W, 11/13/19

Fall'19 CSCE 629

Analysis of Algorithms

Fang Song
Texas A&M U

Lecture 29

- NP-complete problems

Credit: based on slides by A. smith & K. Wayne

Quiz

For each of the following statements, decide T/F/Unknown.

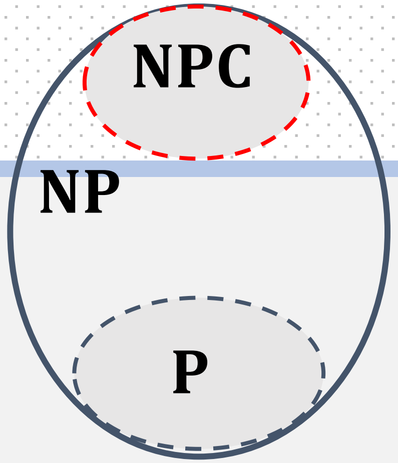
a) All problems in $\mathbf{P}$ can be solved in n^{2019} time.

b) If a problem is in $\mathbf{NP}$, then it cannot be solved in n^{2019} time.

c) If a problem is $\mathbf{NP-Complete}$, then the best algorithm for it takes $2^{\Omega(n)}$ time.

d) There exists a problem in $\mathbf{NP}$ but not in $\mathbf{P}$.

NP-Completeness



Def. A problem Y is NP-Complete if

1. $Y \in \mathbf{NP}$
2. $\forall X \in \mathbf{NP},\ X \leq_{P,Karp} Y$

Theorem. Suppose Y is NP-Complete, then Y is solvable in poly-time iff. $\mathbf{P} = \mathbf{NP}$

Pf.

- ($\Leftarrow$) If $\mathbf{P} = \mathbf{NP}$, then Y can be solved in poly-time since $Y \in \mathbf{NP}$
- ($\Rightarrow$) If Y is solvable in poly-time, consider any $X \in \mathbf{NP}$.

  Since $X \leq_{P,Karp} Y$, X has a poly-time algorithm as well

  I.e., $\mathbf{NP} \subseteq \mathbf{P} \rightarrow \mathbf{P} = \mathbf{NP}$

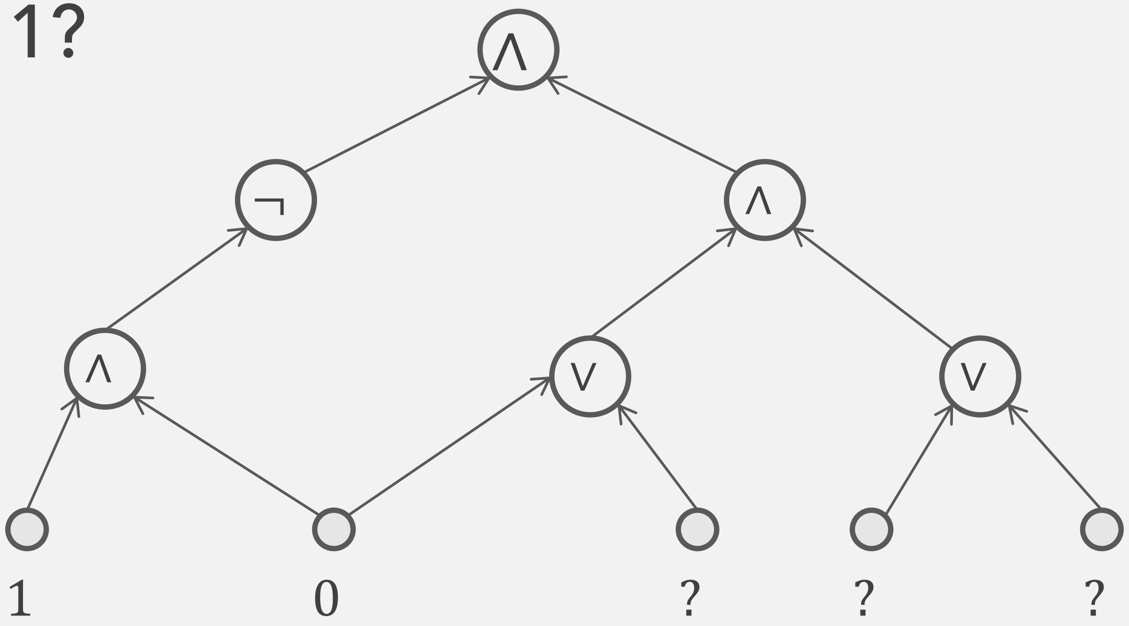
Fundamental question: Are there natural NP-complete problems?

