CMSC 451 - Algorithm Design
Lecture 19 - NP-Completeness: Hamiltonian Cycle

Directed Hamiltonian Cycle: Given a directed graph $G=(V,E)$, is there a simple* cycle that visits all the vertices of G?

*Recall - simple cycle does not repeat any vertices (except first = last)

Formally:

$$DHC = \{ G \mid G \text{ has a Hamiltonian cycle} \}$$

Named for 19^{th} century Irish astronomer, physicist, mathematician William Rowan Hamilton.

NP-Complete Problems:
- Directed/Undirected Hamil. Cycles/Paths
- Traveling Salesman Problem
- Longest (simple) path

Note: Eulerian circuit/path problem
(Cycle/path traversing all edges exactly once) is in P. (DFS)

Component Design:

- So far our reductions have been fairly simple
- This one is more complicated
- Build components (or gadgets) to enforce problem constraints.

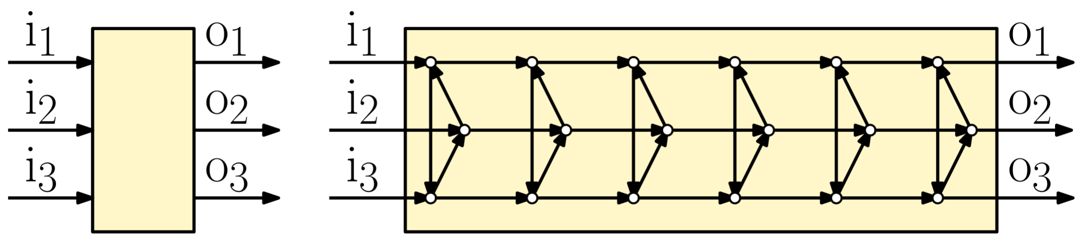
DHC Gadget:

- We'll reduce 3SAT to DHC
- 3SAT - Formula in 3-CNF
  - Conjunction ("and") of...
  - Clauses, each is disjunction ("or") of...
  - Three literals, each is a variable x_j or its complement $\bar{x}_j$

$$F = (\bar{x}_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \bar{x}_2 \vee \bar{x}_3) \wedge (x_2 \vee \bar{x}_1 \vee \bar{x}_3) \wedge (x_1 \vee x_3 \vee \bar{x}_2)$$

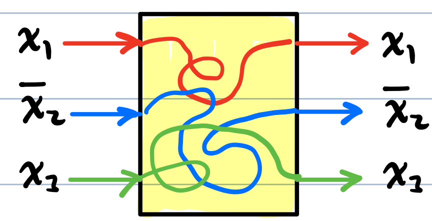
- To satisfy, each clause must have at least one true literal

- For each clause, we create an 18 vertex subgraph, called DHC Gadget.



- We'll create a gadget for each clause

$(x_1 \vee \bar{x}_2 \vee \bar{x}_3) \rightarrow$



- Main property - Can route 1, 2, or 3 paths
  - all vertices are visited
  - paths exit at same position they entered

