Problem 2

(a) 1° Suppose $\{\tau(M), \tau(M_i)\} \neq \{n+1, n+2\}$ and without lost of generality: $\tau(M) < \tau(M')$

2° if $\tau(M) < n+1$, then for $\tau(V_i) = 2n+2 \Rightarrow |\tau(V_i) - \tau(M)| > n+1$ which is illegal

(M' cannot be $2n+2$ or for $\tau(V_i) < n+1$ is more than $n+1$ away)

3° if $\tau(M') > n+2$, then for $\tau(V_i) = 1 \Rightarrow |\tau(V_i) - \tau(M)| > n+1$ which is illegal

(M cannot be 1 or for $\tau(V_i) > n+2$ is more than $n+1$ away)

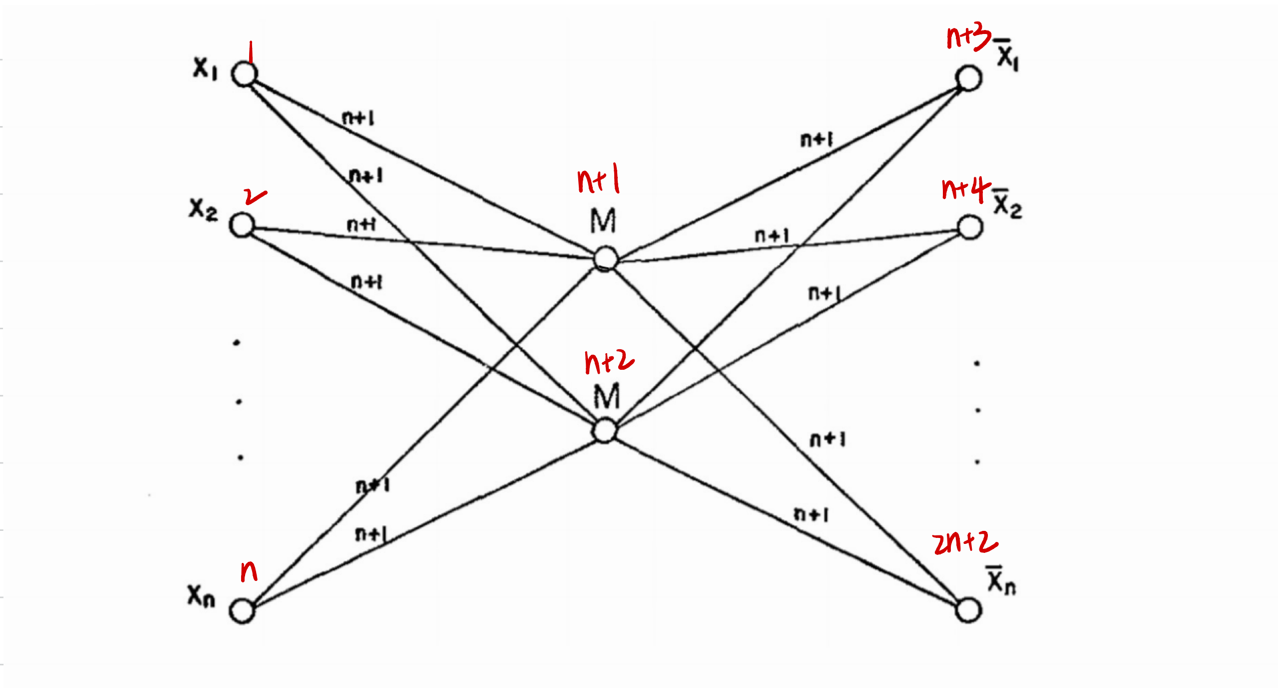
4° Also, if $\{\tau(M), \tau(M')\} = \{n+1, n+2\}$, it satisfy $|\tau(M) - \tau(V_i)| \leq n+1$ for all i,

