I. Claim. Vertex Cover $\le_p$ Ind. Set
Ind. Set $\le_p$ Vertex Cover.

Claim. In any graph, a set $S \subseteq V$ is independent iff $\bar{S}$ is a ~~co~~ vertex cover.

Pf. If S is independent, then no edge has both ~~is contained~~ its endpoints in S. $\Leftrightarrow$ any edge has at least one endpoint in $\bar{S}$. $\Leftrightarrow$ $\bar{S}$ is a vertex cover.

Pf (reduction) A graph has a vertex cover of size $\le k$ $\Leftrightarrow$ the same graph has an ind. set of size $\ge n-k$.

The reduction: $\underbrace{(G, k)}_{\text{vertex cover}} \longrightarrow \underbrace{(G, n-k)}_{\text{ind. set problem}}$

Proof of correctness: If G has a vertex cover S of size $\le k$, then by claim, $\bar{S}$ is an ind. set.

$$|S| + |\bar{S}| = n, \quad |S| \le k \Rightarrow |\bar{S}| \ge n-k.$$

Hence, if (G, k) as a vertex cover problem has answer yes, then $(G, n-k)$ as an ind set problem has answer yes.

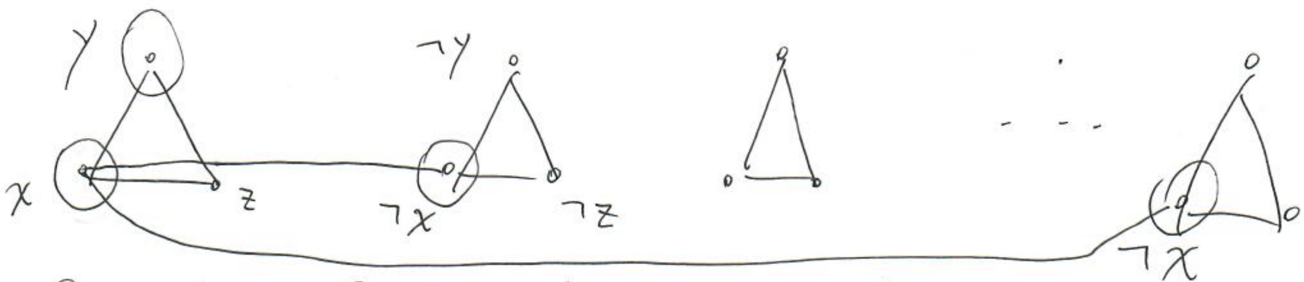
If G has no ~~ind. set~~ vertex cover of size $\le k$, then, suppose $(G, n-k)$ as an ind. set problem has answer yes, that implies a vertex cover of size $\le k$. $\Rightarrow\Leftarrow$

Pf. As argued, Ind. Set is in NP.

① A poly time reduction from 3-SAT to Ind Set

② The 3-SAT instance has answer YES

$\Leftrightarrow$ The Ind Set instance has answer YES.



II. Given a 3-SAT problem with m clauses,

① for each clause, construct 3 nodes. (corresponding to the 3 literals in that clause)

② add an edge between any node representing a variable and any other node representing its negative.

(We need: 3-SAT problem has a solution iff the ind. set problem has ind. set of size $\geq m$.)

③ connect all three nodes in each set corresponding to a clause.

The reduction is obviously poly-time.

III. Correctness.

1. If a 3-SAT instance is satisfiable, the constructed ind. set instance has ~~is~~ an ind. set of size $\geq m$.

<u>Pf</u>. ~~Be~~ Given an assignment of values to the variables we can select from each clause a literal that evaluates to TRUE. Pick the corresponding node in the ind. set. instance, add it to S.

S is of size m, and S is an ind. set.

Because, obviously no edge in each triangle is contained in S; no other edge connecting a variable and its negative can be contained in S. (otherwise the assignment was not valid.)

2. $\exists$ Ind. set of size $m \Rightarrow$ 3-SAT is satisfiable.

Given an ind. set of size m, then each node comes from a diff. triangle. Assign value to the corresponding variables in each triangle.

This ~~is~~ assignment is valid, and all clauses are satisfied. $\boxtimes$

<u>Cor</u>. Set cover is NP-complete.