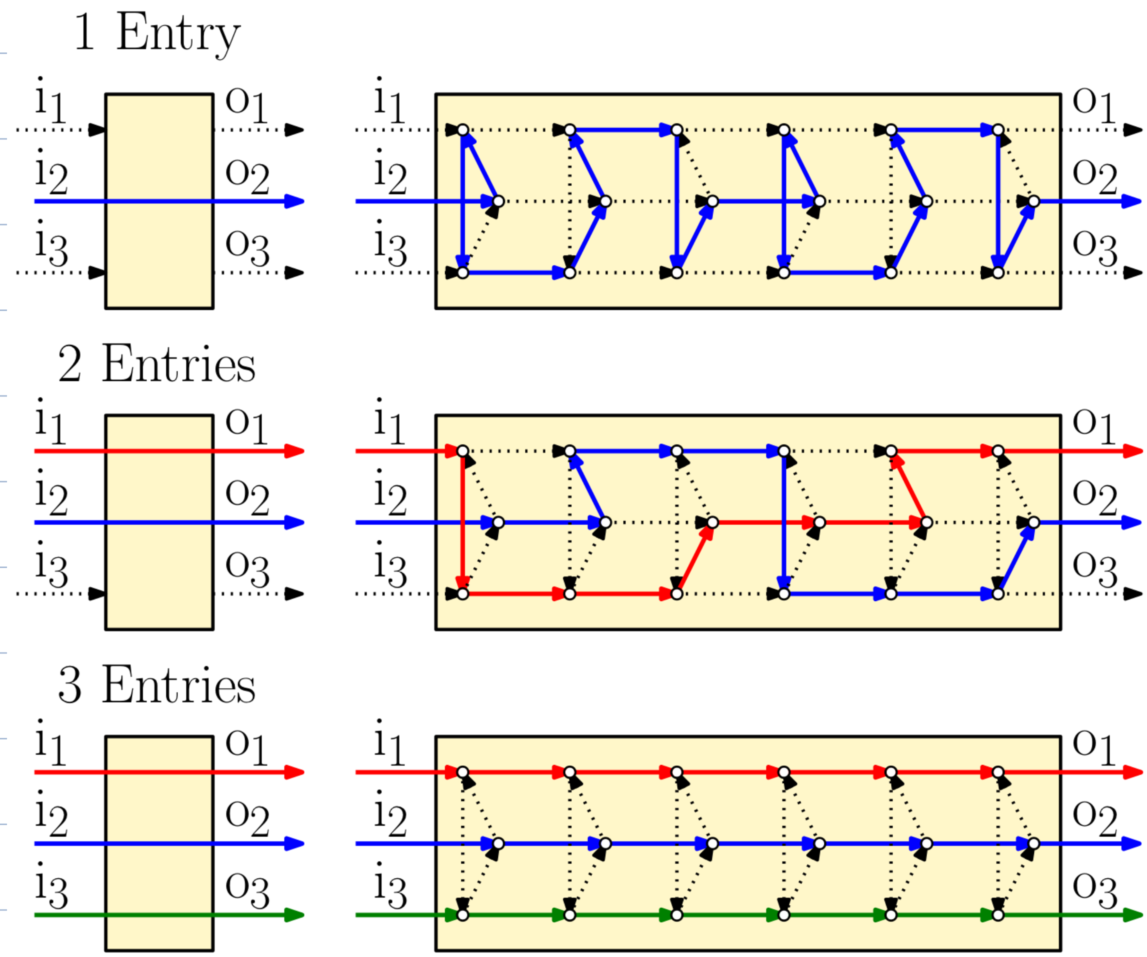
Lemma: (DHC Gadget Properties)

(i) Given any nonempty set of k entry edges, there exist k vertex-disjoint paths that visit all vertices + emerge on same exit edge.

(ii) Given any nonempty set of k vertex-disjoint paths from entry to exit that hit all vertices, they must exit at the same position they entered.

Proof:

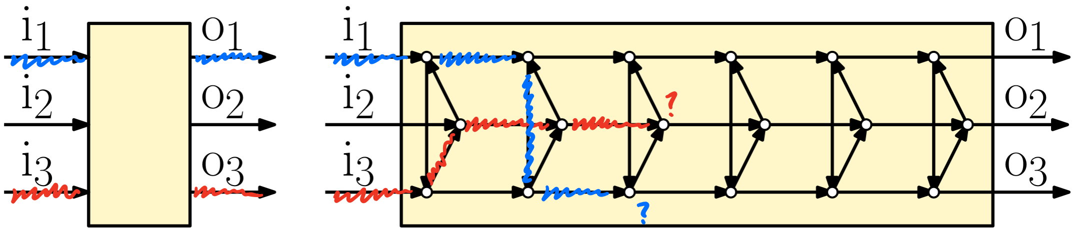
(i) "Proof by picture"



- All vertices visited
- If path enters on edge i_j it exits on o_j

(ii) Trickier. Start with any entry edge combination + see how to get the paths through without missing any vertices.

