More NP-completeness

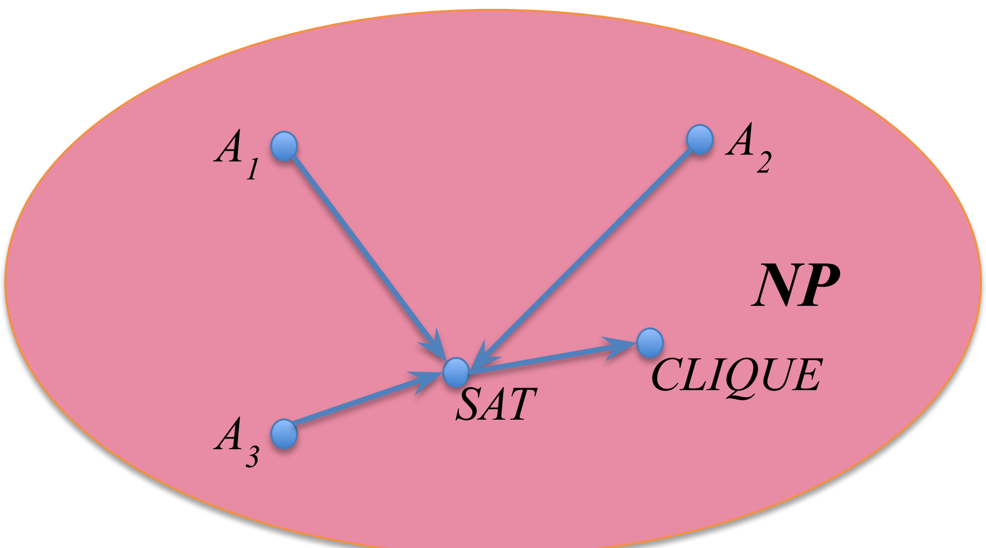
Sipser 7.5 (pages 283-294)

NP's hardest problems

- Definition 7.34:

  A language B is **NP-complete** if

  1. $B \in NP$
  2. $A \leq_p B$, for all $A \in NP$



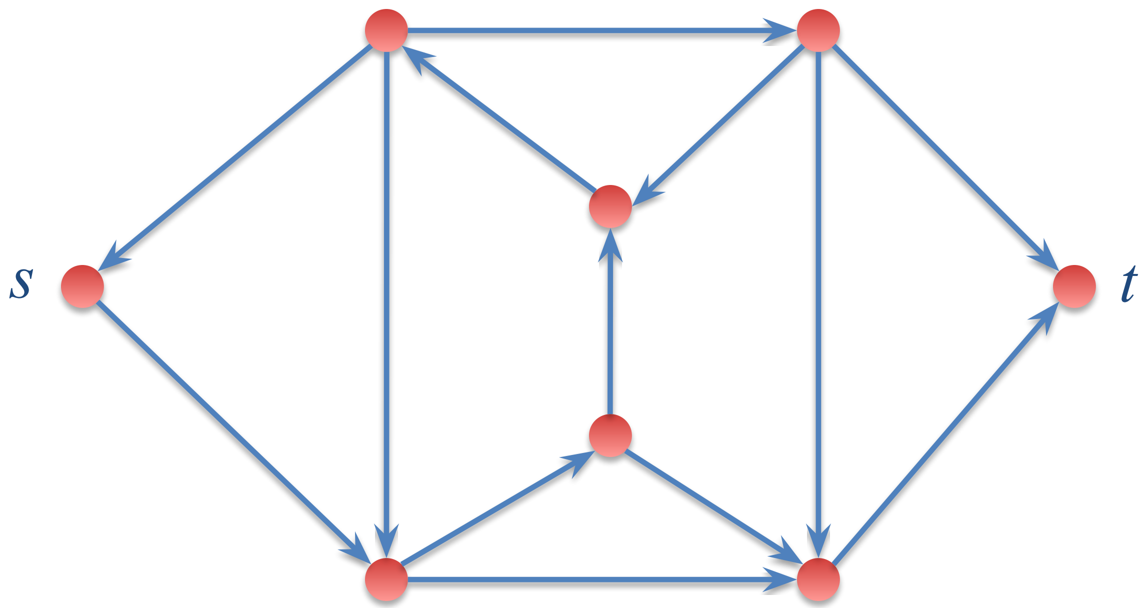
3SAT's main features

- *Choice:*
  Each variable has a choice between two truth values.
- *Consistency:*
  Different occurrences of the same variable have the same value.
- *Constraints:*
  Variable occurrences are organized into clauses that provide constraints that must be satisfied.