Pf. Take ~~a~~ the given graph. Take any node u and split it into two copies u', u''. u' has all the outgoing edges of u, and u'' all the incoming edges of u.

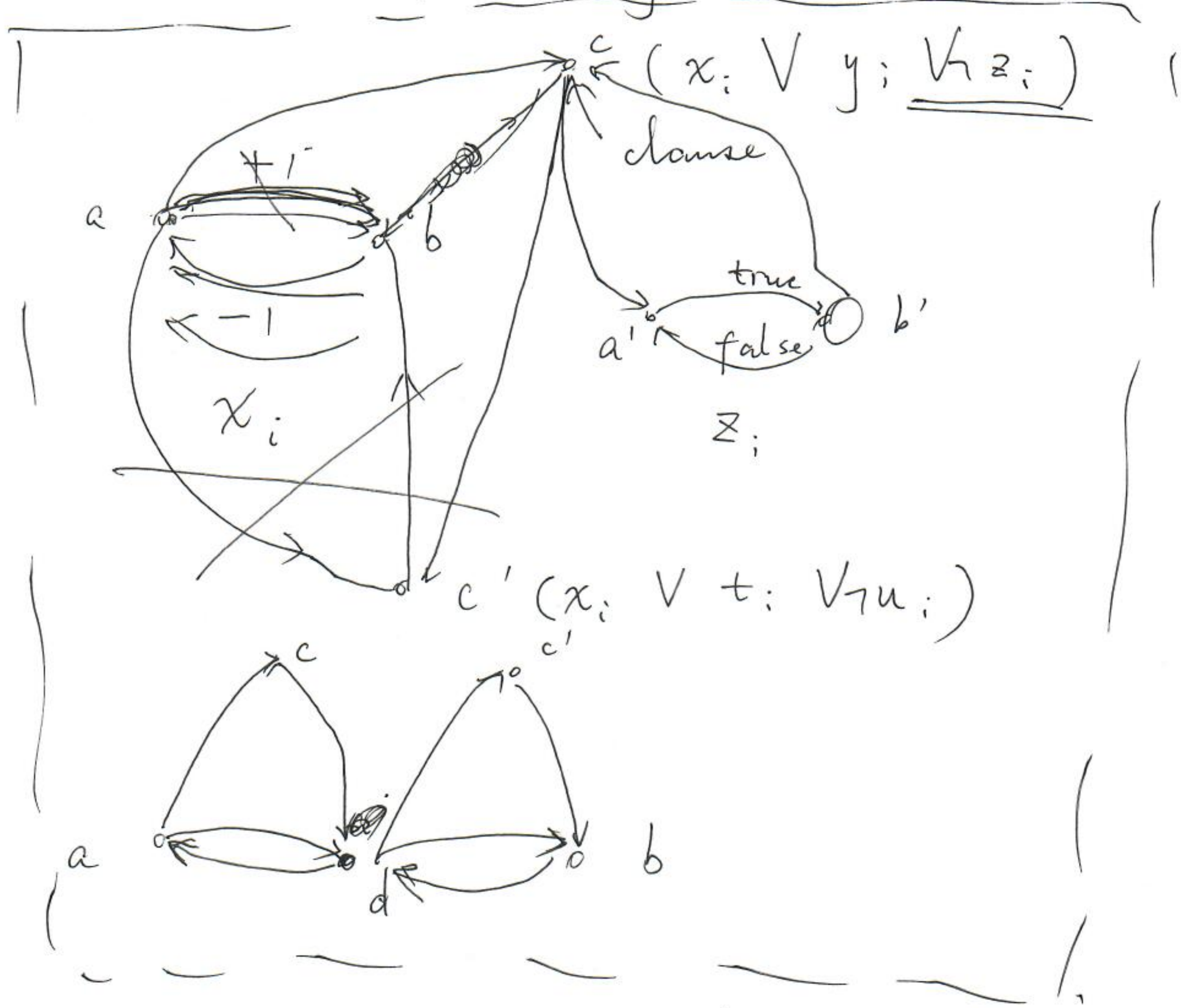
If Hamiltonian Cycle exists in the original graph, we could start from u' in the new graph and ~~the~~ follow the cycle ~~at~~ until we get back to u''. This is a H. path in the new graph.

If H. path exists.

Pf. Hamiltonian Cycle $\in NP$ (immediate)

Certificate: ~~a path~~ a cycle.