X_1	X_2	...	X_n	M	M'	$\bar{X}_n$	...	$\bar{X}_2$	$\bar{X}_1$
1	2		n	$n+1$	$n+2$	$n+3$		$n+1$	$2n+2$

$d = n+1$ $d = n+1$

$\Rightarrow \tau$ is solvable for LAP[G]

5° Thus from 4° & contradiction 1°, 2°, 3° we know, $\{\tau(M), \tau(M')\} = \{n+1, n+2\}$. QED



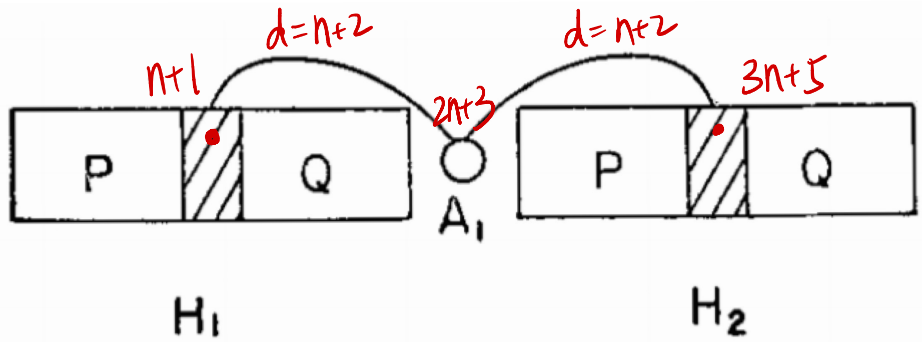
(b) 1° Without loss of generality, let $\tau(V_1)$ is smaller than $\tau(V_2)$. And from (a) we know, μ, μ' of H_1 is $n+1$, $n+2$ and $\tau(V_1) = \{1, 2, \cdots, 2n+2\}$. Also, since A_1 is connected to μ, μ' by a $n+2$ weigted edge $\Rightarrow \tau(A_1) = (n+1) + (n+2) = 2n+3$.

So $\tau(V_2)$ will start from $2n+4$ to $4n+5$. Also, μ, μ' of H_2 is $3n+4$, $3n+5$ which can be shifted from (a). Also, distance of A_1 and μ, μ' is within $n+2$.

2° We can also exchange H_1, H_2 so we know: $\tau(V_1) = \{1, 2, \cdots, 2n+2\}$ or $\{2n+4, \cdots, 4n+5\}$

Also, $\tau(A_1)$ has to be $2n+3$.

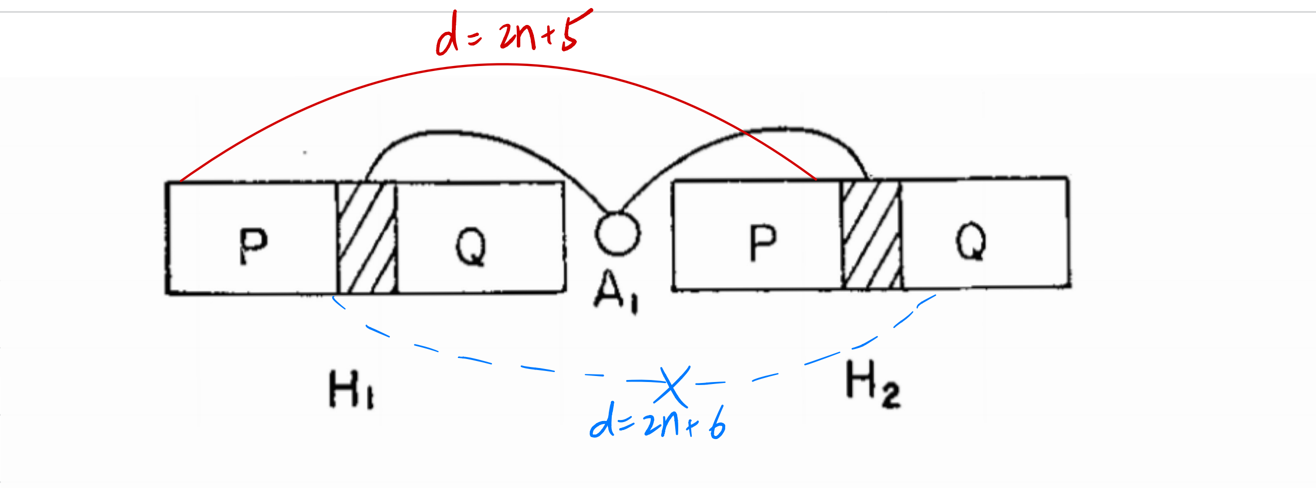
3° So, from 1°, 2° we know: τ is solvable for LAP[G] if the condition mentioned in 1°, 2° is catered. QED.



