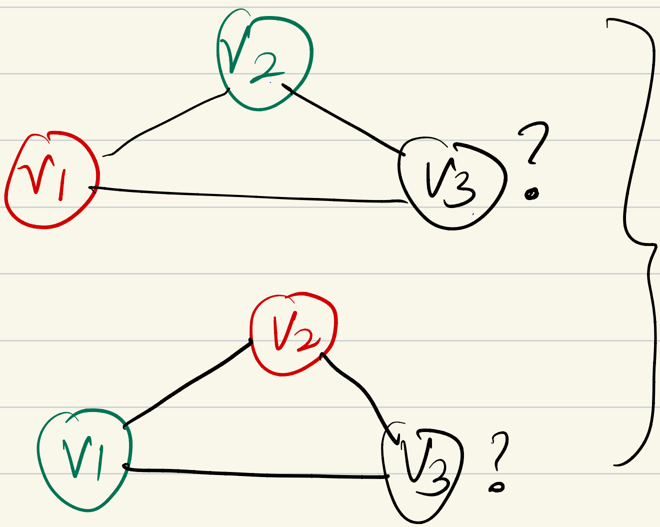
How about now?

$G = (V, E)$

$V = \{v_1, v_2, v_3\}$

$E = \{(v_1 v_2), (v_2 v_3), (v_3 v_1)\}$

k = 2, -colorable? {Green, Red}



NOT, 2-colorable

Encoding

k-coloring problem → problem of satisfiability

If the given graph $G=(V,E)$ is k-colorable? if yes, give us an proper-color assignment.

encode →

→ If the corresponding formula is satisfiable, then $G(V,E)$ is k-colorable & from the satisfying assignment, we can get the proper-color assignment.

→ If the formula is unsatisfiable then graph is not k-colorable.

Let us start with :

1

$G(V,E)$; $V = \{v_1, v_2\}$, $E = \{(v_1, v_2)\}$; $k=2$

For $k=2$:-

propositional variable : v_1, v_2

say: $v_i \vdash 0$ indicates, v_i has Red color &

$v_i \vdash 1$ indicates, v_i has green color.

$$F(v_1, v_2) \Rightarrow v_1 \oplus v_2$$

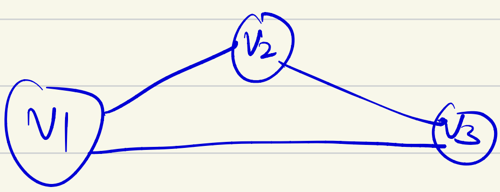
$$F(v_1, v_2) \Rightarrow (\neg v_1 \vee \neg v_2) \wedge (v_1 \vee v_2)$$

2

$G(V,E)$; $V = \{v_1, v_2, v_3\}$

$E = \{(v_1, v_2), (v_2, v_3), (v_1, v_3)\}$

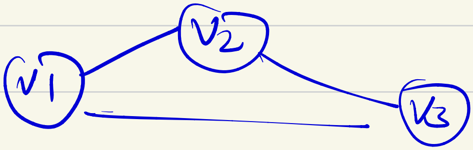
$k = 2$



$G(V, E)$; $V = \{v_1, v_2, v_3\}$

$E = \{(v_1, v_2), (v_2 v_3), (v_1 v_3)\}$

$\underline{K = 2}$



variables: (v_1, v_2, v_3)

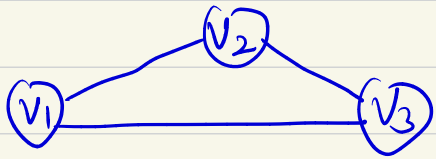
$F(v_1, v_2, v_3) :- (v_1 \oplus v_2) \wedge (v_1 \oplus v_3) \wedge (v_2 \oplus v_3)$

$\hookrightarrow$ UNSAT

v_1	v_2	v_3	$v_1 \oplus v_2$	$v_1 \oplus v_3$	$v_2 \oplus v_3$	F
0	0	0	0	0	0	0
0	0	1	0	1	1	0
0	1	0	1	0	1	0
0	1	1	1	1	0	0
1	0	0	1	1	0	0
1	0	1	1	0	1	0
1	1	0	0	1	1	0
1	1	1	0	0	0	0

$G(V,E)$; $V = \{v_1, v_2, v_3\}$

$E = \{(v_1 v_2), (v_2 v_3), (v_1 v_3)\}$



$k = 3$ say $\{$ Red, Green, Blue $\}$

Propositional Variables =

$= \{v_1^R, v_1^B, v_1^G, v_2^R, v_2^B, v_2^G, v_3^R, v_3^B, v_3^G\}$

Constraints:-

$\langle v_1, v_2 \rangle$ should not have the same color:-

$\rightarrow \neg v_1^R \vee \neg v_2^R$ (see, why $v_1^R \oplus v_2^R$ will not work?)

$\rightarrow \neg v_1^B \vee \neg v_2^B$

$\rightarrow \neg v_1^G \vee \neg v_2^G$

$\langle V_2, V_3 \rangle$ should not have same color:

$$\neg V_2^R \vee \neg V_3^R$$
$$\neg V_2^B \vee \neg V_3^B$$
$$\neg V_2^G \vee \neg V_3^G$$

$\langle V_1, V_3 \rangle$ should not have same color:

$$\neg V_1^R \vee \neg V_1^R$$
$$\neg V_1^B \vee \neg V_3^B$$
$$\neg V_1^G \vee \neg V_3^G$$

* How many such constraints?

For every edge, there will be k many such constraints $\rightarrow$ $\underline{|E| \cdot k}$

Every vertex takes exactly one color.

- $\rightarrow$ Every vertex takes at least one color.
- $\rightarrow$ Every vertex takes at most one color.

Every nodes takes at least one color?

$$v_1^R \vee v_1^B \vee v_1^G$$
$$v_2^R \vee v_2^B \vee v_3^G$$
$$v_3^R \vee v_3^B \vee v_3^G$$

How many such constraints $\rightarrow |V|$ many

Every nodes takes at most one color?

$$\neg v_1^R \vee \neg v_1^G$$
$$\neg v_1^R \vee \neg v_1^B$$
$$\neg v_1^G \vee \neg v_1^B$$

$$\neg v_2^R \vee \neg v_2^B$$
$$\neg v_2^R \vee \neg v_2^G$$
$$\neg v_2^G \vee \neg v_2^G$$

$$\neg v_3^R \vee \neg v_3^G$$
$$\neg v_3^R \vee \neg v_3^B$$
$$\neg v_3^G \vee \neg v_3^B$$

How many such constraints

$\rightarrow {}^{k}C_2 \cdot |V|$

Q Can we come up with a better encoding?

Q Do we really need exactly one color constraints?

Instead, does having at least one color constraints suffices?