The "first" NP-Complete problem

Theorem. Circuit−SAT is NP-Complete [Cook 1971,Levin 1973]

Input. A combinational circuit built out of AND/OR/NOT gates

Goal. Decide if there is a way to set the circuit inputs so that the output is 1?



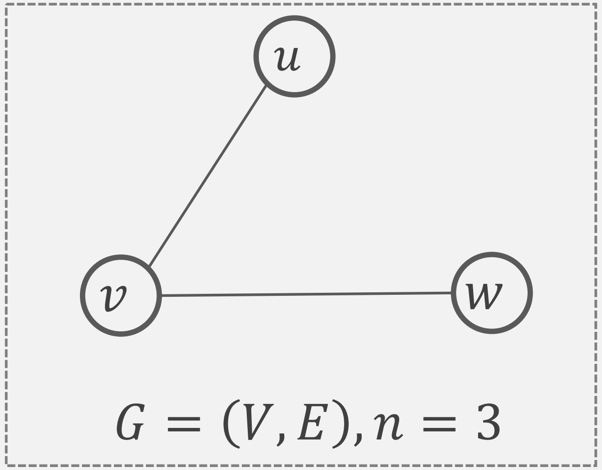
hard-coded inputs   inputs

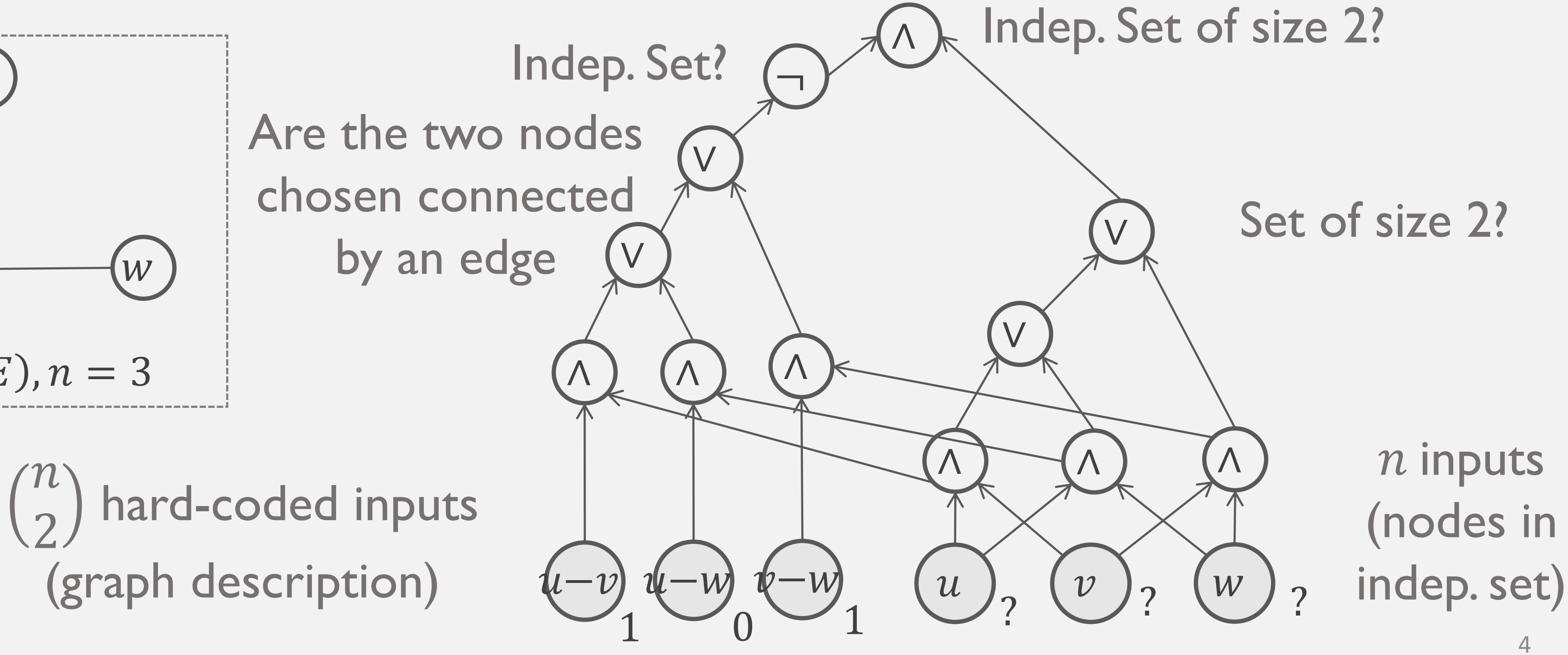