(c) 1° If we join every vertex of P in H_1 to the corresponding vertex in H_2, we can make sure that $P_1 = P_2$, hence $Q_1 = Q_2$. So, consider the boundary of the weighted edge, assuming we are mapping the last element of P in H_1 to the first element of Q in H_2

2° Thus, the distance of $P_{\text{last of } H_1} \to Q_{\text{first of } H_2} = |M| + |M'| + |Q_{H_1}| + |A_1| + |P_{H_2}| + |M| + |M'| + 1$ equals $2n+6$, so if we set the weight to $2n+5$, we can make sure its corresponding vertex is not in Q of H_2. Also, from previous (a). (b) we know, it is also impossible to be M, M' of H_2. So by adding edge weighted of $2n+5$, we can make sure it is in P of H_1.

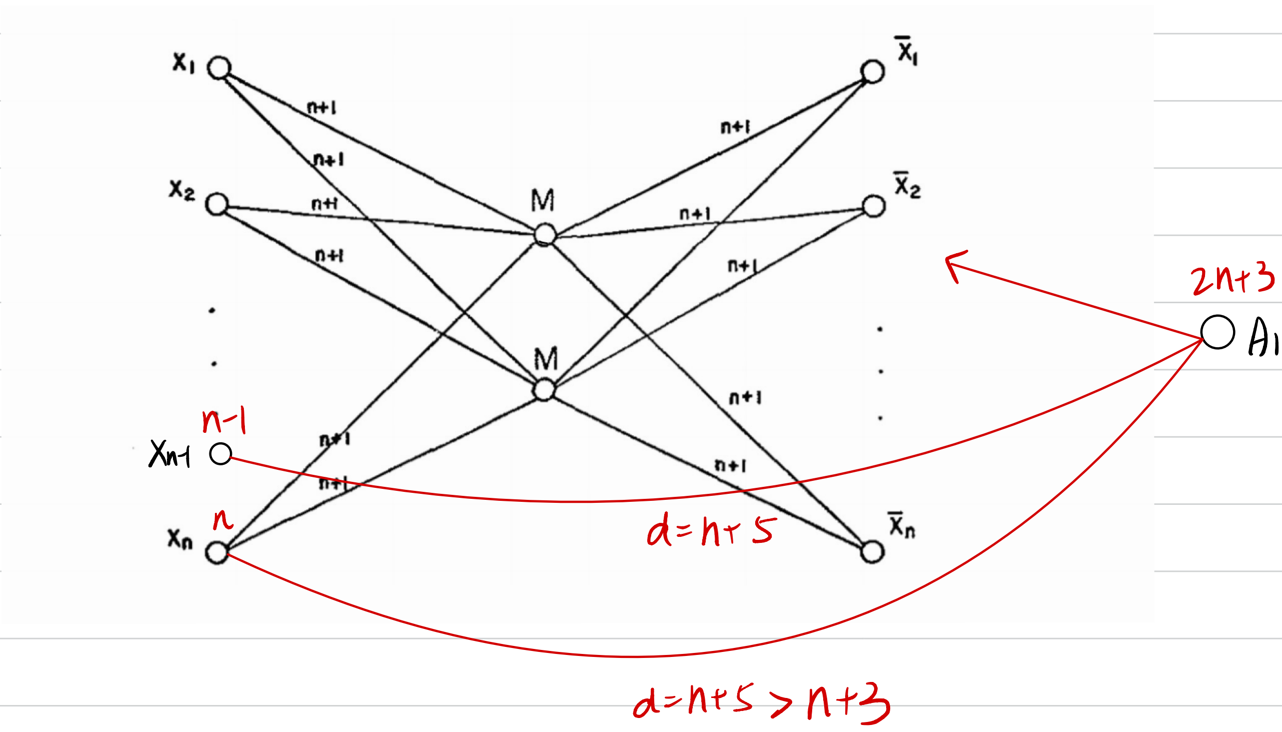
3° Also, it is solvable from (b), and adding edges weighted of $2n+5$ from P in H_1 to corresponding in H_2 is also legal. Thus, I is solvable for LAP[G]



