CPSC 536F Feb 10

Valiant! # Perfect Matchings is complete for #P, i.e.

if you can count the number of perfect matchings in a graph in poly time, then

$P = NP$

Valiant's gadgets:

$$X = \begin{pmatrix} 0 & 1 & -1 & -1 \\ 1 & -1 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 0 & 1 & 3 & 0 \end{pmatrix}$$

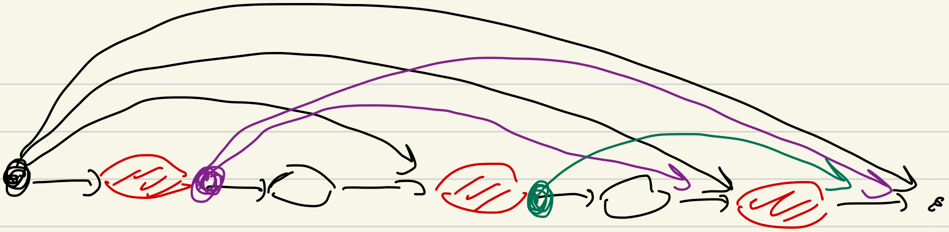
has: $\operatorname{Perm} X = 0$,

$\operatorname{Perm} X(1;1) = \operatorname{Perm} X(4;4)$

$= \operatorname{Perm} X(1,4;1,4) = 0$

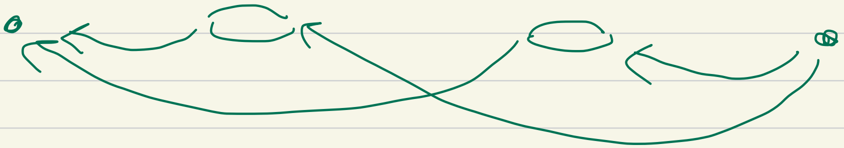
and

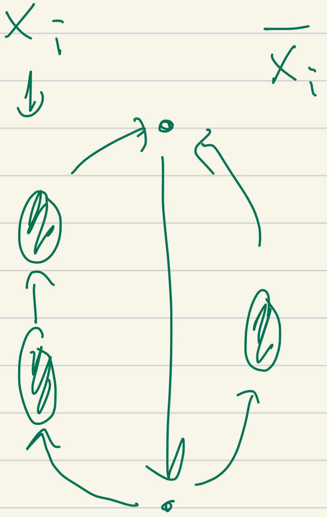
$\operatorname{Perm} X(1;4) = \operatorname{Perm} X(4;1) = 4$



plus downward arrows

Interchange





"track"

Boolean formulas

3SAT = { f that are in 3CNF form s.t.

f has a satisfying assignment }

3CNF :

$f = (\quad C_1 \quad) \text{ AND } (\quad C_2 \quad) \text{ AND }$

$\text{---} \quad \text{AND } (\quad C_m \quad)$

$C_i = lit_1^i \text{ OR } lit_2^i \text{ OR } lit_3^i$

literal = $x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n$

=

idea!

$f = (x_1 \text{ or } x_2 \text{ or } \bar{x}_3)$ AND

$(x_5 \text{ or } \bar{x}_1 \text{ or } x_2)$ AND

$\vdots$

3SAT = { f in 3CNF that

have a satisfying assignment }

satisfying assignment:

$x_1, \ldots, x_n$ set to T, F

s.t. $f(x_1, \ldots, x_n) = \text{true}$

$=$

$\#$3SAT is the problem of given f in 3CNF, print out the number of satisfying assignments!