Stephen Cook   Leonid Levin

Example

Given. Graph G

Construction. Circuit K whose inputs can be set so that K outputs true iff. graph G has an independent set of size 2





Establishing NP-Completeness

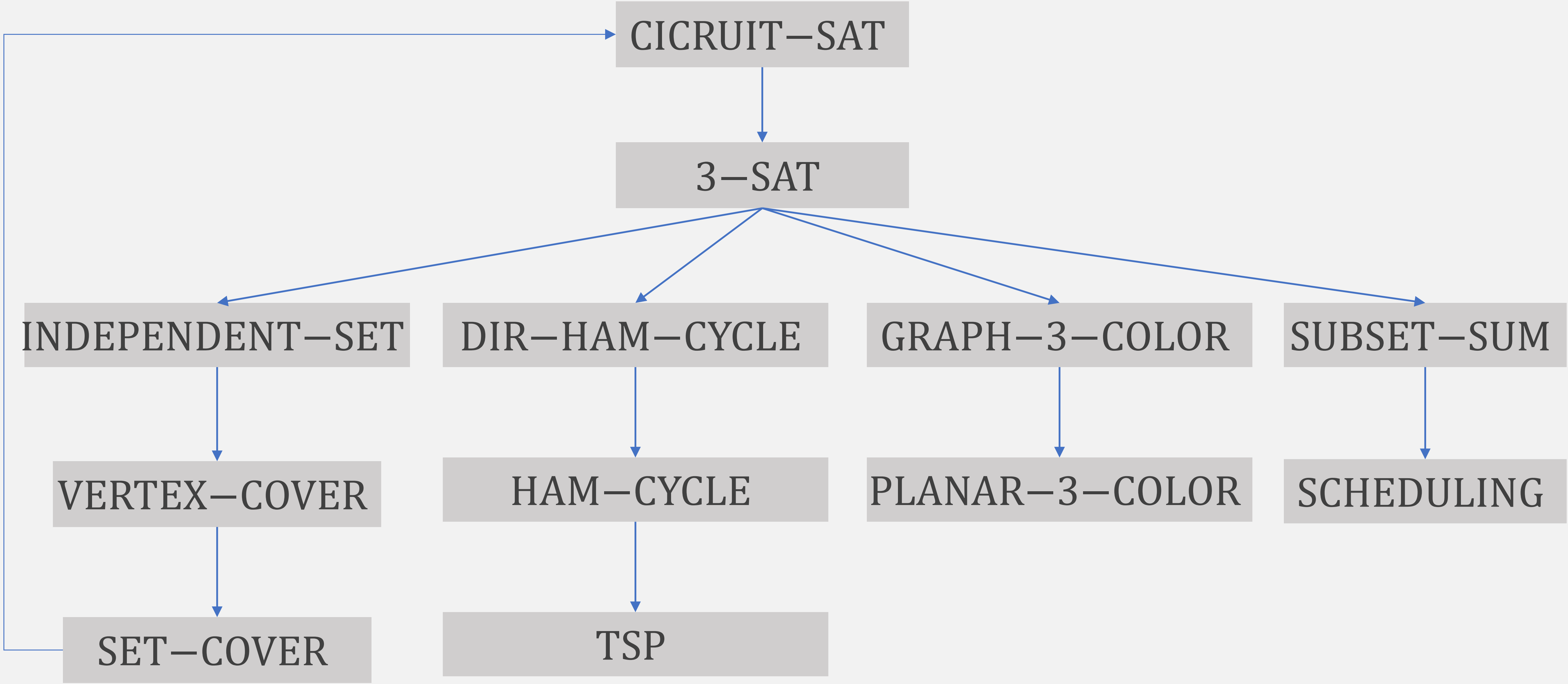
Once we establish first "natural" NP-complete problem, others fall like dominoes ...