(d) 1° In order to let at least one vertex in S' to be in Q_1, add weighted edge from A_1 to S'. So, consider the boundary of the weighted edge, assuming the three vertex are the last three elements in P_1.

2° Thus, the distance of $A_1 \rightarrow P_{\text{last 3rd element in } H_1}$ is $3 + |M| + |M'| + |Q|$ equals $n+5$, so if we set the weight of the edge to $n+4$, we can make sure that not all three vertex is in P_1, so at least one vertex is in Q_1. Thus $S' \cap Q_1$ is not null.

3° Also, it is solvable from (c), and adding edges weighted of $n+4$ from A_1 to S' is also legal. Thus, τ is solvable for LAP[G].

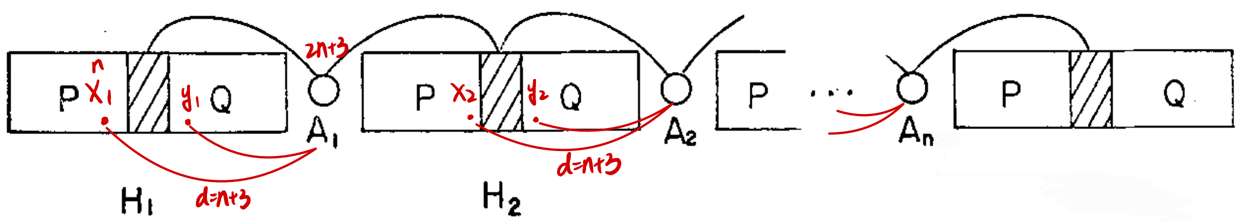


(e) 1° From (c) we know, by joining H_1 and H_2 with A_1 and add those edges, we can make sure $P_1 = P_2$. Also, we apply this to $H_3, H_4, \cdots H_{n+1}$ along with $A_2, A_3, \cdots, A_n$. We can make sure $P_1 = P_2 = \cdots = P_n = P_{n+1}$.

2° So, if we want to achieve that exactly one of x_i, y_i is in P. So, we can add an weighted edge from x_i, y_i to A_i, and make sure every (x_i, y_i) pair isn't both in P_i or Q_i. Without loss of generality, let x_i be in P and y_i in Q. let x_i be the last element in P_i so by adding an weighted edge of distance $A_i \to x_i$, we can make sure that y_i will be in Q_i

3° Thus, distance $A_i \to x_i$ is $1 + |M| + |M'| + |Q|$ equals $n+3$. So, only one of x_i, y_i can get into P_i and the other has to be in Q_i. Also, for every i, only one of x_i, y_i can be in P_i and $|P| = n$ we know, half of them must lie in P. So we can seperate every pair of (x_i, y_i).

4° Also, each is solvable from (c), and $H_3, H_4, \cdots$ is only a shift of index. Thus each is solvable and adding edges weighted of $n+3$ from A_i to x_i, y_i are also legal. Thus, τ is solvable for LAP[G].