$f = f(x_1, \ldots, x_n)$

$\#$ satisfying assignments $\leq 2^n$

So this $\#$ can be expressed in n-bits

For many "NP-complete problems"

3COLOUR, ---

If you can determine #3SAT,

then you can determine #3COLOUR

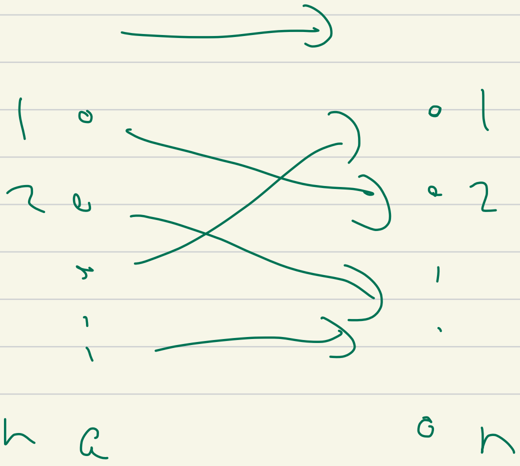
= # legal 3-colourings of a graph

Most people guess that 3SAT can't be solved in polytime (i.e. $P \neq NP$)

Certainly!

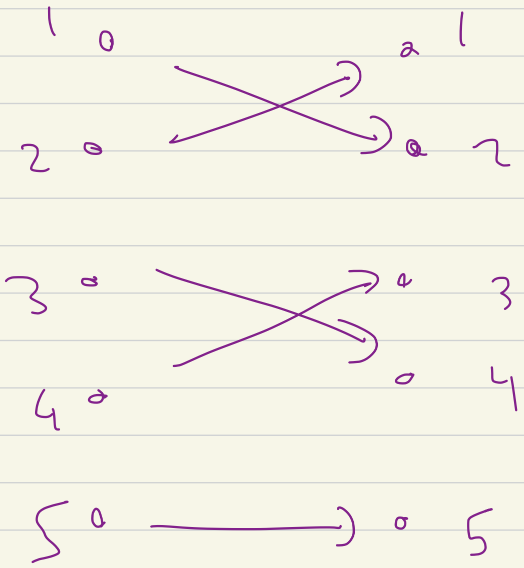
If you can solve #3SAT, then you can tell if this # is > 0 or = 0, so you can solve 3SAT.

On the other hand, given a bipartite graph



So all edges run →

a perfect matching is just a subgraph



There is a standard "augmenting path algorithm" to do this.

Surprise! if you can

solve # Perfect Matchings,

i.e., given n and subset of pairs $\{1, \ldots, n\}$, and you have a subroutine to count perfect matchings, then you can solve #3SAT, #3COLOUR, etc.

And $X \in \{0,1\}^{n \times n}$, so

$X = (x_{ij})_{i,j \in \{1, \ldots, n\}}$ then $\mathrm{Perm}(X) =$

So if you can count

#Perfect Matchings in poly time,

then P = NP; most people

think P ≠ NP, then

count #Perfect Matchings should

not have a short $\left\{\begin{array}{l}\text{formula}\\ \text{circuit}\end{array}\right.$

Today (2022, Feb) we can prove

min formula size for Perm $\geq c \cdot n^2$.

Today: Valiant's result!

if you're given an

$f = (\quad \checkmark \text{ OR } \downarrow \text{ OR } \curlyvee \quad)$ ← easier OR 3 literals

AND

$(\qquad\qquad)$

AND

⋮

AND

$(\qquad\qquad)$

even

$(\qquad) \leftrightarrow (\quad \text{OR} \quad \text{OR} \quad \text{OR} \_ \text{OR} \quad)$

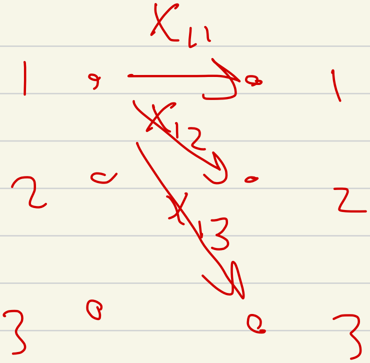
then # $x_1, \ldots, x_n$ s.t. $f(x_1, \ldots, x_n) = T$

Can be solved by solving a poly number of #PerfectMatching problems on computing a Permanent.

