Reduction from CNF-SAT to 3SAT

Given a CNF formula ϕ with clauses of arbitrary length, we construct an equivalent 3-CNF formula ψ such that ϕ is satisfiable if and only if ψ is satisfiable.

Construction

Let ϕ be a CNF formula with clauses $C_1, C_2, \dots, C_m$, where each clause C_i is a disjunction of literals:

$$C_i = (l_1^i \vee l_2^i \vee \cdots \vee l_{k_i}^i), \quad k_i \geq 1.$$

Note that i is an index for the literal, not the non-existent operation of raising the literal to power i.

For each clause C_i:

- If $k_i = 1$, replace C_i by four clauses: $(l_1^i \vee y_1^i \vee y_2^i)$ and $(l_1^i \vee y_1^i \vee \neg y_2^i)$ and $(l_1^i \vee \neg y_1^i \vee y_2^i)$ and $(l_1^i \vee \neg y_1^i \vee \neg y_2^i)$, with new variables y_1^i and y_2^y.
- If $k_i = 2$, replace C_i by two clauses: $(l_1^i \vee l_2^i \vee y^i)$ and $(l_1^i \vee l_2^i \vee \neg y^i)$ with a new variable y^i.
- If $k_i = 3$, keep C_i as is.
- If $k_i > 3$, introduce new variables $y_1^i, y_2^i, \dots, y_{k_i-3}^i$ and replace C_i by the conjunction of:

  $(l_1^i \vee l_2^i \vee y_1^i), (\neg y_1 \vee l_3^i \vee y_2^i), (\neg y_2 \vee l_4^i \vee y_3^i), \dots, (\neg y_{k_i-4} \vee l_{k_i-2}^i \vee y_{k_i-3}^i), (\neg y_{k_i-3}^i \vee l_{k_i-1}^i \vee l_{k_i}^i),$

  to ensure 3 literals per clause. An algorithm constructs ψ in time polynomial (linear, actually) in $(n + m)$.

Correctness

We prove on the board that ϕ is satisfiable $\iff$ ψ is satisfiable.