$$\phi = (x_1 \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_3 \vee \overline{x_4}) \wedge (x_2 \vee x_4 \vee x_5)$$

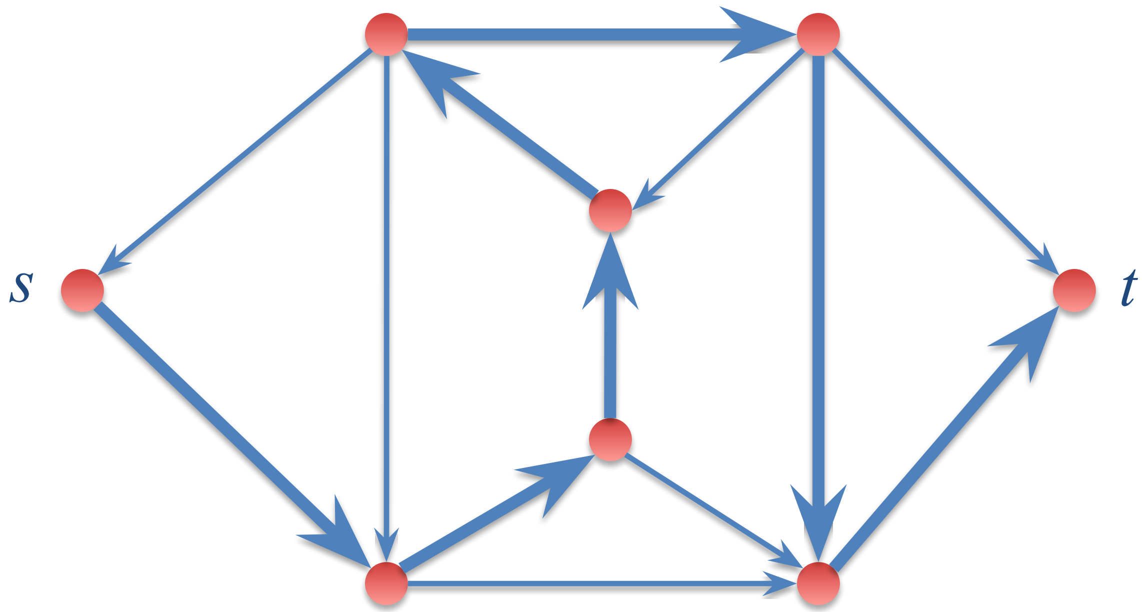
Hamiltonian paths

- *HAMPATH* = {<G,s,t> | ∃ Hamiltonian path from s to t}
- Theorem 7.46: *HAMPATH* is NP-complete.



Hamiltonian paths

- $HAMPATH = \{<G,s,t> \mid \exists \text{ Hamiltonian path from } s \text{ to } t\}$
- Theorem 7.46: *HAMPATH* is NP-complete.



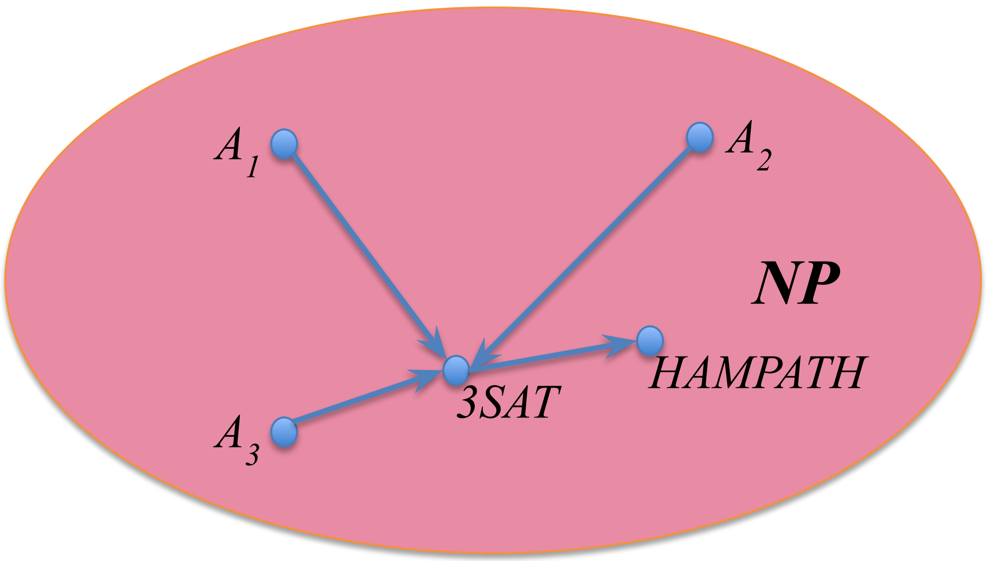
Remember... $HAMPATH \in NP$

N = "On input $<G,s,t>$:

1. Guess an ordering, $p_1, p_2, \ldots, p_n$, of the nodes of G
2. Check whether $s = p_1$ and $t = p_n$
3. For each $i=1$ to $n-1$,
   check whether (p_i, p_{i+1}) is an edge of G.