Permanent(X) : $X \in \mathbb{Z}^{n \times n}$

or

$\{0,1\}^{n \times n}$



$$\mathrm{Perm}\left(\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix},\right)$$

$$\coloneqq \sum_{\substack{\sigma:\{1,\dots,n\}\to\{1,\dots,n\} \\ \text{bijections (one-to-one}}} X_{1\,\sigma(1)} X_{2\,\sigma(2)} \cdots X_{n\,\sigma(n)}$$

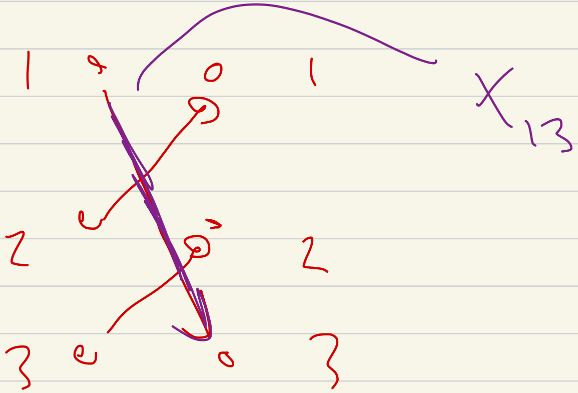
$$=$$

Here $\sigma:\{1,\dots,n\}\to\{1,\dots,n\}$

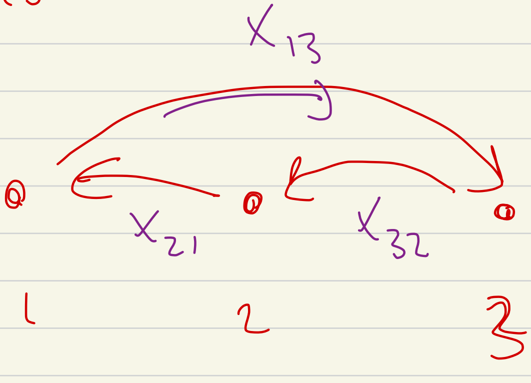
Can view σ as having a cyclic structure:

$$\begin{tikzcd} \sigma(1.) \arrow[r] & 1 \arrow[r] & \sigma(1)=3 \arrow[r] & \sigma(3)=2 \arrow[lll, bend right] \end{tikzcd}$$

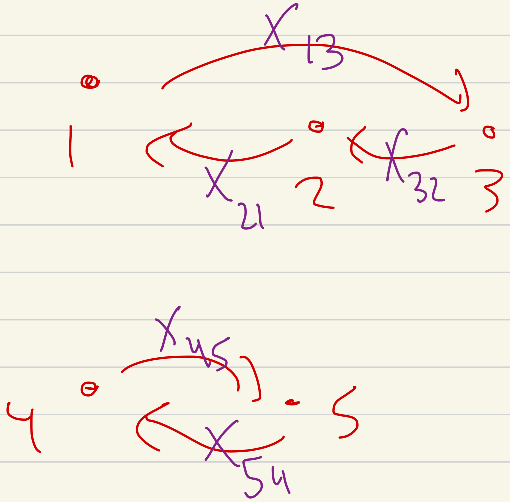
σ ! $1 \to 3 \to 2$ (arrow from 2 back to 1)



Think of a directed graph on 3 vertizes !



$n=5$ σ cyclic structure



$$X_{13}\cdot X_{32}\cdot X_{21}\cdot X_{45}\cdot X_{54}$$

Each σ gives you a subgraph of the complete digraph on n vertices that goes into each vertex once and out of each vertex once, covering all vertices as a union of cycles.

1st Observation:

If you can compute

$$\mathrm{Perm}(X),$$

where entries $X_{ij} \in \{-1, 0, 1, 2, 3\}$

then you can count

# solutions of a Boolean formula,

$f = f(x_1, \ldots, x_n)$ in 3CNF

form

Take an f in 3CNF form:

$(x_1 \text{ or } x_2 \text{ or } \overline{x_3})$ AND

$(x_5 \text{ or } x_9 \text{ or } \overline{x_1})$ AND

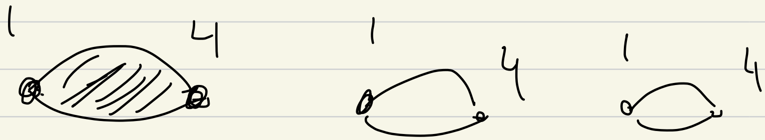
$(x_1 \text{ or } \overline{x_1} \text{ or } x_7)$ AND

