Proving NP- Completeness

Theorem 1.: 3SAT is NP-complete

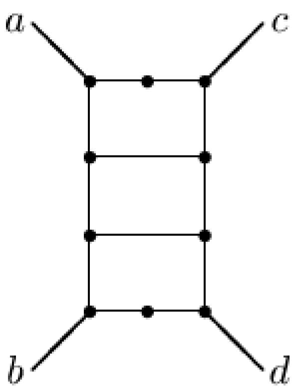
3SAT is the restriction of **SAT** to the case where every clause includes *exactly* three variables. (This regular structure makes it easier to transform than **SAT**.)

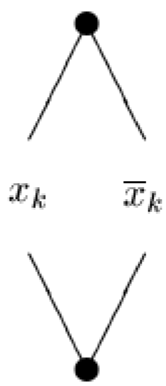
- **3SAT** is in **NP** [obvious, if only because it is a subset of **SAT**].
- Given an arbitrary instance of **SAT**, we show how to transform it into an instance of **3SAT**. Look in turn at each clause of the instance of **SAT**:
- If it has *fewer* than three variables, repeat a variable as necessary; *e.g.* (*a* **or not** *b*) => (*a* **or** *a* **or not** *b*).
- If it has *more* than three variables, introduce a new variable, and use it to split the clause into one with three variables and another with one fewer variables [which can be reduced the same way]: *e.g.* (*a* **or not** *b* **or** *c* **or not** *d* **or not** *e*) => (*a* **or not** *b* **or** *z*) **and** (**not** *z* **or** *c* **or not** *d* **or not** *e*).
- The resulting formula can clearly be produced in time proportional to the size of the instance, and is clearly satisfiable if and only if the original formula was. So
- **SAT** $\leq_p$ **3SAT**. Since **SAT** is NP-complete, so must be **3SAT**. QED.

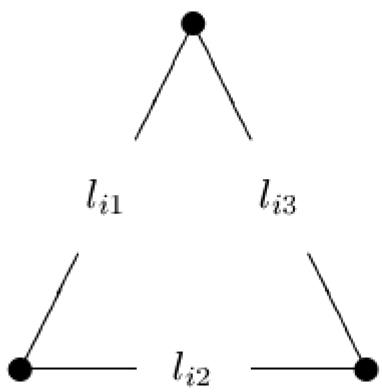
Theorem 2.: Hamilton cycle is NP-complete
Proof: The fact that it is in NP is trivial. To see that it is NP-complete, we'll reduce from 3-SAT. Let: $F = \wedge_i (l_{i1} \vee l_{i2} \vee l_{i3})$ be an instance of 3-SAT over variables $x_1 \ldots x_n$.

Now we wish to create a graph G such that G has a Hamilton cycle if and only if F is satisfiable.
IDEA: construct an exclusive-or gadget. To strip the nodes of the gadget, a Hamilton cycle must do exactly one of the following:
1. enter and leave through a and b, or
2. enter and leave through c and d

We will use the following gadgets:







The gadget on the left is this exclusive-or gadget.
Now, the reduction is as follows:

- For each variable x_k have two nodes as shown in the middle diagram. (Call these two nodes supernodes to distinguish them from the nodes in the exclusive-or gadgets.)
- Similarly, each clause has three supernodes with three paths as in the right diagram. All of the supernodes are connected in all possible ways by edges (These edges are not shown).
- For the xor gadgets:
  - If the i^{th} clause has literal $l_{ij} = x_k$ (or $\neg x_k$) then attach an exclusive-or gadget. Nodes *a* and *b* are attached to the two variable supernodes, and nodes *c* and *d* are attached to the two clause supernodes for l_{ij}.
  - Lastly, if a literal x_k (or $\neg x_k$) occurs in more than one clause, the xor-gadgets are attached in series; *b* of one gadget is tied to *a* of the next.

The reduction is clearly polynomial time.