If any are not, *reject*. Otherwise, *accept*."

$3SAT \leq_p HAMPATH$



Proof outline

- Given a boolean formula φ, we convert it to a directed graph G such that φ has a valid truth assignment iff G has a Hamiltonian graph

$$\phi = (x_1 \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_3 \vee \overline{x_4}) \wedge (x_2 \vee x_4 \vee x_5)$$

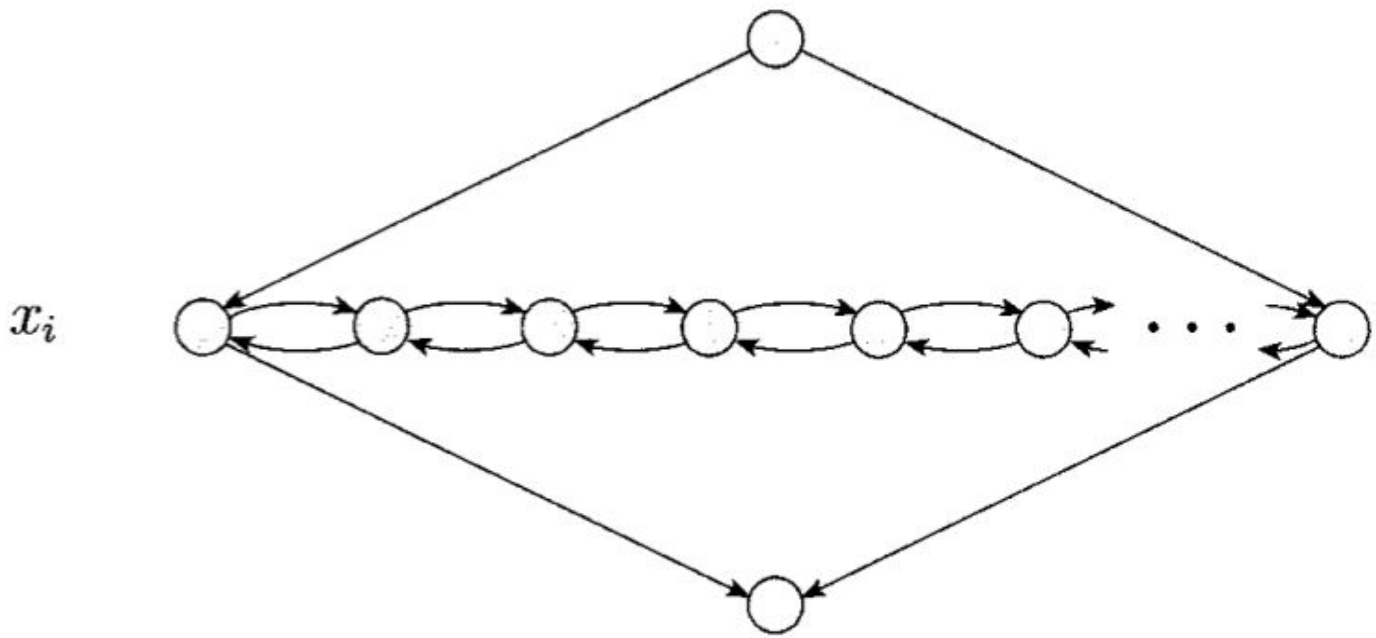
3SAT's main features

- *Choice:*
  Each variable has a choice between two truth values.
- *Consistency:*
  Different occurrences of the same variable have the same value.
- *Constraints:*
  Variable occurrences are organized into clauses that provide constraints that must be satisfied.

