Theoretical Computer Science
Exercise 4 - NP and NP-Completeness

David Bulovic 11819382

June 25, 2020

Showing that the language:
$DoppelSAT = \{\langle\phi\rangle \mid \phi$ is a Boolean formula and ϕ has at least two satisfiable variable assignments.$\}$ is NP-Complete.

(1) First to establish that $DoppelSAT \in$ NP, we show that we can verify a solution in polynomial time:

- Check the evaluation result of ϕ for all variable assignments.
- If ϕ evaluates to TRUE in 2 or more assignments accept, otherwise reject.

(2) Next, to show that we can reduce SAT[1] to $DoppelSAT$:

- Let x_0, x_1, x_2, ... , x_n be the results of ϕ for first, second, ... , n-th variable assignment respectively.
- Let ϕ' be a Boolean formula
- Next, let $\phi' = (x_0 \wedge x_1) \vee (x_0 \wedge x_2) \vee (x_0 \wedge x_3) \vee ... \vee (x_{n-1} \wedge x_n)$
- Polynomial time reduction is: $f\langle\phi\rangle = \phi'$

Now we can show that: $\phi \in DoppelSAT \iff \phi' \in SAT$. This is because ϕ' is formed so that the results of ϕ are paired each with each in a way that the result of ϕ' is TRUE if and only if two of its inputs (i.e. outputs of ϕ) are TRUE. And the other way around, if ϕ evaluates to TRUE for at least two variable assignments it is a part of $DoppelSAT$ (by definition) and ϕ' is a part of SAT (as explained before).

From (1) and (2) we can conclude that $DoppelSAT$ is NP-Complete.

[1] $SAT = \{\langle\phi\rangle \mid \phi$ is a satisfiable Boolean formula.$\}$