3-SAT $\le_p$ H. Cycle.



Given a 3-SAT formula, $\begin{pmatrix} m = \#\text{ clauses}, \\ n = \#\text{ variables} \end{pmatrix}$

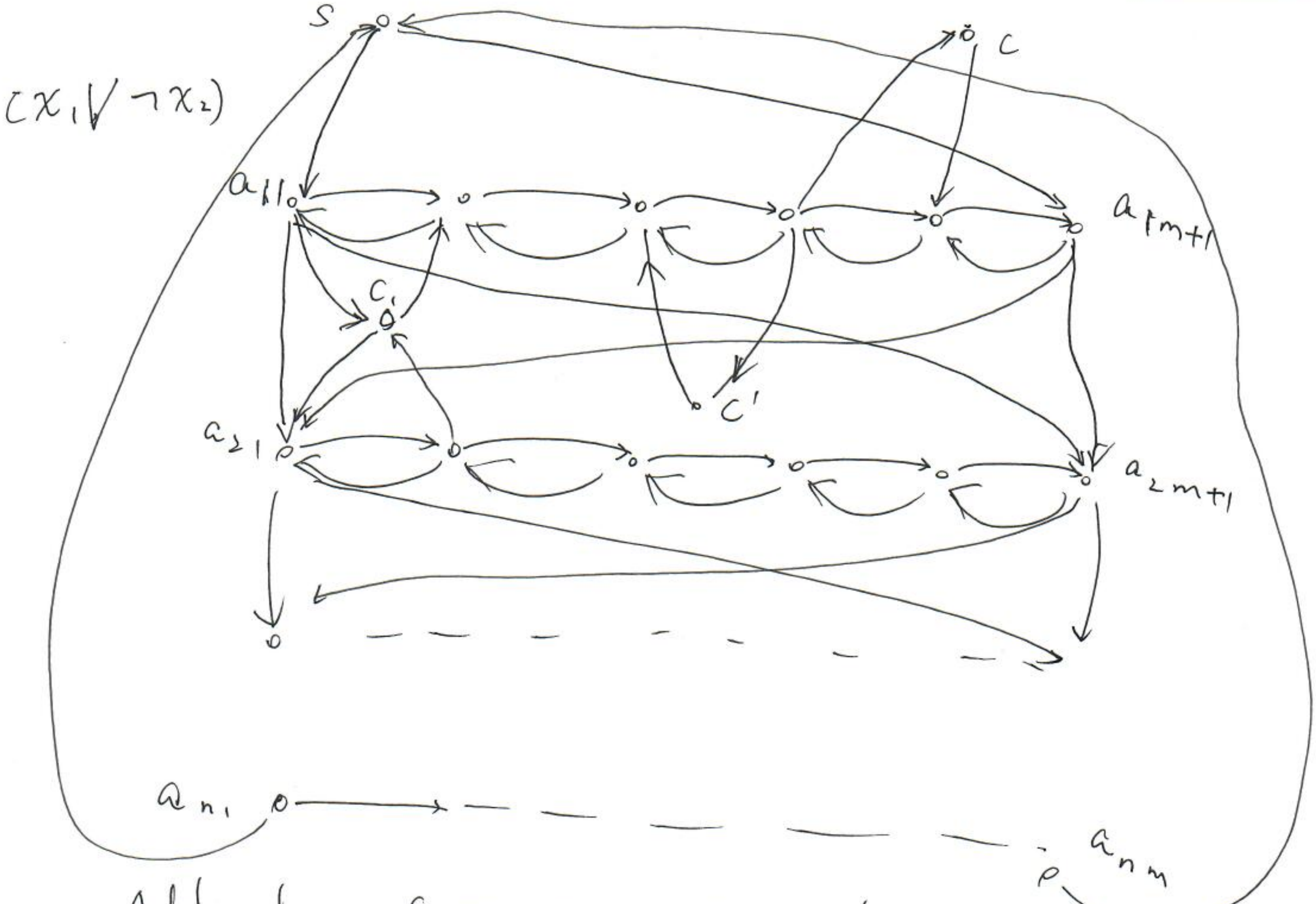
~~st~~ add a start node s.