(f) 1° Consider $H_1 \sim H_{n+1}$ and connect them as we did in (e), then we can seperate every x_i and $\bar{x}_i$. (Joining A_i to $x_i, \bar{x}_i$ with edge of $n+3$)

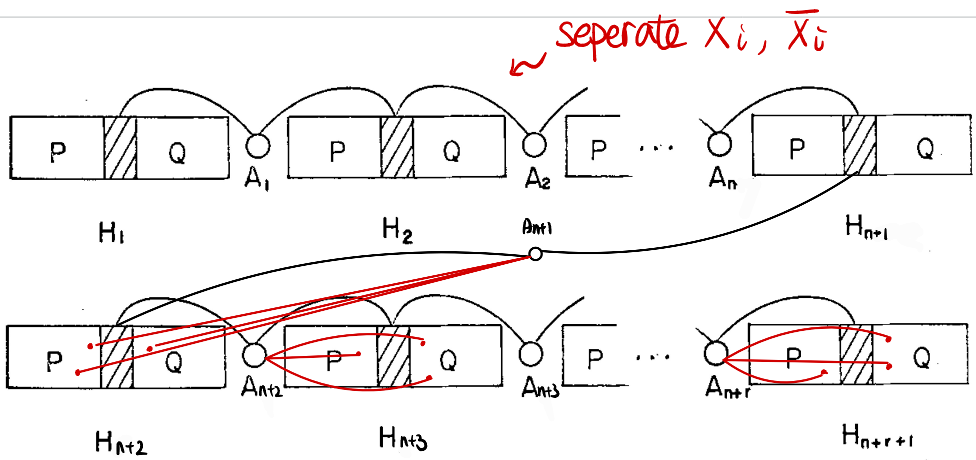
2° Consider $H_{n+2} \sim H_{n+r+1}$, we can convert r clauses of $B \to H$. Assuming literals in P are 0. And S_i' is the F_i clause of B we can apply (d) by connecting A_{n+i} to S_i' with edge of $n+4$.

3° Then at least one liteal of S_i' will fall in Q_{n+1+i}. So the output of i-th clause (H_{n+1+i}) will be 1.

4° So, we can find S_i' for every F_i, if 3-CNF-SAT is solvable according to previous partition. Thus, we can find partition for all clause to be solve. Thus LAP[G] is solved.

5° If LAP[G] is solvable, we can take every S' and map to F_i and make 3-CNF-SAT solvable. Also, if 3-CNF-SAT is unsolvable, we'll fail to find a partition. And if we fail to label, it implies 3-CNF-SAT is unsolvable.

Thus LAP[G] is solvable iff 3-CNF-SAT is solvable.



(g) < LAP $\in$ NP-hard >

1° Given n literal and r clause, we can reduce to LAP in polynomial time. ($r < n^3$, input length $= O(r)$)

2° Thus, construct $n+1$ H and n A, also, it takes $2n+2$ to generate vertex and less than $C^{2n+2}_{2} = 2n^2+3n+2$ to construct edges. Then for every A, we generate less than $2n+2$ edges. So, first $n+1$ H and A we use $n \cdot O(n^2) = O(n^3)$.

3° Then for the following r clauses, construct r H and rA and construct edges, we take $r \cdot O(n^2) = O(n^2 \cdot r)$.

4° Thus, we acquire a graph in $O(n^3+n^2 r) = O(n^5)$. So we turn 3-CNF-SAT to LAP problem in $O(n^5)$. Thus, the reduction is polynomial time.

5° Also, from (f) we know that LAP[G] is solvable iff 3-CNF-SAT is solvable.

< LAP $\in$ NP >

1° Given a G and potential LAP(G) solution, checking all it vertice and edge can check whether it is legal or not in $O((n+r)\cdot(n^2+2n^2+3n+2)) = O(n^5)$

< LAP $\in$ NPC >

1° Since LAP $\in$ NP and LAP $\in$ NP-hard, LAP $\in$ NPC.