$\vdots$

Create from this ~~a~~ permanent question of size roughly the size of the formula

Construction:

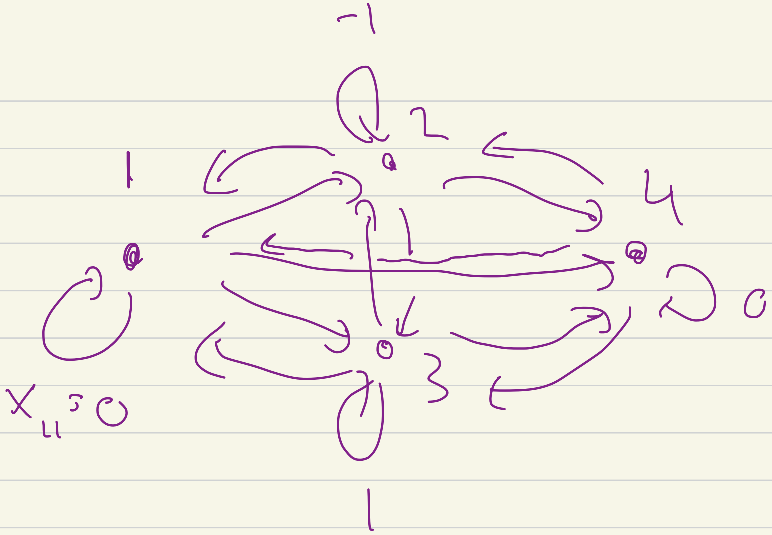
first gadget!



↓

Complete digraph on 4 vertices,
weight the edges as

$$X = \begin{pmatrix} 0 & 1 & -1 & -1 \\ 1 & -1 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 0 & 1 & 3 & 0 \end{pmatrix}$$



Claim!

$$\mathrm{Perm}(X) = 0$$

So

$$\mathrm{Perm}(Y) \text{ and } Y = \left(\begin{array}{cccc|c} 0 & 1 & -1 & -1 & \\ 1 & -1 & 1 & 1 & \\ 0 & 1 & 1 & 2 & \text{any} \\ 0 & 1 & 3 & 0 & \\ \hline & \text{any} & & & \text{any} \end{array}\right)$$

if term

$$\mathrm{Perm}(Y) = \sum_{\sigma:\{1,\dots,m\}\to\{1,\dots,m\}}$$

but sum over all

σ takes $\{1,\dots,4\} \longrightarrow \{1,\dots,4\}$

then

$$\underbrace{Y_{1\,\sigma(1)}\, Y_{2\,\sigma(2)} \;-\;-}_{\text{first four}} \qquad Y_{m\,\sigma(m)}$$

is

σ

Next!

$$\mathrm{Perm}\Big(\underbrace{X(1;1)}_{\substack{X \text{ with row \# } 1 \\ \text{col \# } 1 \\ \text{deleted,}}}\Big)$$

$$= 0$$

$$\mathrm{Perm}\big(X(4;4)\big)$$

$$\mathrm{Perm}\big(X(1,4;1,4)\big) = 0$$

e.g., $X(1,4;1,4) =$

$$\begin{pmatrix} 0 & 1 & -1 & -1 \\ 1 & -1 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 0 & 1 & 3 & 0 \end{pmatrix}$$

left with

$$\mathrm{Perm}\begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix} = 0$$

$=$

but

$$\left.\begin{array}{l} \mathrm{Perm}\ X(1;4) \\ \mathrm{Perm}\ X(4;1) \end{array}\right\} = 4$$

$$X = \begin{pmatrix} 0 & 1 & -1 & -1 \\ 1 & -1 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 0 & 1 & 3 & 0 \end{pmatrix}$$