For each variable x_i, add m nodes

$a_{i1}, \cdots, a_{im+1}$, add edges from a_{ij} to a_{ij+1},

$j = 1, \cdots, m$ ~~+1~~. (interpret $m+2$ as 1)

add edges from a_{ij} to a_{ij-1}, (interpret ~~B~~ 0 as m).



Add edges from s to a_{11} and a_{1m+1}

and edges from $\underline{a_{i1}, a_{im+1} \text{ to } a_{i+1,1} \text{ and } a_{i+1,m+1}}$

$(i = 1, \cdots, n-1)$.

and edges from a_{n1}, a_{nm+1} to s.

For the j-th clause, add node c_j, if it contains x_i, then add edges

a_{ij} to c_j, and c_j to a_{ij+1}.

if the clause contains $\neg x_i$, then add edges

a_{ij+1} to c_j, and c_j to $a_{i,j}$.

($\Rightarrow$): If 3-SAT has a satisfying assignment,

. - - - - -

($\Leftarrow$): If H. cycle exists, translate back to an assignment.

<u>Pf</u> (TSP). TSP $\in NP$.

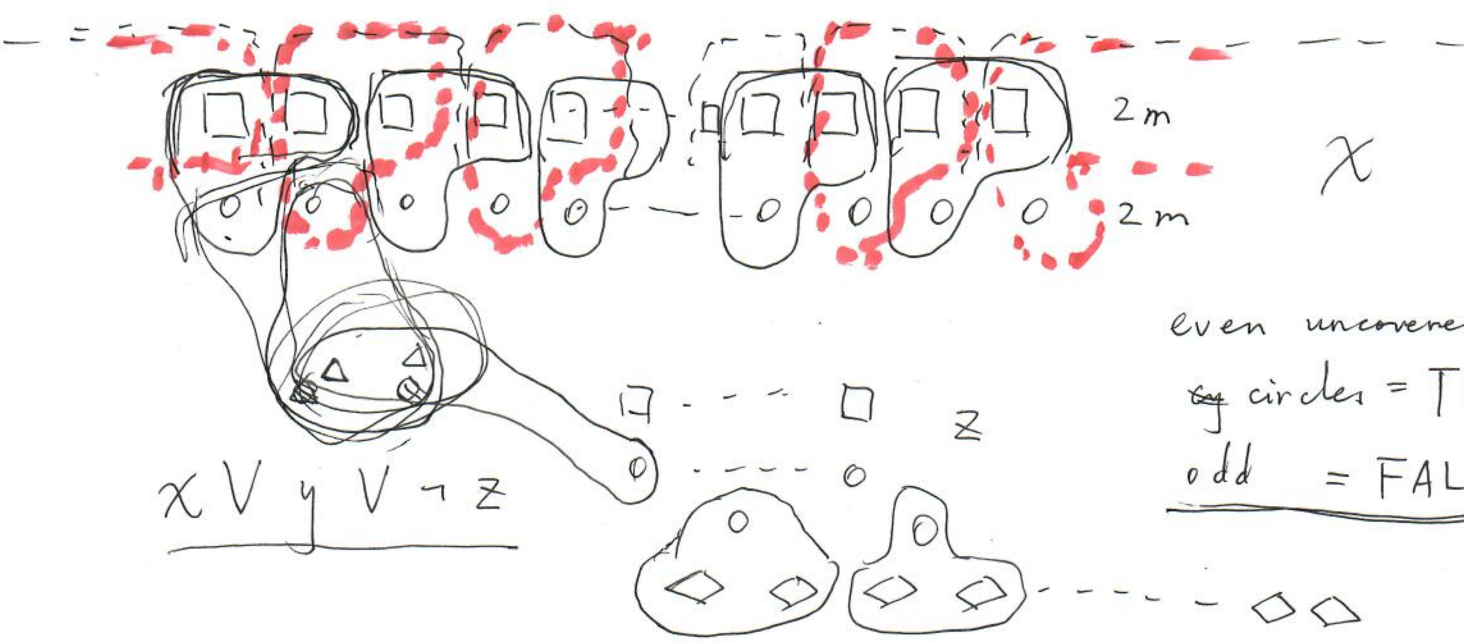
H. ~~Cycle~~ Path $\leq_p$ TSP.

Given directed graph, have cost 1 on each edge.