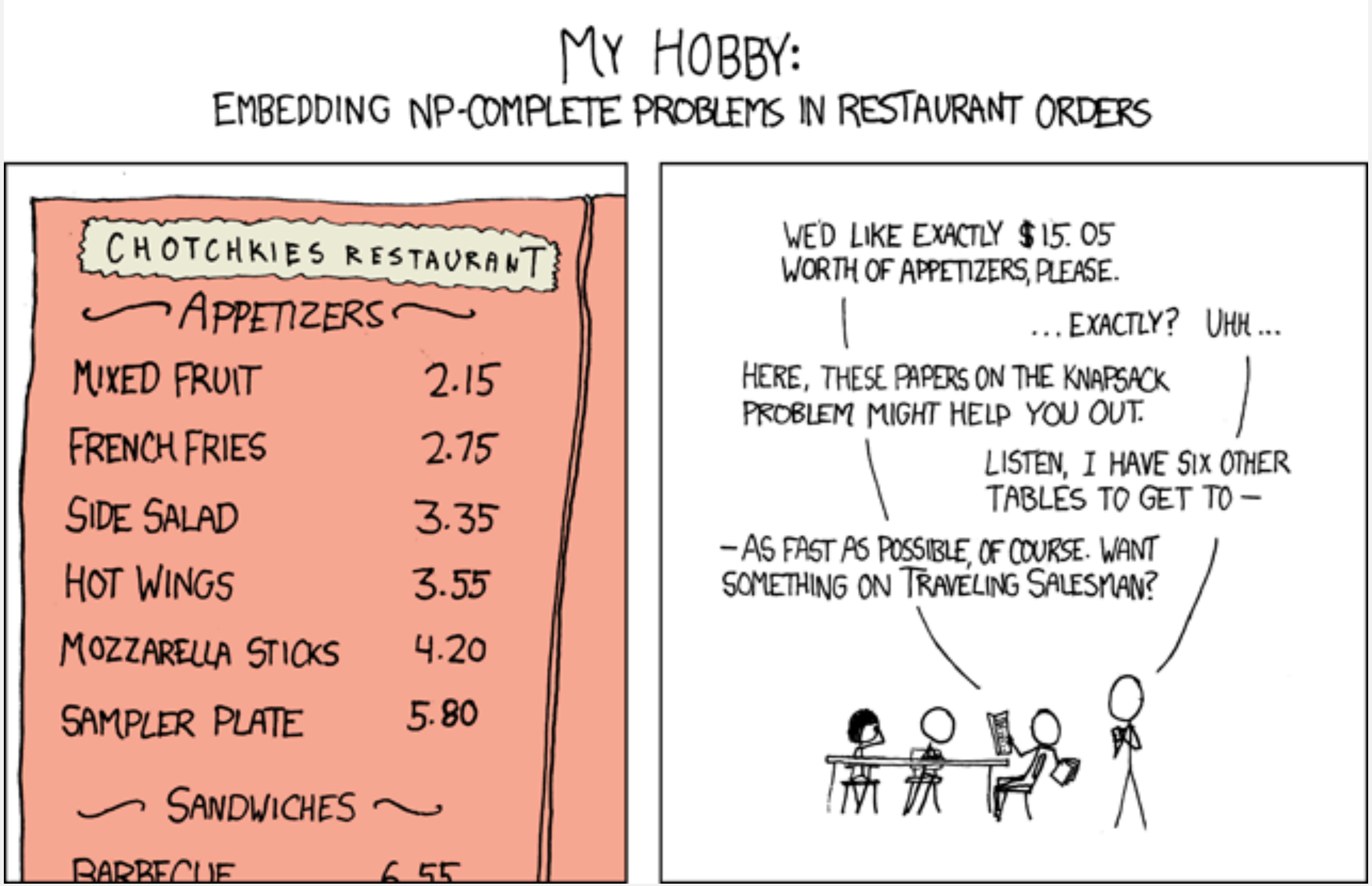
Recipe to establish NP-Completeness of problem Y

