Giorgi Japaridze

Theory of Computability

NP-completeness

Section 7.4

Importance

NP-complete problems form a certain important subclass of **NP**. The phenomenon of NP-completeness was discovered in the early 1970s by Stephen Cook and Leonid Levin.

- If a polynomial time algorithm exists for any of the NP-complete problems, all problems in **NP** would be polynomial time solvable.

- To prove that **P=NP**, it would be sufficient to take any particular NP-complete problem **A** and show that **A∈P**.

- To prove that **P≠NP**, it would be sufficient to take any particular NP-complete problem **A** and show that **A∉P**.

- On the practical side, finding that a given problem **A** is NP-complete may prevent wasting time looking for a (probably nonexistent, or unlikely-to-be-found even if exists) polynomial time algorithm for **A**.

Boolean formulas

Boolean variables **x,y,…** take one of the two values **0** (*false*) or **1** (*true*).

Boolean operations: $\neg$ (NOT), $\wedge$ (AND), $\vee$ (OR). We write $\underline{\mathbf{A}}$ for $\neg\mathbf{A}$.

Boolean formulas are constructed from variables and operations in the standard way. Once a truth assignment for variables is given, the value of a compound formula is calculated as follows:

$\underline{0} = 1$

$\underline{1} = 0$

$0 \wedge 0 = 0$

$0 \wedge 1 = 0$

$1 \wedge 0 = 0$

$1 \wedge 1 = 1$

$0 \vee 0 = 0$

$0 \vee 1 = 1$

$1 \vee 0 = 1$

$1 \vee 1 = 1$

If **x=0** and **y=1**, what is the value of the following formula?

$(y \wedge (\underline{x} \vee \underline{y})) \vee (x \vee \underline{y})$

$(1 \wedge (\underline{0} \vee \underline{1})) \vee (0 \vee \underline{1})$

$(1 \wedge (1 \vee 0)) \vee (0 \vee 0)$

$(1 \wedge 1) \vee 0$

$1 \vee 0$

1