$=$

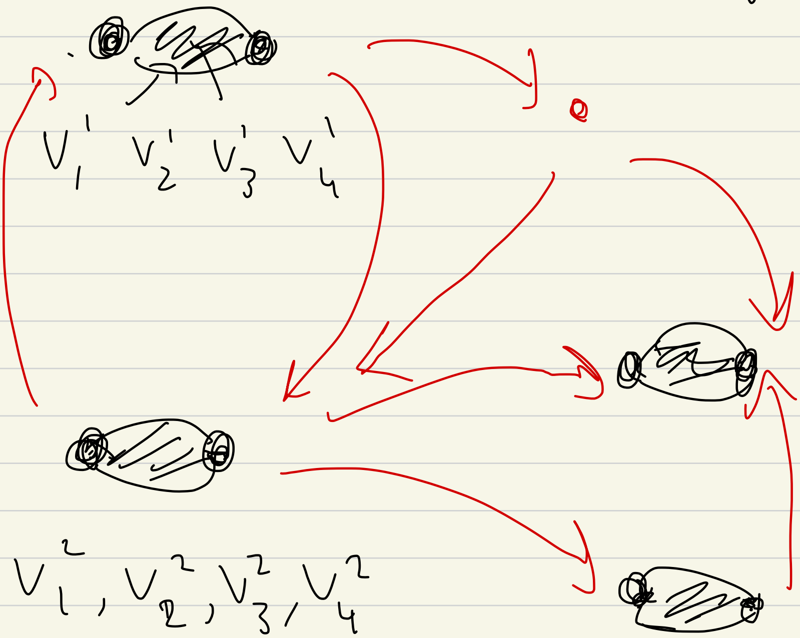
HW! The same can't happen to any square matrix if Perm $\overset{\text{replace}}{\rightsquigarrow}$ Det.

$\Leftarrow$

Claim:

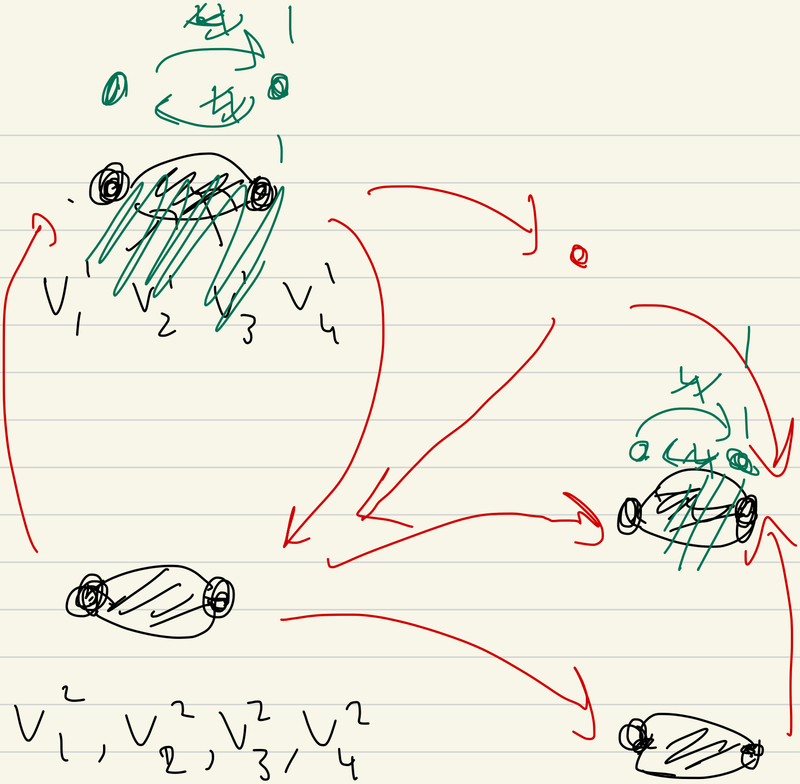
Graph:

with Valiant's weights



plus, then Perm ↑

is



i.e. when we sum over all

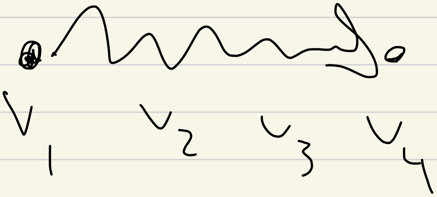
σ bijections on vertex set,

look at cyclic structure,

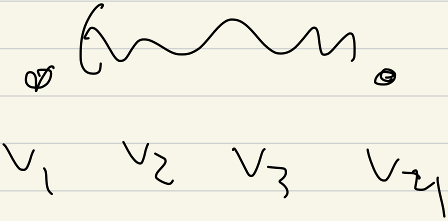
$$\mathrm{Perm}(\text{BigGraph}) = 4^{\text{Valid Pieces}} \; \mathrm{Perm}\left(\begin{matrix}\text{smaller}\\ \text{graph}\end{matrix}\right)$$

