Simulation of Proof Systems

Nils Wisiol

Definition 1. *A function $h \in \mathcal{FP}$ is called* proof system *for a language L if the range of h is L. A string w with $h(w) = x$ is called an h-*proof *for x.*

With this definition, a proof system for L is basically a polynomial-time bounded function that enumerates L. To give an example, let *sat* be defined by

$$sat(x) = \begin{cases} \varphi & (x = \langle \alpha, \varphi \rangle \text{ and } \alpha \text{ is an satisfying assignment for } \varphi), \\ \perp & (\text{otherwise}). \end{cases}$$

Then h is a proof system for SAT.

Notice, in spite of its time bound against the input, the shortest proof of a string $w \in L$ can be be very long. There may be various proof systems for a language L. In order to make them comparable, we define the notion of *simulation* of proof systems.

Definition 2. *Let h and h' be proof systems for a language L. If there is a polynomial p and a function f such that for all $w \in \Sigma^*$*

$$h(f(w)) = h'(w)$$

and $|f(w)| \leq p(|w|)$, then h simulates *h'.*

Informally speaking, f translates h-proofs into polynomial length bounded h'-proofs. Notice, f could be hard or even impossible to calculate.

Definition 3. *A proof system h for a language L is called* optimal, *if it simulates every proof system for L.*

Definition 4. *Let* OPT *be the complexity class of all languages that have an optimal proof system.*