1. Show that $Y \in \mathbf{NP}$
2. Choose an $\mathrm{NP-complete}$ problem X
3. Prove that $X \leq_{P,Karp} Y$

Justification. If X is an NP-complete problem, and Y is a problem in $\mathbf{NP}$ with the property that $X \leq_{P,Karp} Y$ then Y is NP-complete (by transitivity)

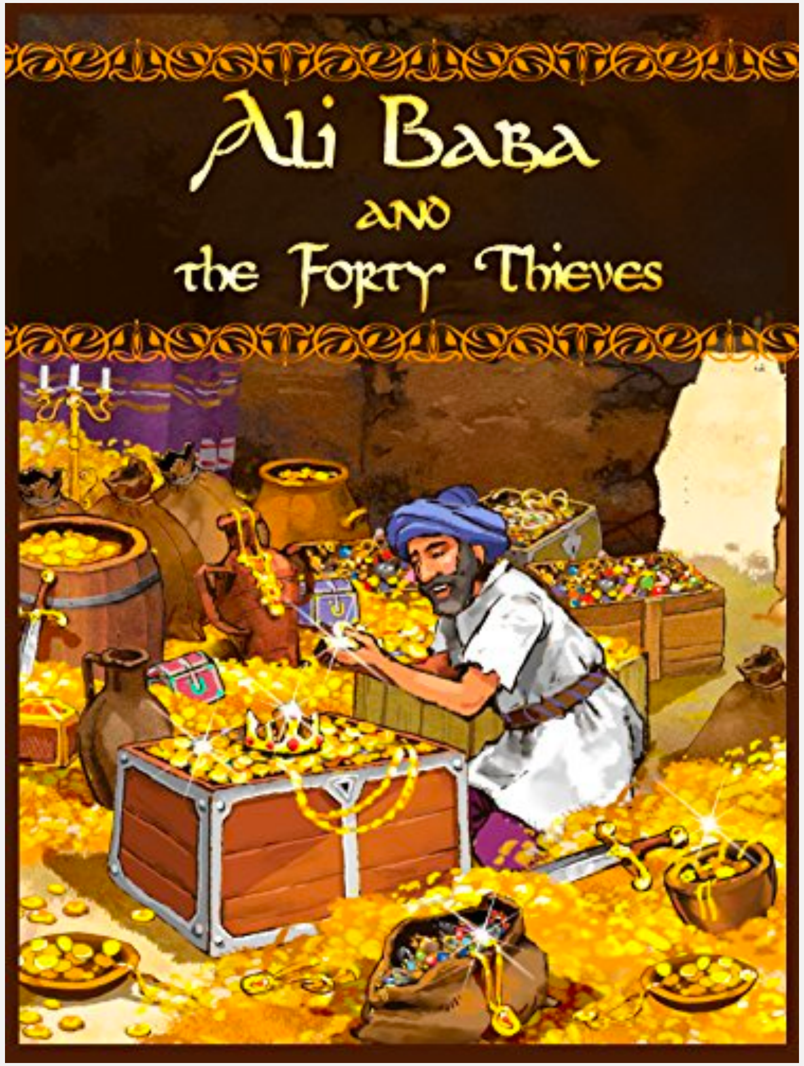
NP-Completeness





https://xkcd.com/287/

Alibaba's knapsack



https://images.app.goo.gl/pwGFyw2pp6Xmx6CB8

Modern Version

Alibaba Group

11.11
天猫全球狂欢节
2019

Practicing reductions

- Circuit−SAT ≤ 3−SAT

+

$$3\text{−SAT} \le_P \text{INDEPENDENT−SET} \le_P \text{VERTEX−COVER} \le_P \text{SET−COVER}$$

- 3−SAT ≤ HAM−CYCLE

⇒ They are all NP-Complete!