Perm' meaning each

has to traverse



OR

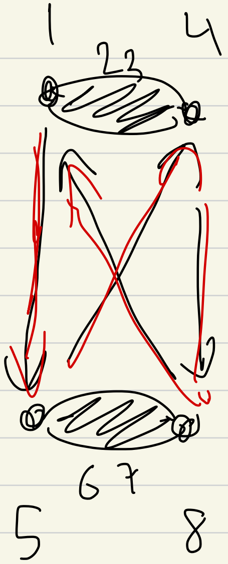


Break!

10:26 — 10:31

Think about this step...

Perm

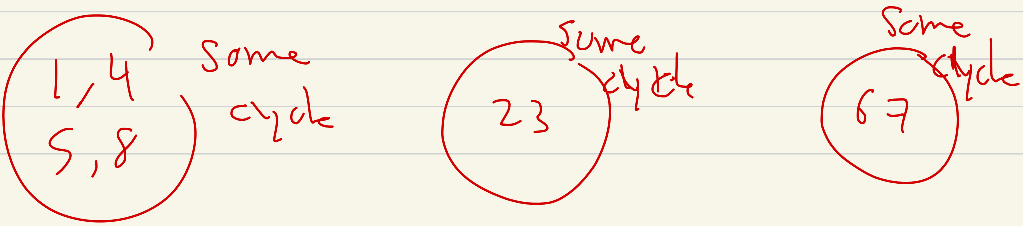


then

cyclic structure of σ

break into conn comp,

you can't have

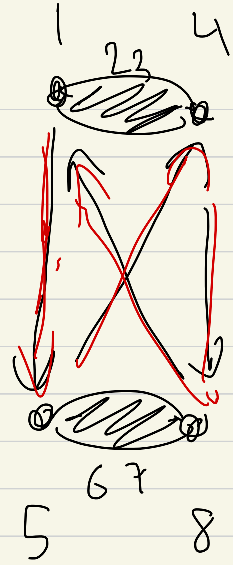


$$\mathrm{Perm}\left(X^{8 \times 8} \right)$$

$$= \sum_{\sigma\text{'s}}$$

$$= \sum_{\substack{\text{partitions} \\ \{1, \dots, 8\}}} \; \sum_{\substack{\sigma\text{'s with} \\ \text{a given} \\ \text{partition} \\ \text{of vertex} \\ \text{set, each} \\ \text{part in a cycle}}}$$

Perm(



)

$$= \sum x_{1\sigma(1)} \cdots x_{8\sigma(8)}$$

σ s.t.

