7.6 Effiziente Algorithmen

Um die Ergebnisse aus 7.4 übertragen zu können, müssen auch die Begriffe des Vorteils und der Irrtumswahrscheinlichkeit auf Familien verallgemeinert werden. Gegeben sei dazu eine Menge $L \subseteq \mathbb{F}_2^*$, also eine Sprache über dem binären Alphabet, wobei wie üblich $L_n := L \cap \mathbb{F}_2^n$ gesetzt wird, sowie eine Abbildung

$$f\colon L \longrightarrow \mathbb{F}_2^* \text{ mit } f(L_{r(n)}) \subseteq \mathbb{F}_2^{s(n)}, \tag{2}$$

wobei $r(n)$ die monoton wachsende Folge der Indizes i mit $L_i \neq \emptyset$ ist. Diese Abbildung soll mit einer PPS wie in (1) berechnet werden.

Beispiele

1. Die Funktion $f(x, y, z) := xy \bmod z$ für n-Bit-Zahlen x, y, z lässt sich mit einem (deterministischen) Schaltnetz

   $$C_n\colon \mathbb{F}_2^{3n} \longrightarrow \mathbb{F}_2^n$$

   der Größe $\#C_n = \mathrm{O}(n^3)$ (und Irrtumswahrscheinlichkeit 0) berechnen; hier ist $r(n) = 3n$ und $s(n) = n$.

2. Sei L die Menge der (binären Codierungen von) ungeraden Zahlen ≥ 3 und $f : L \longrightarrow \mathbb{F}_2$ der „Primzahlanzeiger“ wie in Abschnitt 7.4. Die dort vorgestellte PPS für den strengen Pseudoprimzahltest hat die Größe $\mathrm{O}(n^3)$ und konstanten Vorteil $\frac{1}{4}$ sowie konstante Irrtumswahrscheinlichkeit $\frac{1}{4}$. Mit t Basen kommt man auf die Größe $\mathrm{O}(tn^3)$ und die Irrtumswahrscheinlichkeit $\frac{1}{4^t}$.

Definition 1. Eine Funktion $\varphi\colon \mathbb{N} \longrightarrow \mathbb{R}_+$ heißt **vernachlässigbar**, wenn für jedes nichtkonstante Polynom $\eta \in \mathbb{N}[X]$ gilt

$$\varphi(n) \leq \frac{1}{\eta(n)} \text{ für fast alle } n \in \mathbb{N}.$$

Definition 2. Sei $f : L \longrightarrow \mathbb{F}_2^*$ wie in (2). Sei C eine PPS, die bei der Berechnung von f auf $\mathbb{F}_2^{r(n)}$ eine Irrtumswahrscheinlichkeit von ε_n hat, die als Funktion von n vernachlässigbar ist. Dann heißt C **effizienter Algorithmus** für f.

f heißt **effizient berechenbar**, wenn es einen effizienten Algorithmus für f gibt.

Für den Primzahltest von Rabin, also die wiederholte Ausführung des strengen Pseudoprimzahltests, kann man diese Forderung erfüllen, wenn man die Zahl t der Basen mit n wachsen lässt; damit die Familie polynomial bleibt, nimmt man t als Polynom $\tau \in \mathbb{N}[X]$. Dann hat C_n n deterministische und $n\tau(n)$ probabilistische Eingänge, die Größe $\mathrm{O}(n^3\tau(n))$ und die Irrtumswahrscheinlichkeit $\frac{1}{4^{\tau(n)}}$. Damit ist gezeigt:

Satz 1 *Der Primzahltest von* RABIN *ist ein effizienter probabilistischer Algorithmus für die Bestimmung der Primzahleigenschaft.*