Richard M. Karp

REDUCIBILITY AMONG COMBINATORIAL PROBLEMS†

Richard M. Karp

University of California at Berkeley

96

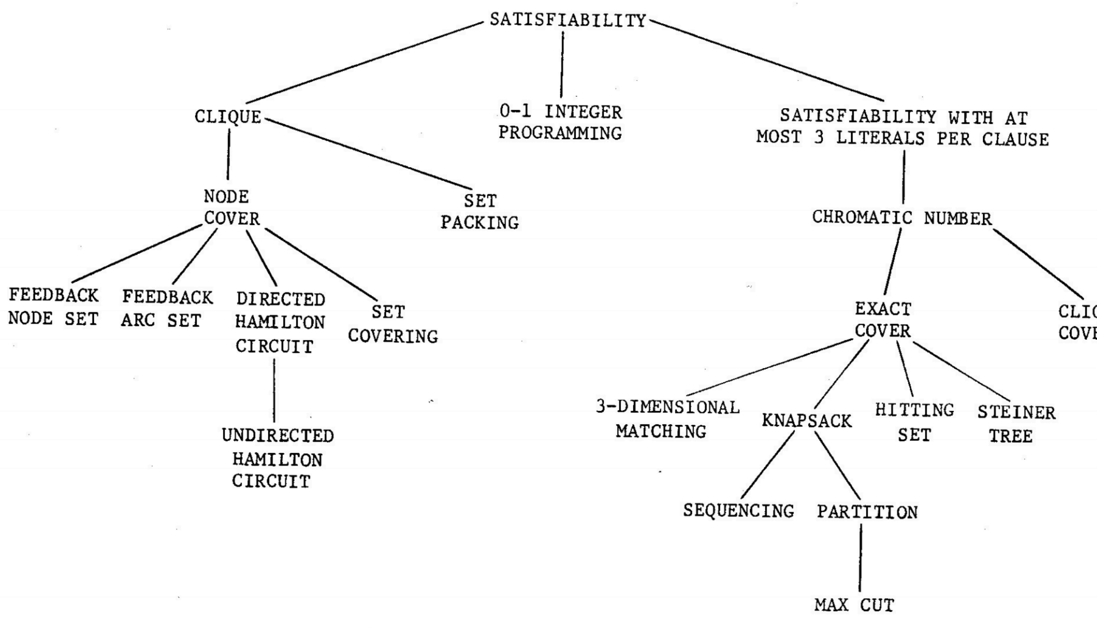


FIGURE 1 - Complete Problems

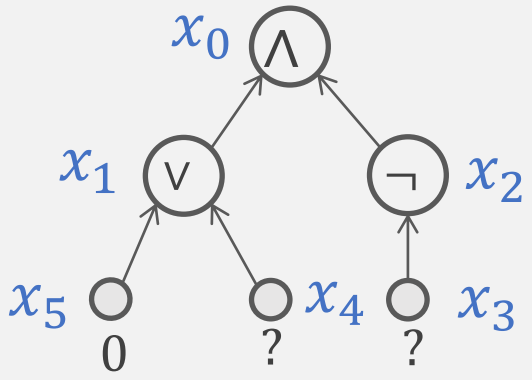
RICHARD M. KARP

$3-\mathbf{SAT}$ is NP-Complete

Theorem. $3-\mathbf{SAT}$ is NP-Complete

Pf. We show $\text{Circuit}-\text{SAT} \leq_P 3-\text{SAT}$

- Given a circuit K, create a $3-\text{SAT}$ variable x_i for each gate
- Make circuit compute correct values at each node



$x_2 = \neg x_3$	$\Rightarrow x_2 \vee x_3, \overline{x_2} \vee \overline{x_3}$
$x_1 = x_4 \vee x_5$	$\Rightarrow x_1 \vee \overline{x_4}, x_1 \vee \overline{x_5}, \overline{x_1} \vee x_4 \vee x_5$
$x_0 = x_1 \wedge x_2$	$\Rightarrow \overline{x_0} \vee x_1, \overline{x_0} \vee x_2, x_0 \vee \overline{x_1} \vee \overline{x_2}$