- 1 entry - Easy - Just cycle around
- 3 entries - Easy - Just go straight through
- 2 entries - (Exercise) $\square$

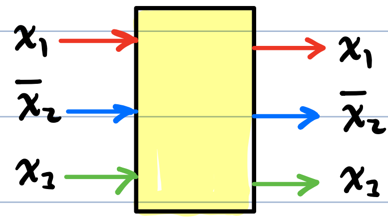


Try to finish these paths without missing or repeating. Only one way!

The Reduction: $3SAT \leq_p DHC$

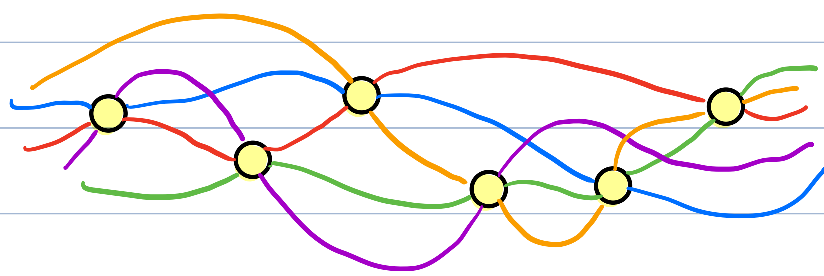
- Map each clause to a DHC gadget

$(x_1 \vee \bar{x}_2 \vee \bar{x}_3) \rightarrow$

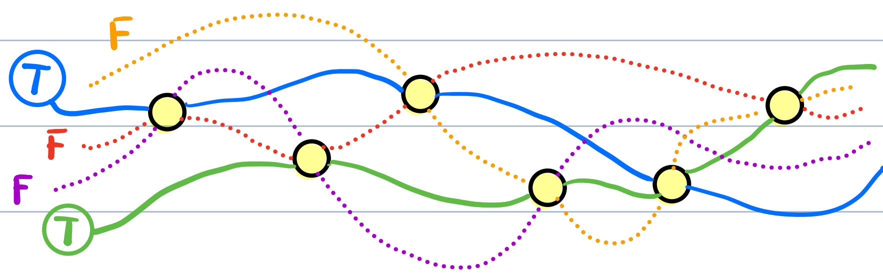


- Think of gadgets as bulbs on string of lights

  - Each bulb has 3 wires through it
  - Wires correspond to literals x_j or $\bar{x}_j$



  - Wires can be "live" (true) or "dead" (false)



- Want the whole string to light up

At least one live wire through each bulb

$\equiv$ At least one true literal for each clause

- Logical consistency:

