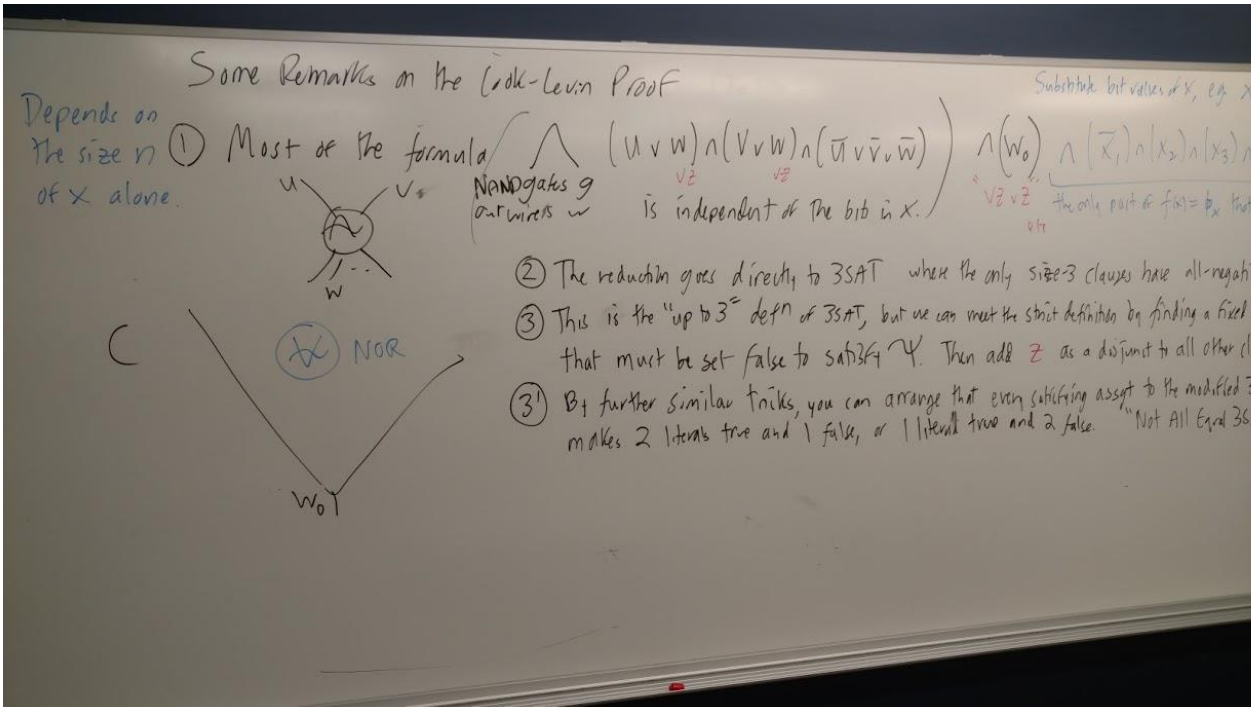
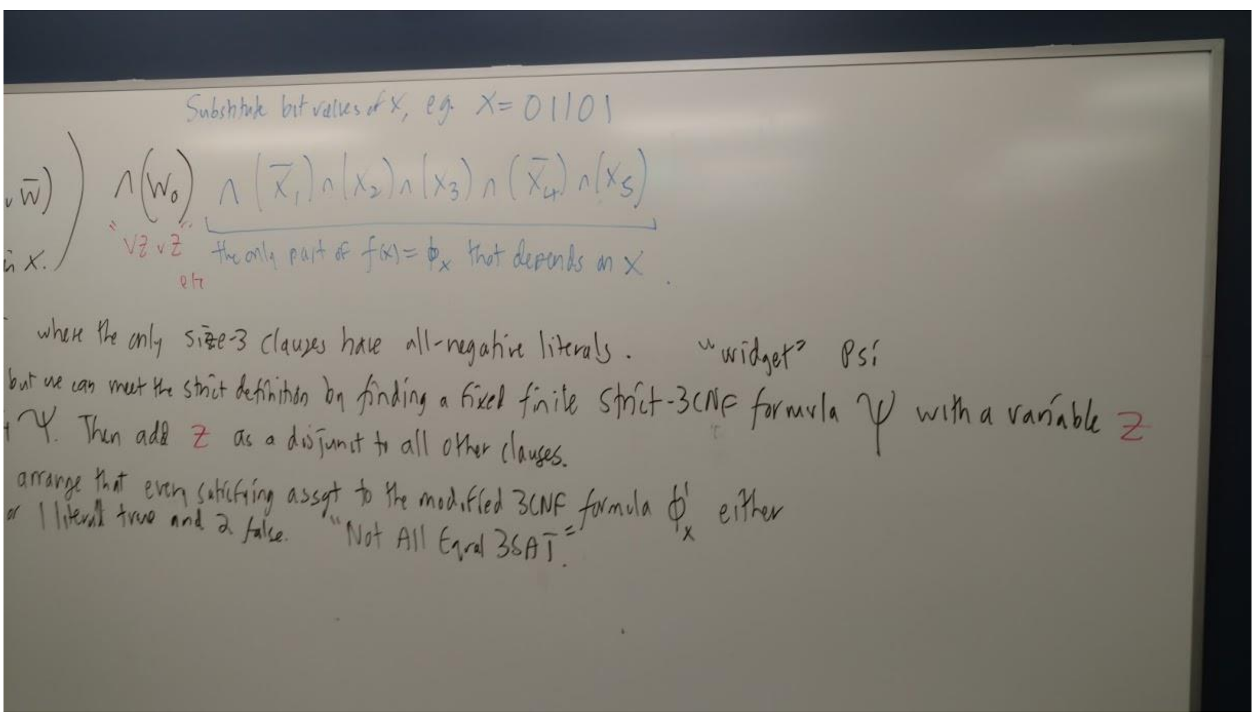
CSE596 Lecture Fri. Oct. 25, 2019





