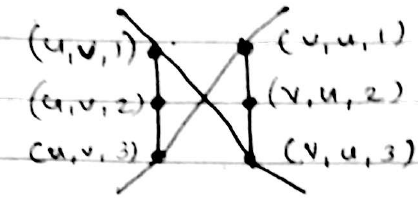
1. Given simpler graph.



- Textbook gadget states the property that if cycles starts/enters (u,v,1) then it should ~~exit~~ exit through (u,v,6) which is last vertex of the same side of the graph.
- Considering above simpler graph, their is no gaurantee that if a cycle enters from (u,v,1) then it will surely exit through (u,v,3). Possibility is the cycle may exist either (v,u,1) or (v,u,3)
- If all the vertices are to be traversed then valid cycle from (u,v,1) through (v,u,3) can exist
- But this violets the property stated by textbook.
- By the same reduction we cannot gaurantee that vertex cover in G exist based on hamilton cycle existence without the property in G'.

Therefore, above scenario proves that the gadget / widget can't be replaced with the simpler problem.

2. The reduction from 3CNF-SAT gives/forms equations that has variables, clauses and SAT problems. boolean nature.

Initially, the rightmost 3 columns are computed/set as;

$$\begin{array}{ccc} 0 & p & 0 \\ 0 & p & 0 \\ \hline 1 & q & 0 \end{array}$$

Considering boolean expression/equation of 3 CNF SAT given in the textbook.

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

Each pair of literals x_i and $\overline{x_i}$, added to left side of formula.

$$\begin{array}{l} 0\ d_i\ 0\ 1\ y_i\ 0\ c_i y_i\ 0\ b_i y_i\ 0\ a_i\ 0 \\ 0\ e_i\ 0\ d_i y_i\ 0\ c_i y_i\ 0\ b_i y_i\ 0\ a_i\ 0 \\ \hline \overline{x_i}\ 0\ e_i z_i\ 0\ d_i z_i\ 0\ x_i z_i\ 0\ b_i\ a_i\ 0 \end{array}$$

Thus,

$b_i = 2a_i$

$x_i = 2b_i + c$ (where c = carry $(y_i + y_i) \in \{0,1\}$)

$= 4a_i + c = c \pmod 4$

$d_i = 2c_i + c$

$e_i = d_i + 1 + c$

$= 2c_i + 1 + 2c$

$\overline{x_i} = d_i + e_i$

$= 4c_i + 1 + 3c$

$= 3c + 1 \equiv 1 - c \pmod 4$

Likewise for each clause add: $(x_1 \vee \neg x_1 \vee \neg x_2)$

$$\begin{array}{l} 0\ u_{ab}\ 0\ x_1\ 0\ 1\ r_i\ 0\ g_i w_i\ 0\ f_i\ 0 \\ 0\ \neg x_2\ 0\ \neg x_1\ 0\ h_i r_i\ 0\ g_i w_i\ 0\ f_i\ 0 \\ \hline 0\ t_i\ 0\ u_{ab}\ 0\ t_i s_i\ 0\ h_i x_i\ 0\ g_i\ 0 \end{array}$$

$\therefore t_i = u_{ab} = \{2,3\} \bmod 4$

2. → $(x_3 \vee x_2 \vee x_4)$

$$\begin{array}{l} 0\ u_{ab}\ 0\ x_3\ 0\ 1\ r_i\ 0\ g_i w_i\ 0\ f_i\ 0 \\ 0\ x_4\ 0\ x_2\ 0\ h_i r_i\ 0\ g_i w_i\ 0\ f_i\ 0 \\ \hline 0\ t_i\ 0\ u_{ab}\ 0\ h_i s_i\ 0\ h_i x_i\ 0\ g_i\ 0 \end{array}$$

$\therefore t_i = u_{ab} = \{1,2,3\} \bmod 4$

→ $(\neg x_1 \vee \neg x_3 \vee x_4)$

$$\begin{array}{l} 0\ u_{ab}\ 0\ \neg x_1\ 0\ 1\ r_i\ 0\ g_i w_i\ 0\ f_i\ 0 \\ 0\ x_4\ 0\ \neg x_3\ 0\ h_i r_i\ 0\ g_i w_i\ 0\ f_i\ 0 \\ \hline 0\ t_i\ 0\ u_{ab}\ 0\ h_i s_i\ 0\ h_i x_i\ 0\ g_i\ 0 \end{array}$$

$\therefore t_i = u_{ab} = \{1,2,3\} \bmod 4$

The reduction holds for NP completeness for any mod multiple of 4. Solution to above puzzle gives satisfying assignment to our problem, no matter what would be base of puzzle.

Reference - www.cs.umd.edu/~hajagha/ALB14.pdf.

3. (a) Given chain of n struts that are linked together into a cycle by hinges can be laid out flat on a single line with the help of folding method.

- To check if it's a valid folding will take O(n) polynomial time. Hence chain folding is in NP

3. (b) Considering PARTITION $X\{x_1, x_2 \dots x_n\}$, we can construct a chain with n struts

- For CHAIN-FOLDING problem we can achieve this by setting length of strut
- Valid folding will help us to calculate angle between two adjacent struts.
- If angle between x_i and $x_{(i+1) \bmod n}$ is equal to π then set $a_i = 1$ (in a single line)
- If angle is 2π then set $a_i = -1$.
- Considering above scenario, we can claim CHAIN-FOLDING is NP-hard

(c)

$$F(\langle S, t \rangle) = \begin{cases} \{1,2\} & \text{if } t > \alpha \\ \{1,2,3\} & \text{if } t = 0, \alpha \\ S & \text{if } t = \alpha/2 \\ S \cup \{\alpha + t, 2\alpha - t\} & \text{otherwise} \end{cases}$$

where

$$\alpha = \sum_{x \in S} x$$

- There is no subset whose sum of elements will be t such that $t > \alpha$. Therefore, we construct an impossible instance of PARTITION.
- A valid instance of PARTITION will be created if $t = 0$
- If $t = a/2$, then valid folding for PARTITION will be equivalent to SUBSET-SUM.
- Lastly, if none of the above is true then we add elements $\alpha + t$, $2\alpha t$ to the instance of PARTITION x.
- Thus PARTITION is NP hard.