For any variable, only one literal can be live

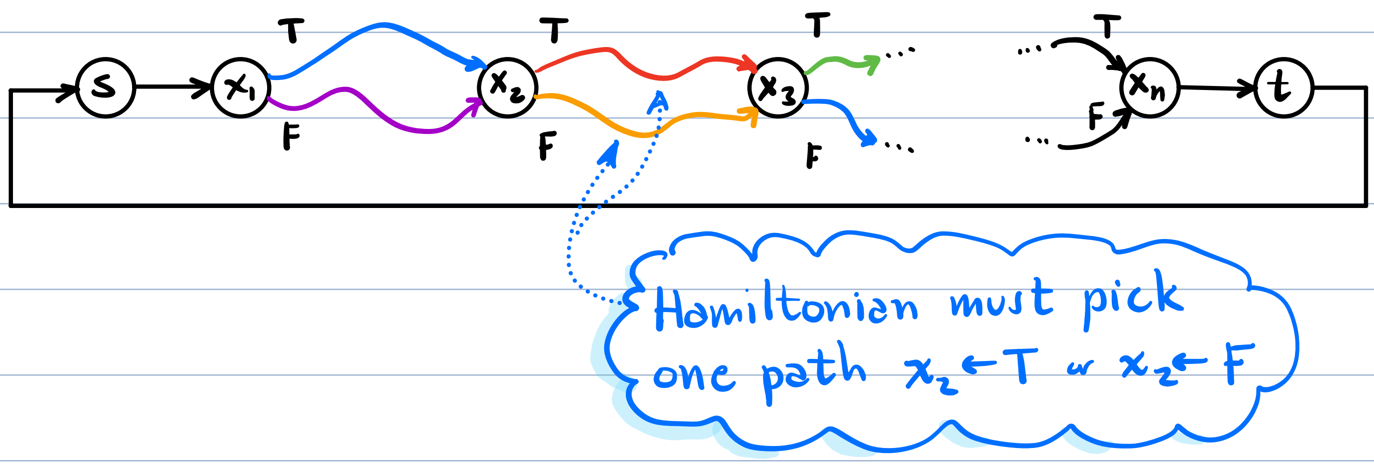
x_j ～ or x_j ⋯

$\bar{x}_j$ ⋯ $\bar{x}_j$ ～

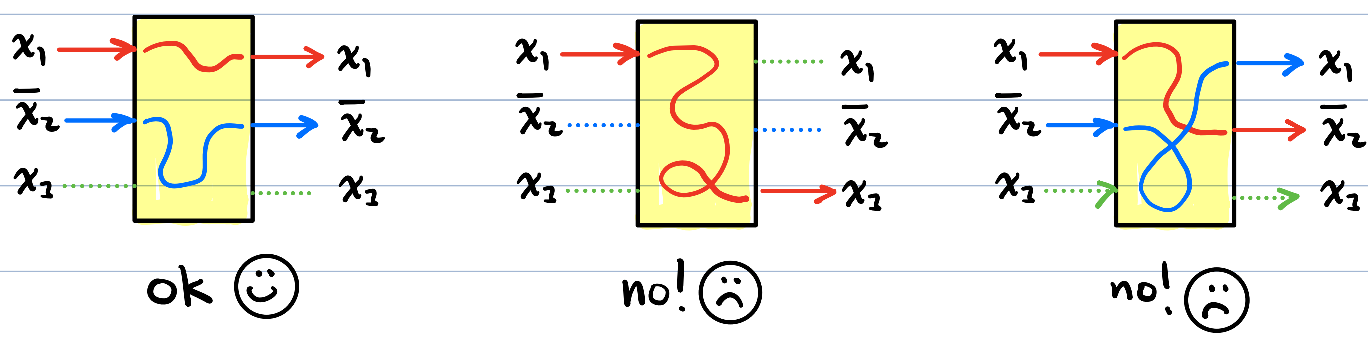
$x_j \leftarrow$ true

$x_j \leftarrow$ false

- To make a cycle, we daisy-chain these paths together:

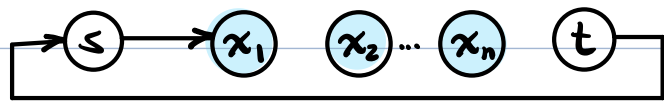


- Gadget designed to keep wires from crossing



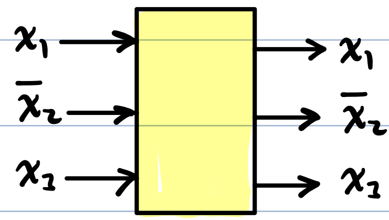
Reduction: Given formula F in 3-CNF form:

- Create **vertices**:
  - s – start
  - t – end
  - $x_1, x_2, \dots, x_n$ – variable vertices
- **Edges:** (s, x_1) (t, s)