- Hard-coded input values and output value

$x_5 = 0 \Rightarrow \overline{x_5}$ $\quad$ $x_0 = 1 \Rightarrow x_0$

- Final step: turn clauses into exactly 3 literals by adding dummy variables

EX. $x_1 \vee x_2 \Rightarrow x_1 \vee x_2 \vee y, x_1 \vee x_2 \vee \overline{y}$

Circuit K satisfiable iff. $\exists$ truth assignment satisfying all clauses constructed

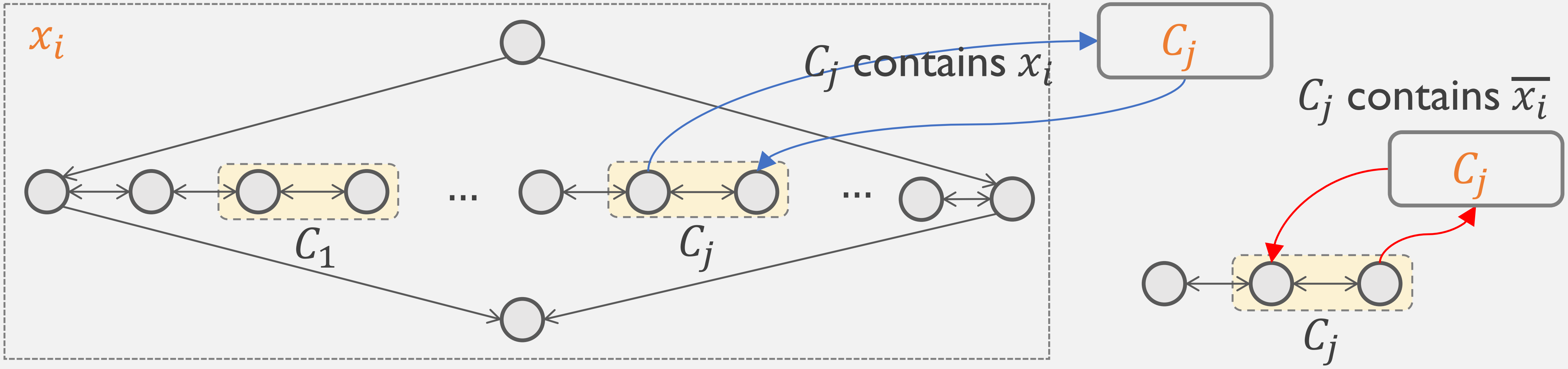
! Don't forget to show $3-\mathbf{SAT} \in \mathbf{NP}$

(DIR−)HAM−CYCLE is NP-Complete

(DIR−)HAM−CYCLE. Given a directed graph $G = (V, E)$, does there exist a directed cycle Γ that visits every node exactly once?