$\exists$ H. Path $\Leftrightarrow$ The TSP problem has a tour with cost $n-1$.

<u>Pf</u>. (3DM) 3DM $\in NP$. (immediate)

3-SAT $\leq_p$ 3DM.



even uncovered ~~cy~~ circles = TRUE

odd = FALSE

Reduction: We introduce 3 classes of nodes:

I. Variable nodes: for each of the n variables, we introduce $4m$ nodes, where m is the number of clauses. Let these nodes for variable x_i be $a_{i1}, a_{i2}, \cdots, a_{i,2m}$ and

$b_{i1}, b_{i2}, \cdots, b_{i,2m}$.

II. Clause nodes: for clause j, we have two nodes c_{j1}, c_{j2}.

III. Clean-up nodes: we have $2(n-1)m$ clean-up nodes $d_1, d_2, \cdots, d_{2(n-1)m}$.

Superedges:

For every variable x_i we have edges

Type A: $(a_{i,2k+1}, a_{i,2k+2}, b_{i,2k+1})$, for $k = 0, 1, \cdots, m-1$

Type B: $(a_{i,2k}, a_{i,2k+1}, b_{i,2k})$, for $k = 1, 2, \cdots, m$ (interpreting $2m+1$ as 1)

For every clause j, if the clause includes x_i, we have edge

$(b_{i,2j}, c_{j1}, c_{j2})$

if the clause includes $\neg x_i$, we have edge

$(b_{i,2j-1}, c_{j1}, c_{j2})$

For every b_{ij}, $i = 1, \cdots, n$, $j = 1, \cdots, 2m$, we have edges

$(b_{ij}, d_{2k-1}, d_{2k})$, for $k = 1, \cdots, (n-1)m$.

Proof of correctness:

($\Rightarrow$): Given a satisfying assignment for the SAT formula, if ~~a~~ variable x_i is TRUE, we cover the corresponding a_{ij} nodes using type A edges, otherwise we use type B edges. Each clause j has at least one literal evaluating to TRUE. If it's x_i, we cover c_{j1}, c_{j2} using the edge $(b_{i,2j}, c_{j1}, c_{j2})$; if it's $\neg x_i$, we cover c_{j1}, c_{j2} using $(b_{i,2j-1}, c_{j1}, c_{j2})$. We cover all the rest of the b_{ij} nodes using the edges involving the clean-up nodes. By construction these are all valid edges and it's straightforward to check that all nodes are ~~in~~ covered by exactly one edge selected.

($\Leftarrow$) Given a perfect matching, each variable's nodes must be covered using ~~only~~ either only type A edges or only type B edges. If type A edges are used, we assign the corresponding variable TRUE, otherwise we assign FALSE. Since all clause nodes are covered, at least one literal in each clause evaluates to TRUE by this assignment. Therefore we have a satisfying assignment.

Pf. Subset Sum $\in NP$.

3-D Matching $\leq_p$ Subset Sum.

$$
\begin{array}{ccc|ccc|ccc}
 & X & & & Y & & & Z & \\
1 & \cdots & n & n+1 & \cdots & 2n & 2n+1 & \cdots & 3n
\end{array}
$$

X	Y	Z	
0 1 0 · · · · 0	0 · · 1 0 · · 0	0 · · · 1 0 · · · 0	e_1
1 0 · · · · · 0	1 · · · · · 0	1 · · · · · 0	e_2
1 0 · · · · · 0	0 · · · · · · 1	0 · · · · 1 · · · 0	e_3
1 1 0 · · · · 0	1 0 · · 1 0 · · · 0	1 0 · · 1 · · · · 0	e_1+e_2
2 1 0 · · · · 0	1 · · 1 0 · · · 0 1	1 · · · 1 · · · 1 · · · 0	$e_1+e_2+e_3$

1^{st}

2^{nd}

If we could find a subset of binary vectors whose sum is 1 1 1 · · · · · · 1 1 1, then we are "almost" done. ⟵ Intuition

$|T| = m$

Given X, Y, Z and edge set T, for each edge that contains the i-th element in X, j-th elem in Y and k-th element in Z, we create a number

$$(m+1)^{k-1} + (m+1)^{n+j-1} + (m+1)^{2n+i-1};$$

we set a target $\sum_{i=0}^{3n-1} (m+1)^i$.