Some Remarks on the Cook-Levin Proof

Example: $\phi = (\overline{x}_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x}_2 \vee x \ldots$

Depends on the size n of x alone

① Most of the formula $\bigwedge_{\text{NAND gates } g \text{, our wires } w} (u \vee w) \wedge (v \vee w) \wedge (\overline{u} \vee \overline{v} \vee \overline{w})$ is independent of the bits in x. $\wedge (w_0) \wedge (\overline{x}_1) \wedge (x_2)$

"$\vee z$" "$\vee z$" "$\vee z \vee z$" the only part of $f(x)$

"Rungs and Gadgets Architecture"

IND. SET: INST: An undirected graph G, an integer k.

Ques: ¿ $(\exists S \subseteq V): |S| \geq k$ and $\bigwedge_{u,v \in S} (u,v) \notin E$?

Correctness. $(\Longrightarrow)$: If ϕ is satisfiable, we can take any sat. assgt $a_1 \ldots a_n \in \{0,1\}^n$.

Define S_a by choosing the corresponding rung nodes

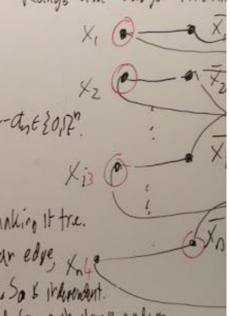
x_i if $a_i = 1$

$\overline{x_i}$ if $a_i = 0$

and choosing a node in each clause gadget for the literal making it true.

That literal and its rung partner are not connected by an edge, because all "crossing edges" go to the opposite literal. S, S_a is independent.

Conversely $(\Longleftarrow)$: If S is indept of size k, then S must have one node from each clause gadget, and the corresponding assgt to the rung nodes is enough to satisfy ϕ. ⊠



Example: $\phi = (\overline{x}_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x}_2 \vee x_4) \wedge (x_2 \vee \overline{x}_3 \vee \overline{x}_4)$

Overall Idea: Subsets S of V (obeying some constraints) will be in 1-1 correspondence with assignments $\vec{a}$ to ϕ. The chosen nodes in S from the rung nodes will define the assignment. ① Allocate rung $(x_i, \overline{x}_i)$ for each i.

② For every clause $(x_i \vee \overline{x}_j \vee x_k)$ (e.g.) allocate three nodes for a clause gadget and connect them in a triangle.

This completes $V(G)$.

After step 2, already the max possible size of an ind. set in G is $n+m$. Take $k = n+m$.

③ For every literal $\pm x_i$ in any clause C_j, add an edge from x_i in C_j to $\overline{x_i}$ in the rung. And for $\overline{x_i}$ make edge go to x_i in the rung. This completes the Construction, whose Complexity is Poly(n) time since this needs just one "pass" over ϕ.