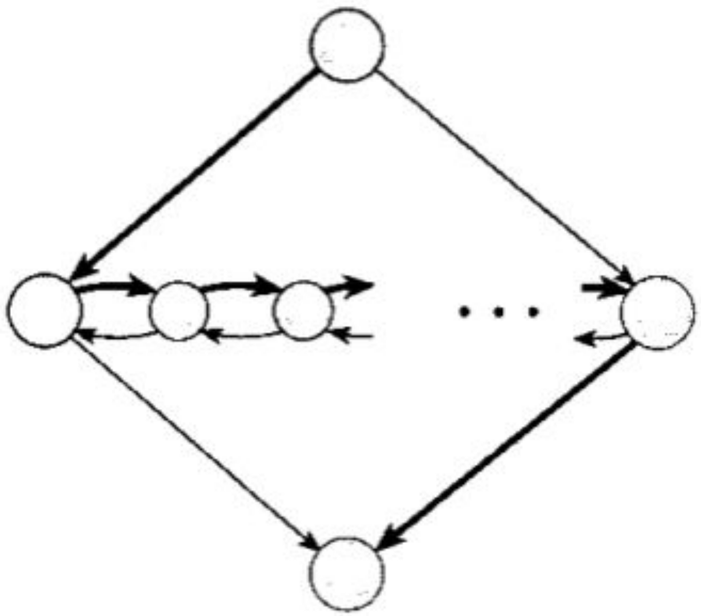
*We model each of these three features by a different a "gadget" in the graph *G*.

The choice gadget

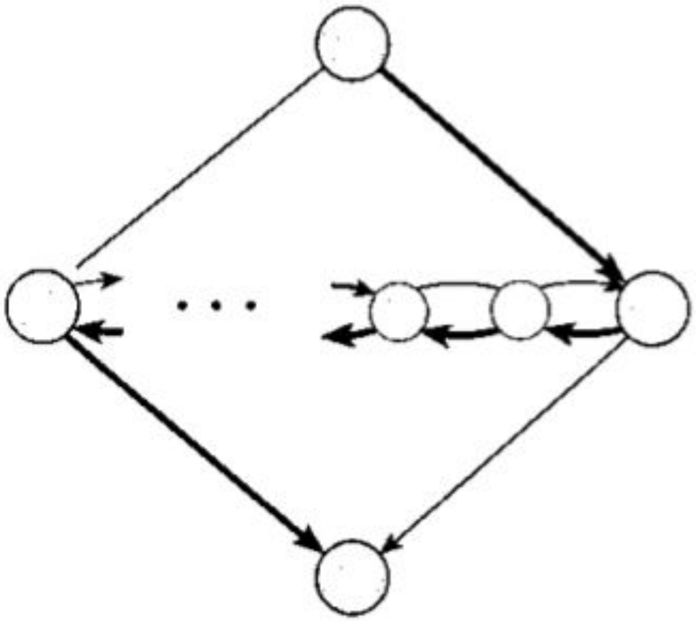
- Modeling variable x_i



Zig-zagging and zag-zigging

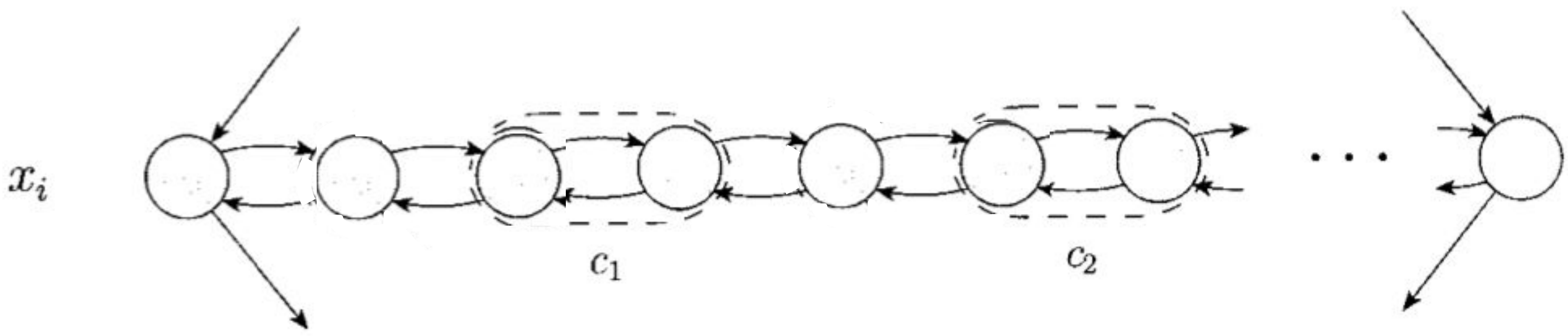


Zig-zag
(TRUE)



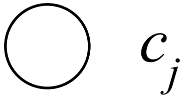
Zag-zig
(FALSE)