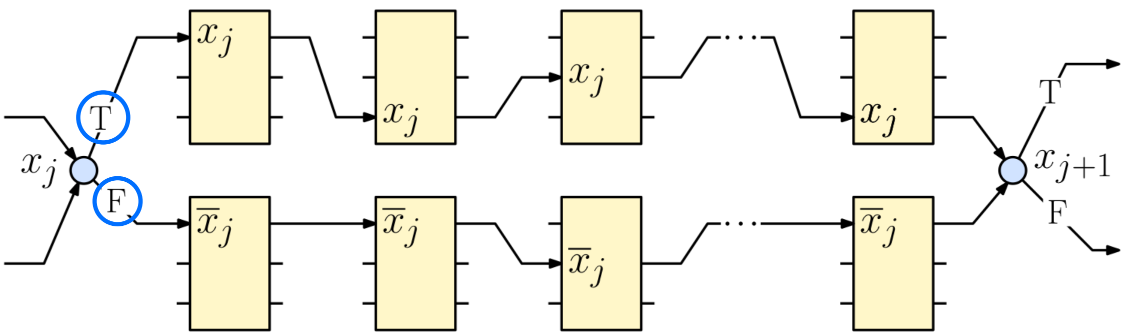


- For each **clause**, create a **gadget**

$(x_1 \vee \bar{x}_2 \vee \bar{x}_3) \rightarrow$



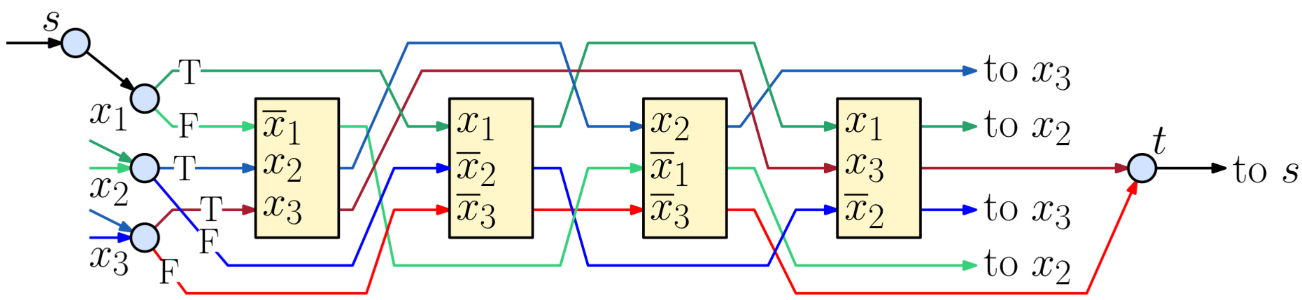
- For each **variable vertex** x_j create **two paths**, T + F.
  - T – goes through all gadgets with x_j
  - F – " " " " " $\bar{x}_j$



- Both end at x_{j+1} (or t, if $j = n$)
- Call the graph G

Example:

$F = (\overline{x}_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x}_2 \vee \overline{x}_3) \wedge (x_2 \vee \overline{x}_1 \vee \overline{x}_3) \wedge (x_1 \vee x_3 \vee \overline{x}_2)$



Running time - Linear in size of F

Note: Reduction does not know/care which variables are true or false.

Correctness:

Lemma: F is satisfiable iff G has a Hamil. cycle

Proof:

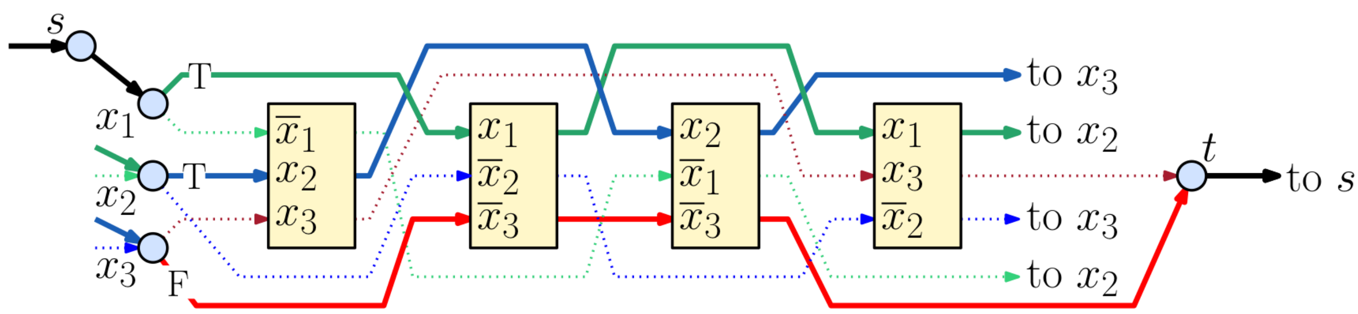
($\Rightarrow$) Suppose that F has a satisfying assignment. We'll present a Hamil. cycle in G:

- Start at (s) and go to (x_1)

- For each x_j, if
  - x_j = true – follow T path out of x_j
  - x_j = false – follow F path out of x_j

- On arriving at t return to s

Example: $x_1 = T$, $x_2 = T$, $x_3 = F$

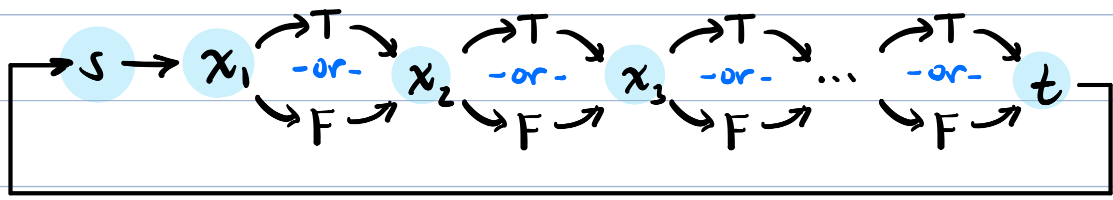


We assert this is a Hamil. cycle.

- Satisfying assignment ⇒
  - each clause has at least one true literal
  - each gadget visited by at least one path
- By DHC Property (i):
  - if at least one path enters
    - all vertices visited once
    - each path exits at same position it enters (wires never cross)

⇒ Hamiltonian cycle in G ✓

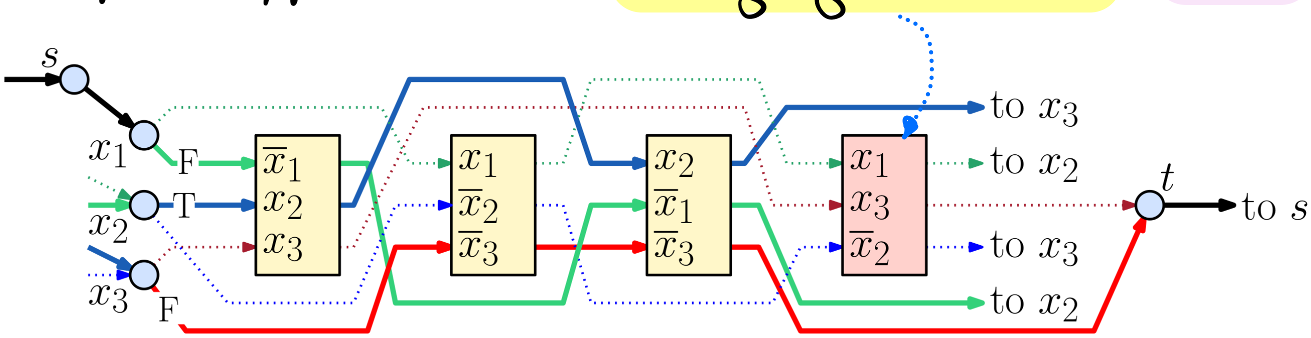
($\Leftarrow$) Suppose G has a Hamiltonian cycle.
It must have following structure



- For each variable x_j
  - if it takes the T path set $x_j \leftarrow$ true
  - " " " " F " " $x_j \leftarrow$ false
- Assert that this is satisfying assignment
- Since this is Hamiltonian, every gadget is visited
- By DHC Gadget property (ii) entry pattern = exit pattern (wires never cross)

$\Rightarrow$ Every clause has at least one true literal

Example: Suppose not. Then some gadget is missed $\rightarrow$ not HC



$\Rightarrow$ F is satisfiable ✓

$\square$

Main result:

Theorem: DHC is NP-complete

Proof: Suffices to show:

$DHC \in NP$: Certificate is a list of vertices
- check that each vertex appears exactly once
- check that every consecutive pair (including last + first) are adjacent
- if so - verification accepts, else rejects

$3SAT \leq_p DHC$:
- From previous lemma.

$\square$