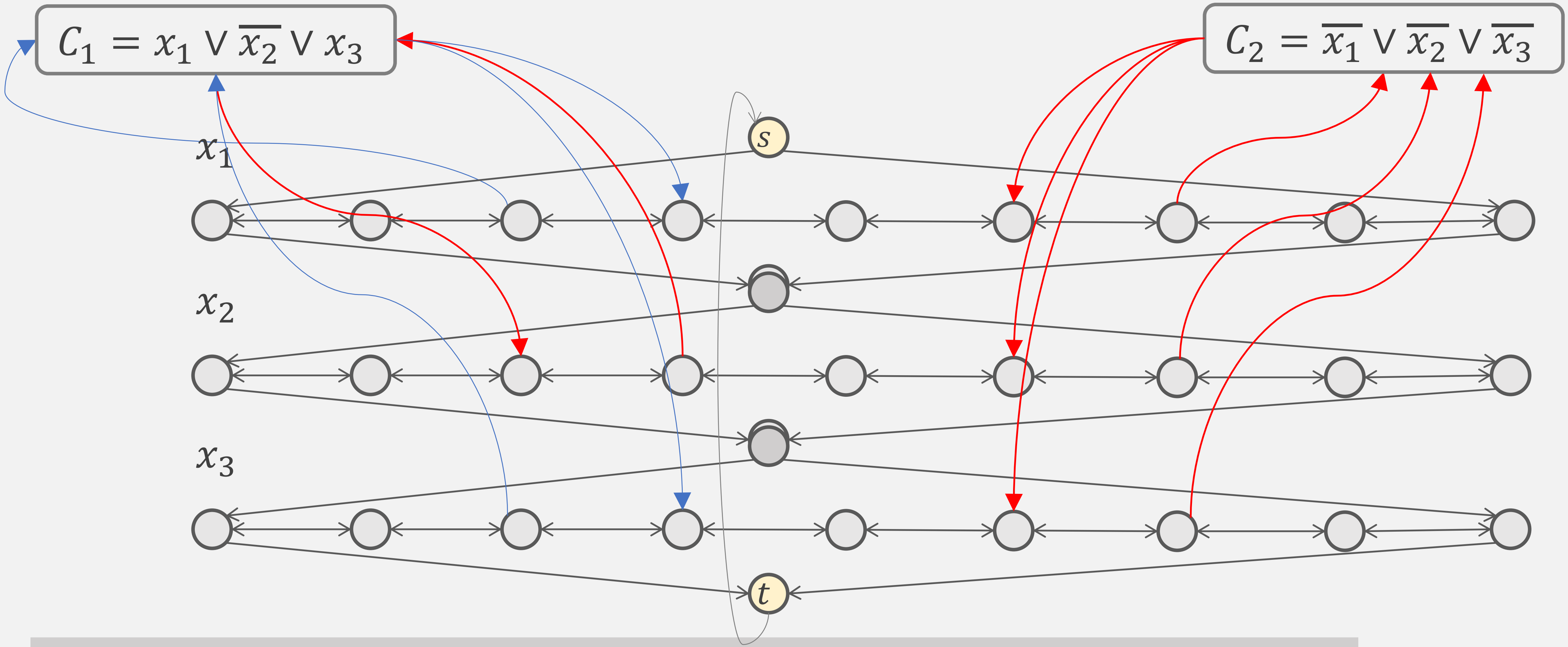
Theorem. $3\text{−SAT} \leq_P \text{(DIR−)HAM−CYCLE}$

Pf. Given 3−SAT instance Φ in CNF: n variables x_i and k clauses C_j



Intuition: traverse row i from left to right $\Leftrightarrow$ set variable x_i = true

$3\text{–SAT} \leq_P (\text{DIR–})\text{HAM–CYCLE}$



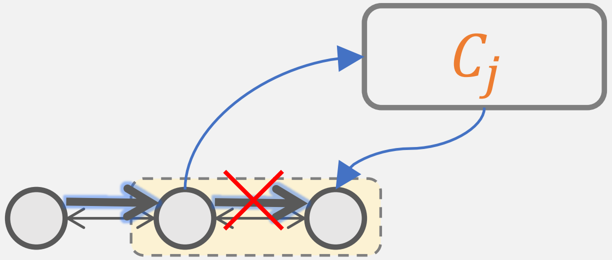
Claim. Φ is satisfiable iff. G has a Hamiltonian cycle

$3\text{−SAT} \le_P \text{(DIR−)HAM−CYCLE}$

Claim. Φ is satisfiable iff. G has a Hamiltonian cycle

(⇒) Suppose Φ has a satisfying assign. $\boldsymbol{x}^*$. Define an H-Cycle in G:

- if $x_i^* = \text{true}$, traverse row x_i from left to right
- if $x_i^* = \text{false}$, traverse row x_i from right to left
- For each clause C_j pick (only) one row i and take a detour



(⇐) Suppose G has a H-Cycle Γ. Define a satisfying assign. in Φ:

- In Γ, replace edges going/leaving C_j with the edge of the corresponding two nodes in some row. This gives a new cycle Γ' in $G - \{C_1, C_2, \dots, C_k\}$
- In Γ', set $x_i = \text{true}$ if Γ' traverses row i left-to-right; set $x_i = \text{false}$ otherwise.