Homework 8: NP-Completeness

Theory of Computing (CSE 30151), Spring 2024

Due: 2024-04-30 5pm

Instructions

- Create a PDF file (or files) containing your solutions. You can write your solutions by hand, but please scan them into a PDF.
- Please name your PDF file(s) as follows to ensure that the graders give you credit for all of your work:
  - If you're making a complete submission, name it `netid-hw8.pdf`, where *netid* is replaced with your NetID.
  - If you're submitting some problems now and want to submit other problems later, name it `netid-hw8-part123.pdf`, where `123` is replaced with the problem number(s) you are submitting at this time.
- Submit your PDF file(s) in Canvas.

Problems

1. Define a *neural network* with ℓ inputs and m hidden units to be a function that takes a sequence of inputs

   $$x_1, \ldots, x_\ell \in \{0, 1\}$$

   and computes

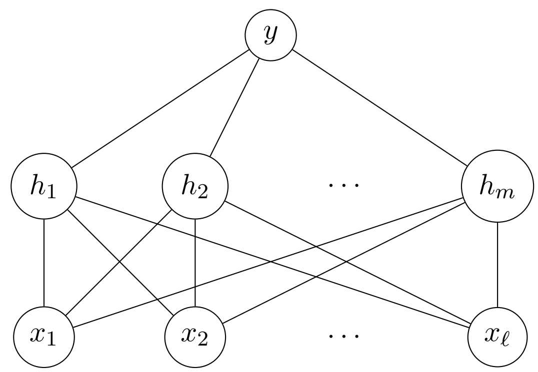
   $$\begin{aligned} h_j &= H\left(\sum_{i=1}^{\ell} x_i u_{ij} + a_j\right) & j = 1, \ldots, m \\ y &= H\left(\sum_{j=1}^{m} h_j v_j + b\right) \end{aligned}$$

where

$$H(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{otherwise.} \end{cases}$$

The weights u_{ij}, a_j, v_j, and b can be any rational numbers, and they don't depend on $x_1, \ldots, x_\ell$. Let's say that the size of the neural network is the number of units ($n = \ell + m + 1$).

We may visualize the dependencies between the variables like this:



Prove that it is NP-complete to decide, given a neural network with ℓ inputs, whether there is any sequence of inputs $x_1, \ldots, x_\ell$ that makes $y = 1$.

2. In the game of Digits,[1] you are given

- A target number t (e.g., 56)
- A multiset S of source numbers (e.g., $\{2, 3, 4, 5, 10, 25\}$)
- A set $\mathcal{O}$ of operations ($\{+, -, \times, \div\}$)

and the goal is to write an expression involving the source numbers, operations, and parentheses that evaluates to the target number. You don't have to use every number, but each number can only be used as many times as it occurs in S. If this possible, we say that $(t, S, \mathcal{O})$ is solvable. For example, $(56, \{2, 3, 4, 5, 10, 25\}, \{+, -, \times, \div\})$ is solvable because

$$2 \times 4 \times (10 - 3) = 56.$$

Prove that $\{+, \times\}$-DIGITS is NP-complete:

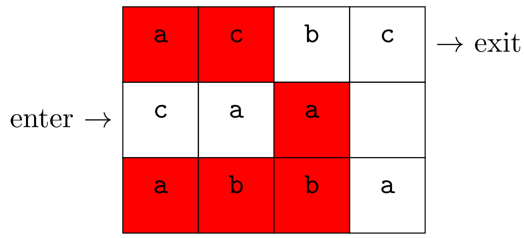
$$\{+, \times\}\text{-DIGITS} = \{\langle t, S\rangle \mid (t, S, \{+, \times\}) \text{ is solvable}\}.$$

Assume that the size of $\langle t, S\rangle$ is the total number of bits in t and S.

Challenge problem (0 points): Is the full game $\{+, -, \times, \div\}$-DIGITS also NP-complete?

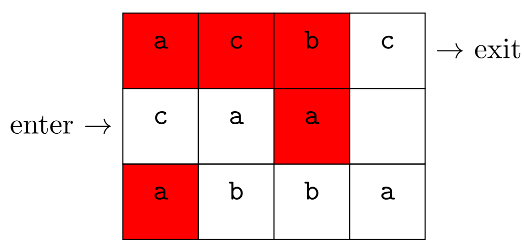
[1] `https://playdigitsnyt.com`

3. You are at the entrance to a room whose floor is made entirely of trapdoors, and underneath the floor is a pool of lava. Each trapdoor is either open (red) or closed (white). For example:



In this example, there is unfortunately no way to get from the entrance to the exit without falling into lava. (The trapdoors have rounded corners in such a way that makes diagonal moves impossible and that exceeds my artistic abilities.)

Fortunately, at the entrance to the room is a control panel that has buttons labeled with symbols. For any symbol σ, if you push the button labeled σ, then all the trapdoors labeled σ that are closed become open, and all the trapdoors labeled σ that are open become closed. For example, after you push button `b`, the floor looks like this:



Now there is a safe path from the entrance to the exit.

In general, a room can be any size rectangle, with an entrance and exit anywhere around the perimeter, any initial pattern of open/closed trapdoors, and any number of labels/buttons.

Prove that it is NP-complete to decide, given a room as defined above, whether there is a subset of buttons that you can push to make a safe path from the entrance to the exit.