cycles break $\{1, \ldots, 8\}$

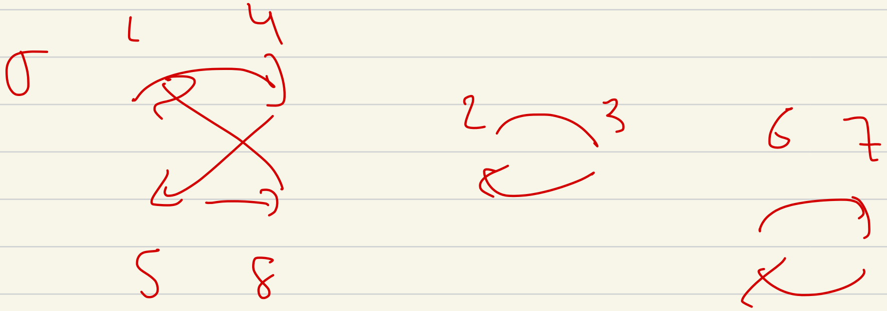
into cylce on 1, 4, 5, 8

" " 2, 3

" " 6, 7

$$+ \sum_{\sigma} \sim \text{---}$$ all other partitions

$\sigma : \{1, \dots, 8\} \to \{1, \dots, 8\}$



↕

partition of

$\{1, \dots, 8\}$

into

$\{1,4,5,8\}$ $\{2,3\}$ $\{6,7\}$

same cycle

same cycle

same cycle

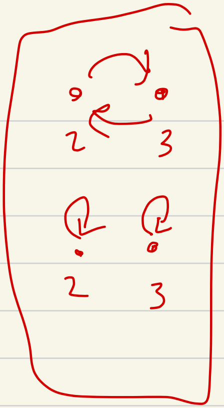
$\downarrow$

cycles on 1, 4, 5, 8

$$\begin{tikzcd}
1 \arrow[r] & 4 \arrow[r] & 5 \arrow[r] & 8 \arrow[lll, bend right]
\end{tikzcd}$$

$$\begin{tikzcd}
1 \arrow[r] & 5 \arrow[r] & 8 \arrow[r] & 4 \arrow[lll, bend left]
\end{tikzcd}$$

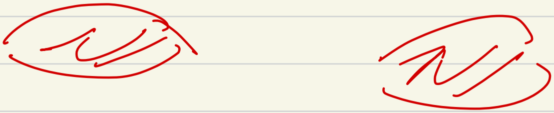
$\vdots$



Subdivide

$\sigma : \{1, \ldots, 8\} \to \{1, \ldots, 8\}$

which parts of



are taken to themselves

The rest:

$$(x_1 \text{ or } x_2 \text{ or } \overline{x_3})$$

AND

$$(x_5 \text{ or } x_7 \text{ or } \overline{x_9})$$

AND

$$\vdots$$

produce a graph G

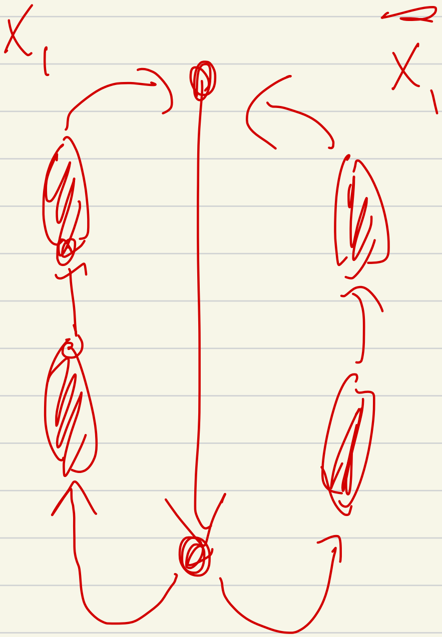
that counts # solutions of

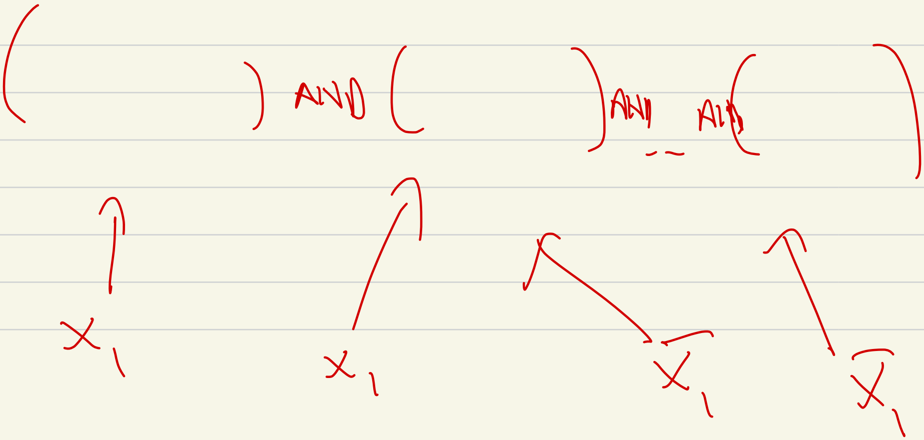
i.e. Perm(G)

