CPSC 320: Intermediate Algorithm Design and Analysis

Tutorial: Week 13 (NP-completeness worksheet)

Author: Susanne Bradley

Today we're going to go through a step-by-step process for proving a problem is NP-complete. The following problem statement is taken from the CPSC 320 2016W1 offering, written by Steve Wolfman.

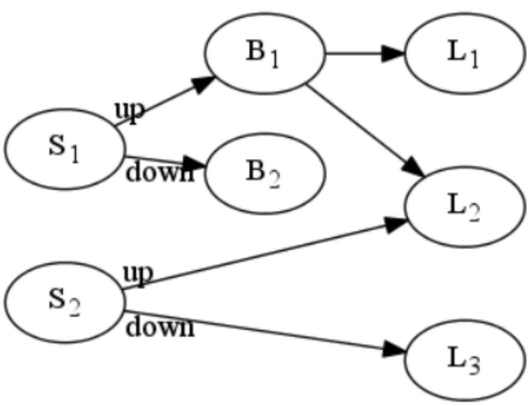
Transformers

In the ELEC problem, you're given a network of electrical wires which can be represented as a directed, acyclic graph (DAG) with three types of nodes:

- "Switch" nodes supply power. They have **no** wires coming in and two wires going out labeled "up" and "down". They also have a switch. If the switch is in the up position, then power (electricity) flows into the up wire. If the switch is in the down position, then power flows into the down wire.
- "Branch" nodes can have one wire coming in (which may or may not carry power) and any number of wires going out. If the wire coming in carries power, then all wires going out also carry power. Otherwise, none of the wires carries power.
- "Load" nodes represent electrical devices that must be powered. They have one or more wires coming in and none going out. If any wire coming in carries power, the load is powered. Otherwise, it is not.

The solution to an ELEC instance is YES if some configuration of the switches powers all the loads; otherwise, it's NO.

1. Indicate a configuration of the switches in the following network that powers all the loads by writing "up" or "down" on each switch node. (Switch nodes are labeled S, branch nodes B, and load nodes L.)



Proving ELEC is in NP

Complete the following proof that ELEC is in NP (by filling in the parts following the "..."):

A good *certificate* for ELEC is...

We can check this certificate in polynomial time by...

Proving ELEC is in NP-hard

We need to find an NP-hard problem to reduce to ELEC. *Before we move on to the next page,* what problems have we encountered so far that might be good choices? (I.e., what NP-hard problems kind of *sound like* ELEC?)

Proving ELEC is in NP-hard, continued...

We are going to reduce from SAT to ELEC! (So congratulations if SAT was among your guesses to the previous question!)

Complete the following reduction (by filling in the parts following the "..."):

Define switches S_i in ELEC that represent...

For each switch S_i, define a branch node B_1^i that connects to S_i's "up" wire, and a branch node B_2^i that connects to S_i's "down" wire.

Define load nodes L_j in ELEC that represent...

Connect branch nodes to load nodes as follows: ...

Solve the ELEC instance. Then, the answer to SAT is YES if and only if...

Proving correctness of your reduction to ELEC

Complete the following proof of correctness (by filling in the parts following the "..."):

Consider the case where the answer to the original SAT instance is YES. This means there exists a truth assignment such that all clauses in the instance evaluate to TRUE. We can then construct a solution to our reduction's ELEC instance as follows: ...

Therefore, if the SAT instance has answer YES, our reduction will return YES.

Now, consider the case where the answer to our reduced ELEC instance is YES. This means there exists a...

We can therefore construct a solution to the original SAT instance as follows: ...

Therefore, if the answer to the reduced ELEC instance is YES, the answer to the original SAT instance must have also been YES. This completes our proof.