The consistency gadget

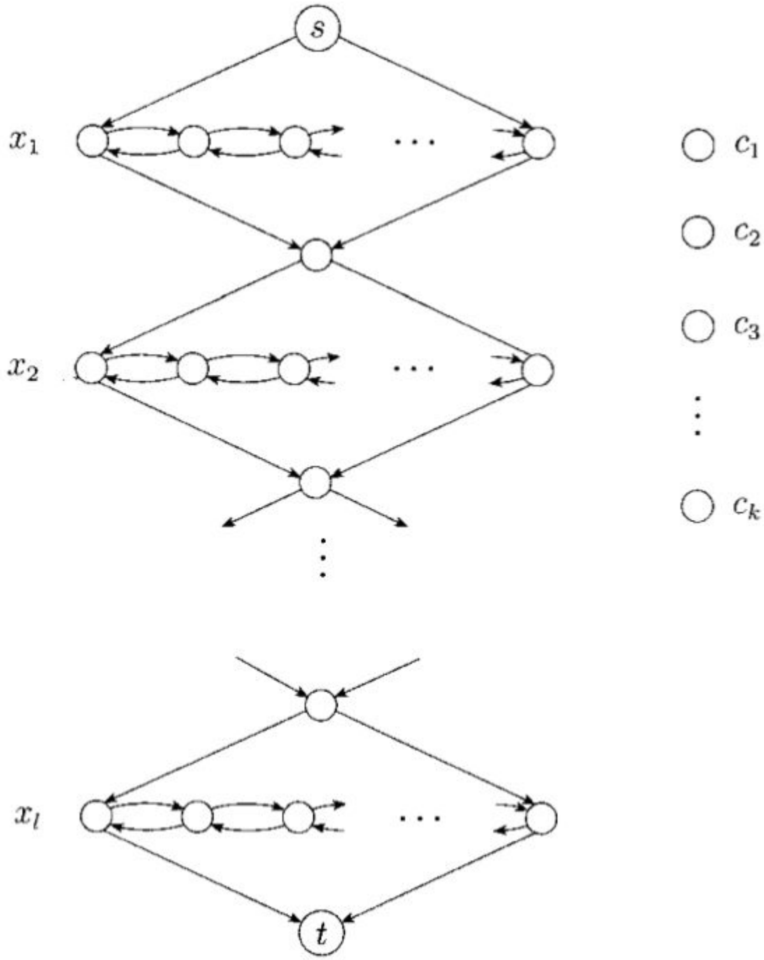


Clauses

- Modeling clause c_j

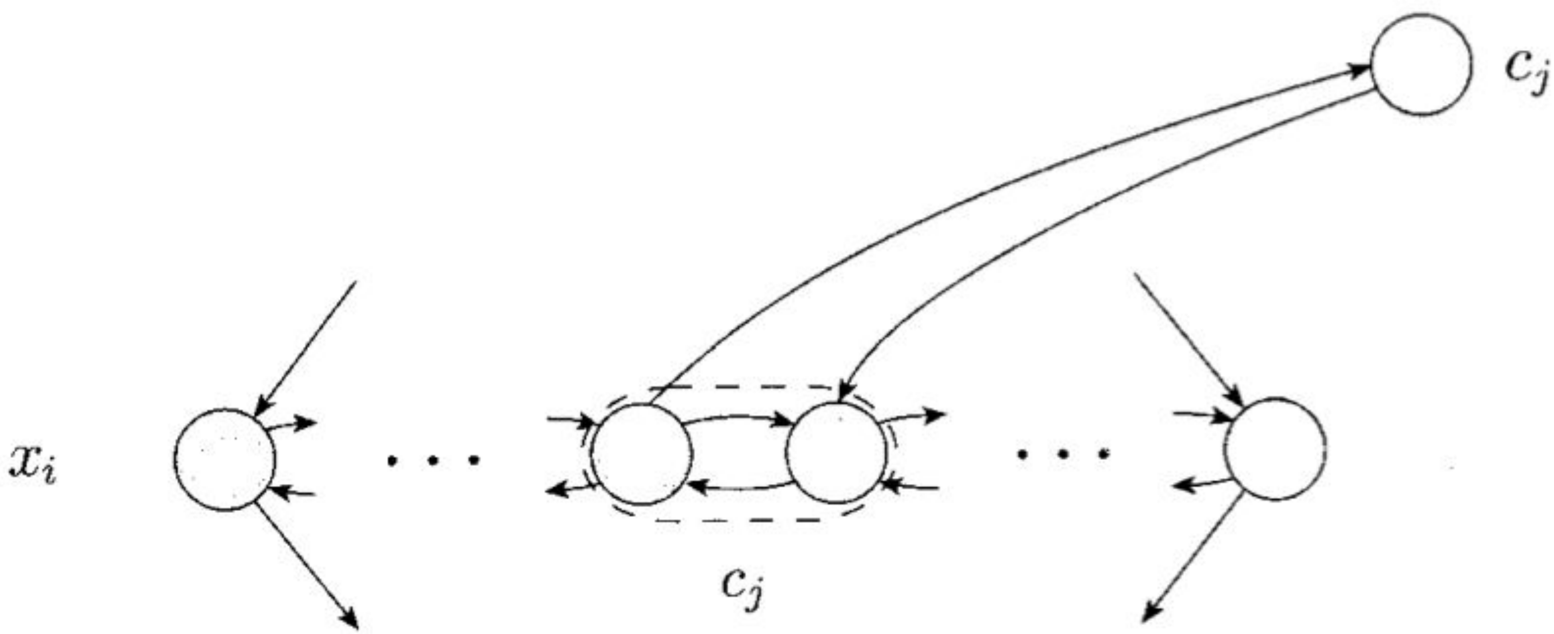


The global structure



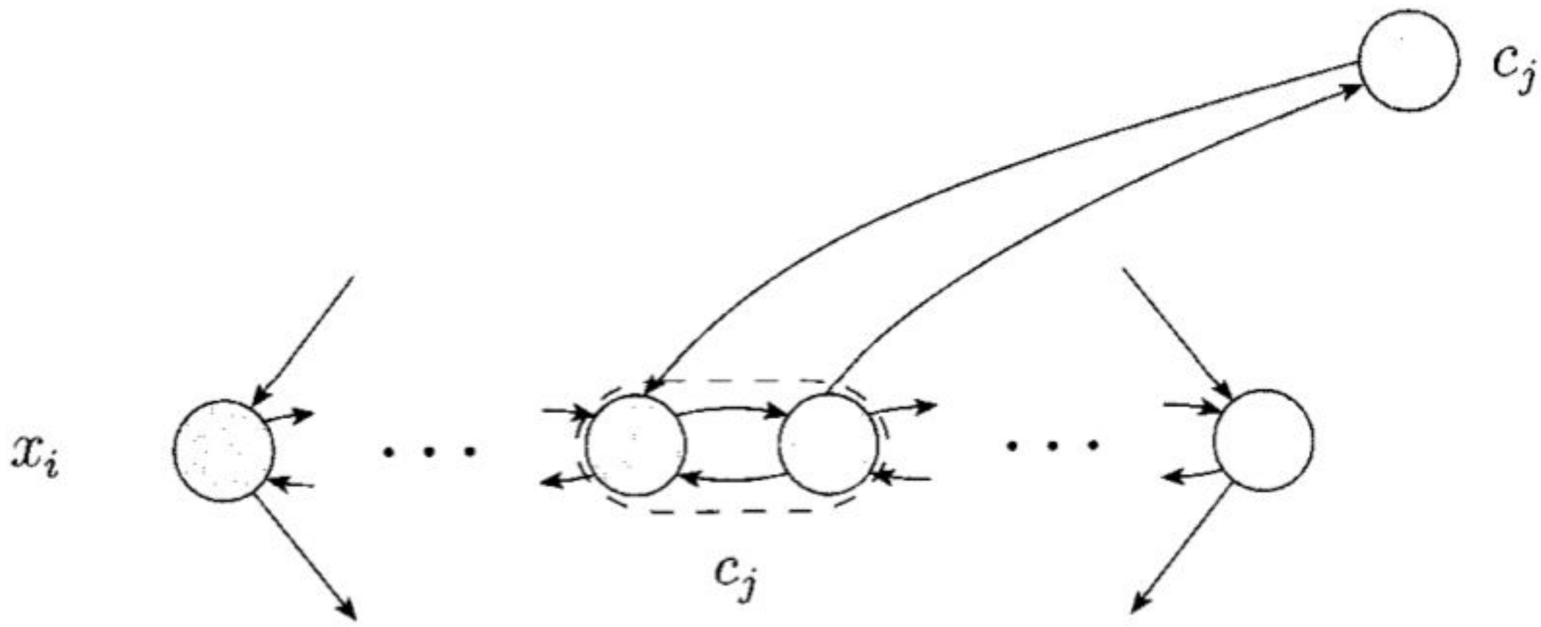
The constraint gadget

- Modeling when clause c_j contains x_i

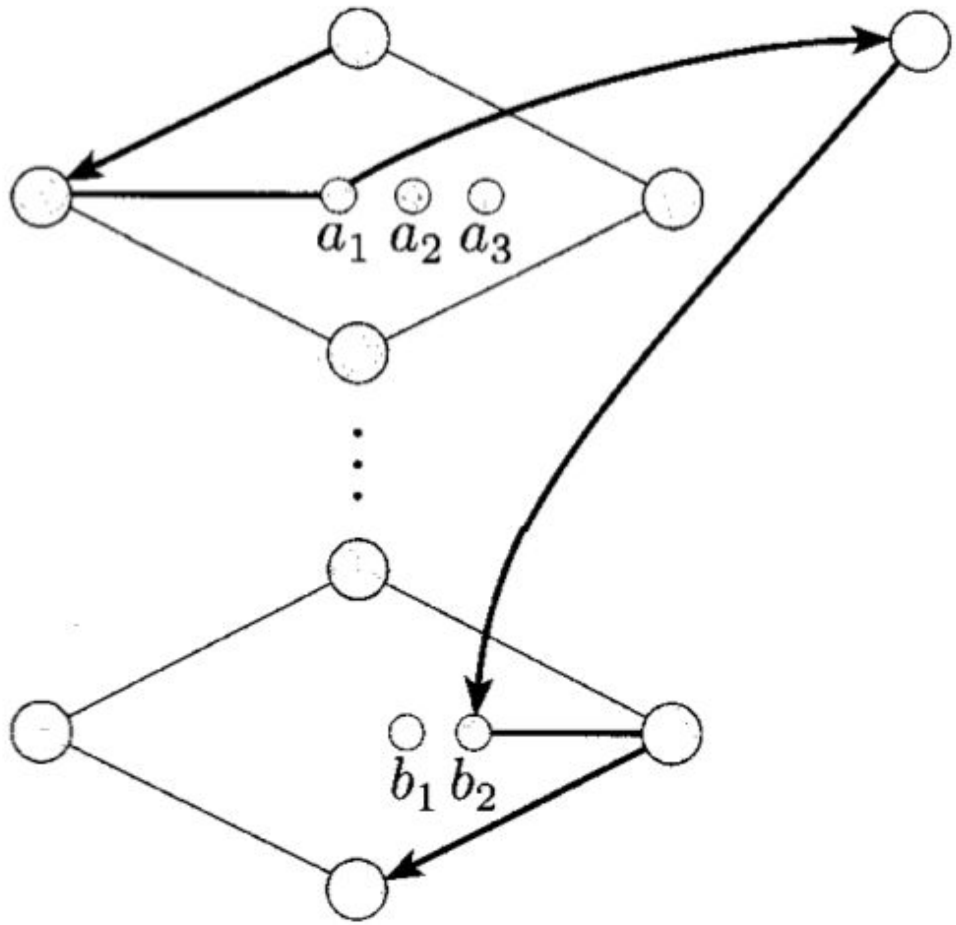


The constraint gadget

- Modeling when clause c_j contains $\overline{x_i}$

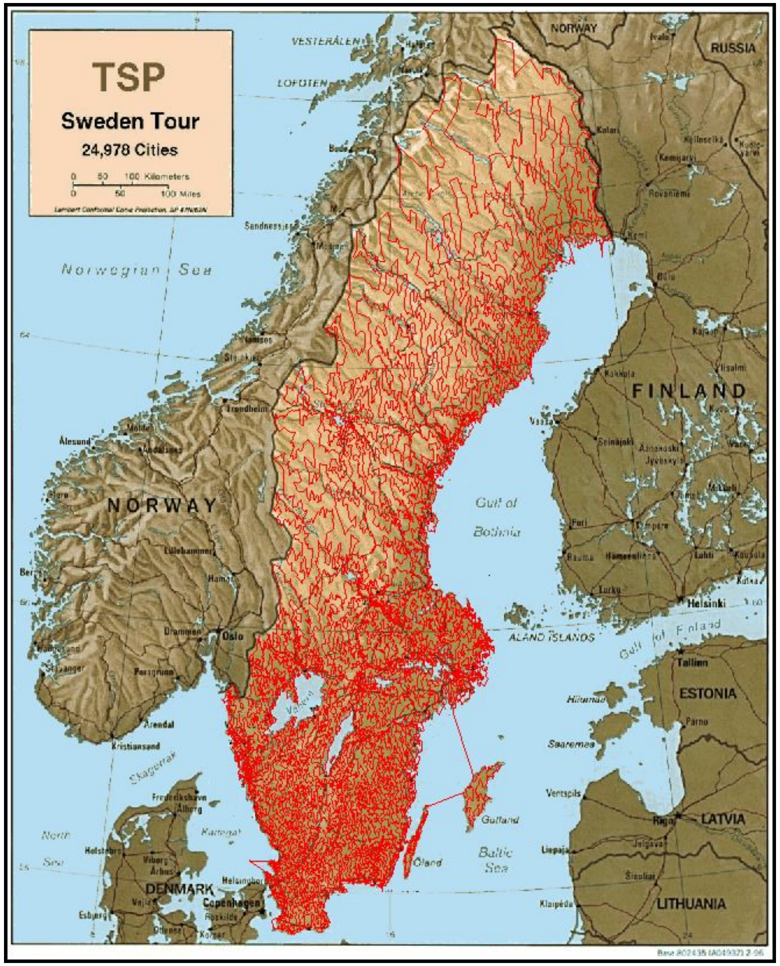