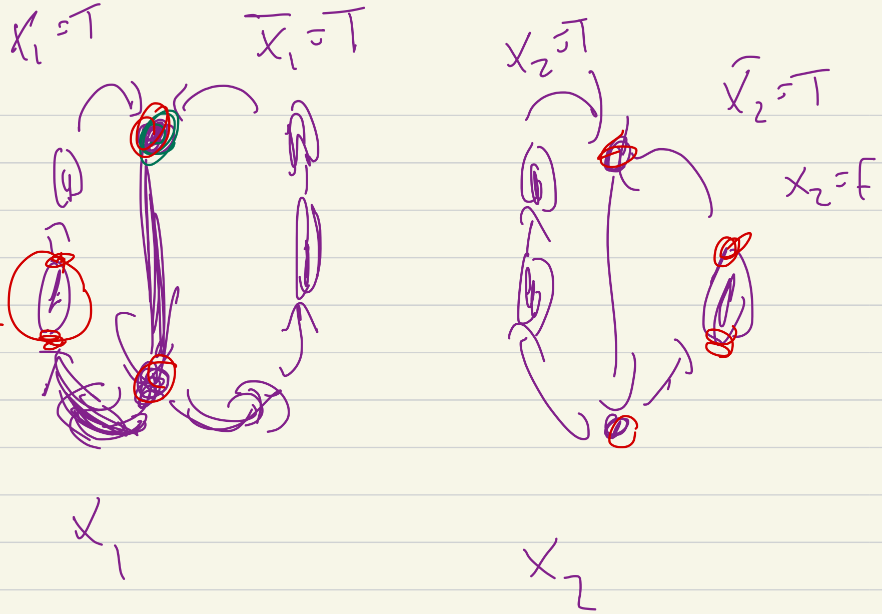
Part of Y:

Each variable $x_1, \dots, x_n$

Say x_1 and in the clauses







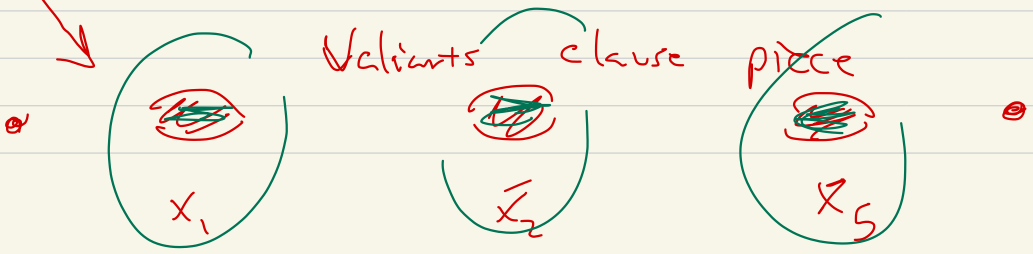
tracks:

for each clause (x_1 or $\bar{x}_2$ or x_5)

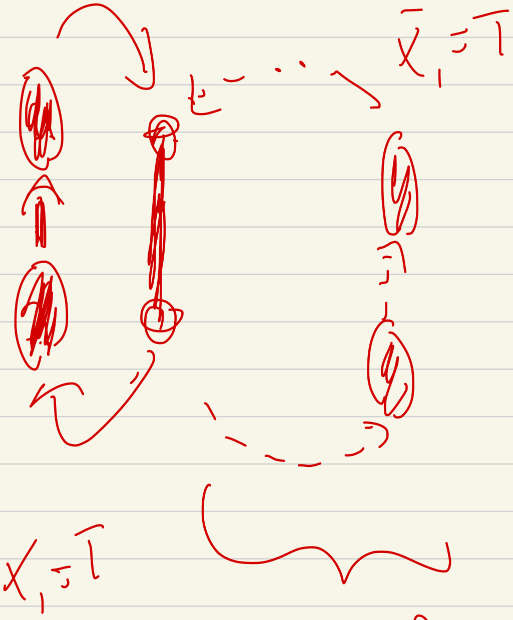
interchange:

x_1 on $\bar{x}_2$ cr x_5

Valiants clause piece



If you choose $x_1 = T$



none of these edges are traversed in the x_1 piece

Class ends

The Complexity of Computing the Permenent

1979