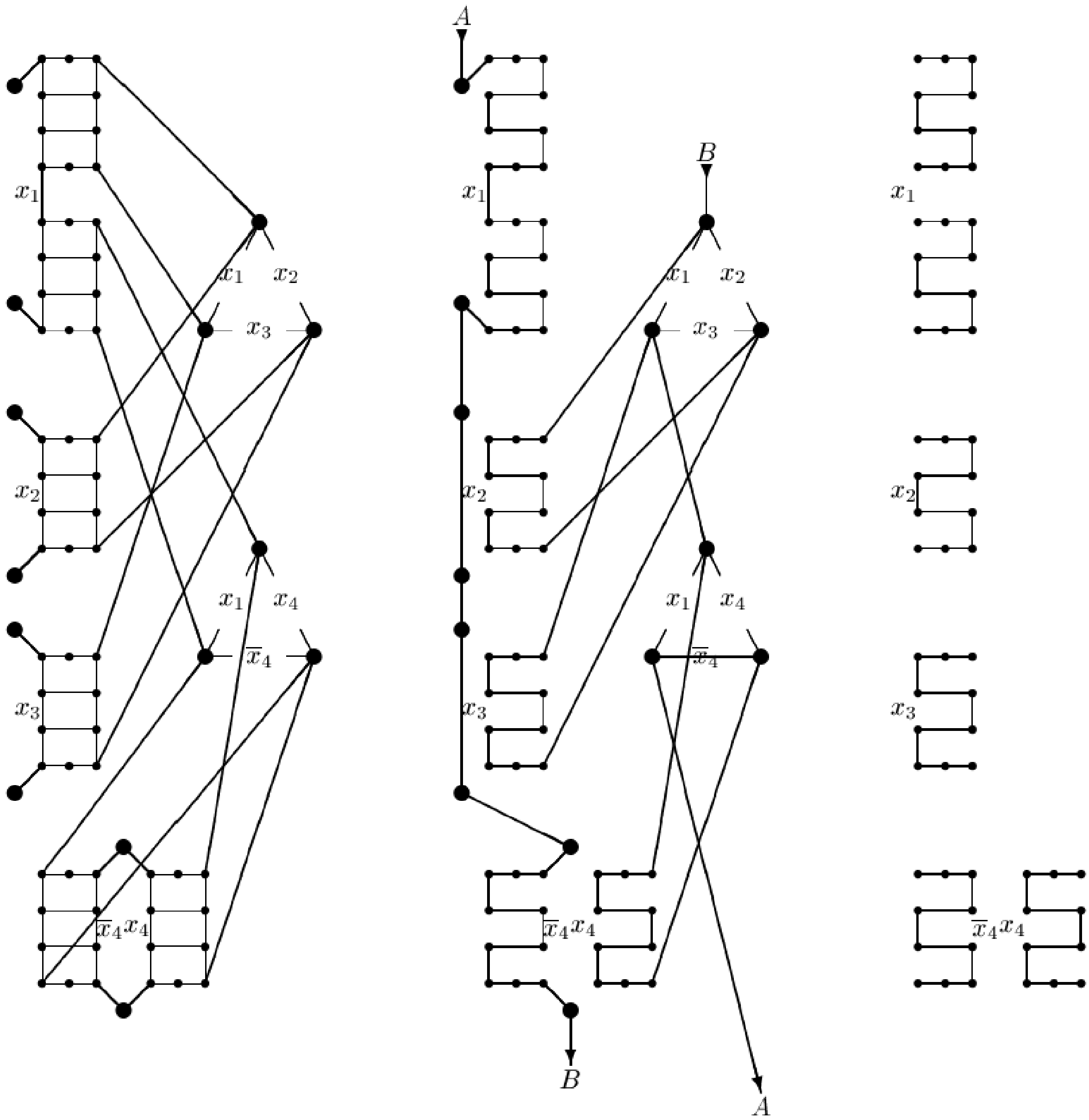
Example:

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

1. The graph on the left represents the reduction, where all supernodes are connected in a clique.

2. The middle shows a Hamilton cycle representing the satifying assignment x_1=T and x_2= x_3= x_4=F.

3. On the right is shown how to extract an assignment from a Hamilton cycle. Just look at which side of each xor-gadget is stripped. Here x_1=x_2=T, x_3=F, and x_4 and $\neg x_4$. are both False. That is, if x_4 were set to either true or false, the formula would still be satisfied.



Theorem 3.:TheTSP is NP-complete.
Proof.:
1.TSP is NP!
2. HAM-CYCLE $\leq_p$ TSP (we give a graph, and a bound: B)
Let $G = (V;E)$ be an instance of HAM-CYCLE. For the complete graph $G' = (V;E')$ and two vertices $i;j \in V$, we may define
$c(i; j) = 0$ if $\{i;j\} \in E$
$c(i; j) = 1$ if $\{i;j\} \notin E$
Clearly, G has a hamiltonian cycle if and only if G' has a tour of cost at most 0.