A situation that cannot occur

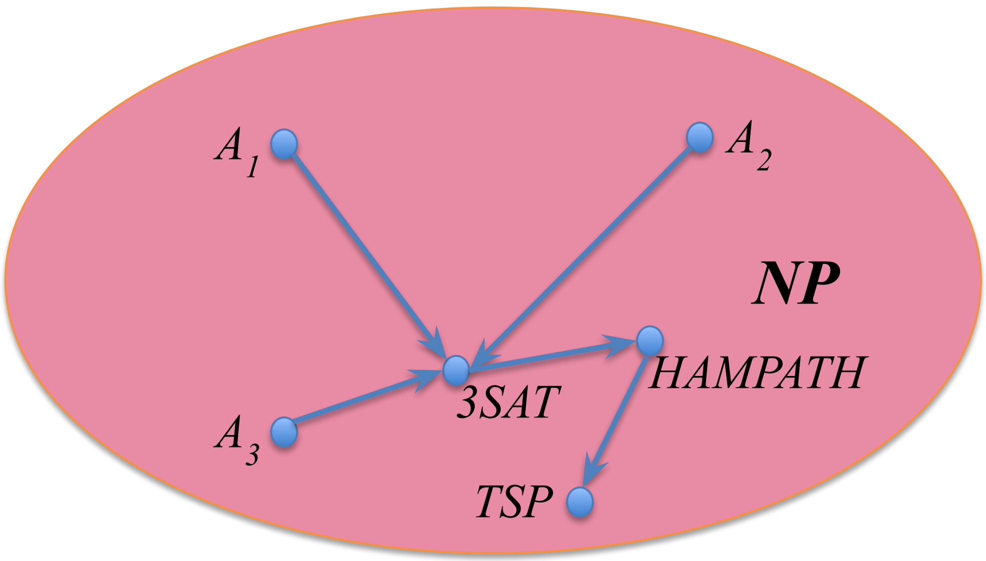


TSP is NP-complete

- *TSP*: Given n cities, $1, 2, \ldots, n$, together with a nonnegative distance d_{ij} between any two cities, find the shortest tour.



$HAMPATH \leq_p TSP$



SUBSET-SUM is NP-complete

- *SUBSET-SUM* =

  $\{\langle S,t\rangle \mid S = \{x_1, \ldots, x_k\}$ *and,*

  for some $\{y_1, \ldots, y_l\} \subseteq S, \Sigma\, y_i = t\}$
- Why is *SUBSET-SUM* in NP?

$3SAT \leq_p SUBSET\text{-}SUM$

$$\phi = (x_1 \vee \overline{x_2} \vee x_3) \wedge (x_2 \vee x_3 \vee \cdots) \wedge \cdots \wedge (\overline{x_3} \vee \cdots \vee \cdots)$$

	1	2	3	4	$\cdots$	l	c_1	c_2	$\cdots$	c_k
y_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
z_1	1	0	0	0	$\cdots$	0	0	0	$\cdots$	0
y_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
z_2		1	0	0	$\cdots$	0	1	0	$\cdots$	0
y_3			1	0	$\cdots$	0	1	1	$\cdots$	0
z_3			1	0	$\cdots$	0	0	0	$\cdots$	1
$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
y_l						1	0	0	$\cdots$	0
z_l						1	0	0	$\cdots$	0
g_1							1	0	$\cdots$	0
h_1							1	0	$\cdots$	0
g_2								1	$\cdots$	0
h_2								1	$\cdots$	0
$\vdots$									$\ddots$	$\vdots$
g_k										1
h_k										1
t	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3