Zufallsfunktionen

Definition Echte Zufallsfunktionen:

Sei $Func_n = \{f \mid f : \{0,1\}^n \to \{0,1\}^n\}$. Wir bezeichnen $f \in_R Func_n$ als *echte Zufallsfunktion* auf n Bits.

Anmerkungen:

- Können $f \in Func_n$ mittels vollständiger Wertetabelle beschreiben.
- Damit kann f als Bitstring der Länge $n \cdot 2^n$ dargestellt werden: n Bits pro $f(x)$ für alle $x \in \{0,1\}^n$.
- Es gibt $2^{n \cdot 2^n}$ Strings dieser Länge $n \cdot 2^n$, d.h. $|Func_n| = 2^{n \cdot 2^n}$.

Definition längenerhaltende, schlüsselabhängige Funktion

Sei F ein pt Algorithmus. F heißt *längenerhaltende, schlüsselabhängige Funktion* falls F eine Fkt. $\{0,1\}^m \times \{0,1\}^n \to \{0,1\}^n$ berechnet. Notation: $F_k(x) := F(k,x)$, wobei k der Schlüssel ist.

Anmerkung:

- Zur Übersichtlichkeit der Notation verwenden wir stets $m = n$.

Pseudozufallsfunktion

Definition Pseudozufallsfunktion

Sei F eine längenerhaltende, schlüsselabhängige Funktion. Wir bezeichnen F als *Pseudozufallsfunktion*, falls für alle ppt $\mathcal{D}$ gilt

$$\left|\mathrm{Ws}[\mathcal{D}^{F_k(\cdot)}(1^n) = 1] - \mathrm{Ws}[\mathcal{D}^{f(\cdot)}(1^n) = 1]\right| \leq \mathrm{negl}(n),$$

wobei $k \in_R \{0,1\}^n$ und $f \in_R \mathit{Func}_n$.

Anmerkungen:

- Die Beschreibungslänge von f ist $n2^n$ Bits, d.h. exponentiell in n.
- Daher erhält ein ppt $\mathcal{D}$ nicht f, sondern Orakelzugriff auf f und F_k.
- $\mathcal{D}$ kann nur polynomiell viele Anfragen an sein Orakel stellen.
- Danach muss $\mathcal{D}$ entscheiden, ob sein Orakel einer echten Zufallsfunktion oder einer Pseudozufallsfunktion entspricht.

Existenz von Pseudozufallsfkt

Satz Existenz von Pseudozufallsfunktionen

Pseudozufallsfunktionen existieren gdw Pseudozufallsgeneratoren existieren.

Beweisskizze. $\Rightarrow$ ist klar.

$\Leftarrow$: Sei $G : \{0,1\}^n \to \{0,1\}^{2n}$ ein Pseudozufallsgenerator mit Expansionsfaktor $\ell(n) = 2n$. Wir schreiben

$$G(x) = (G_0(x), G_1(x))$$

mit $G_i(x) \in \{0,1\}^n$ für $i \in \{0,1\}$.

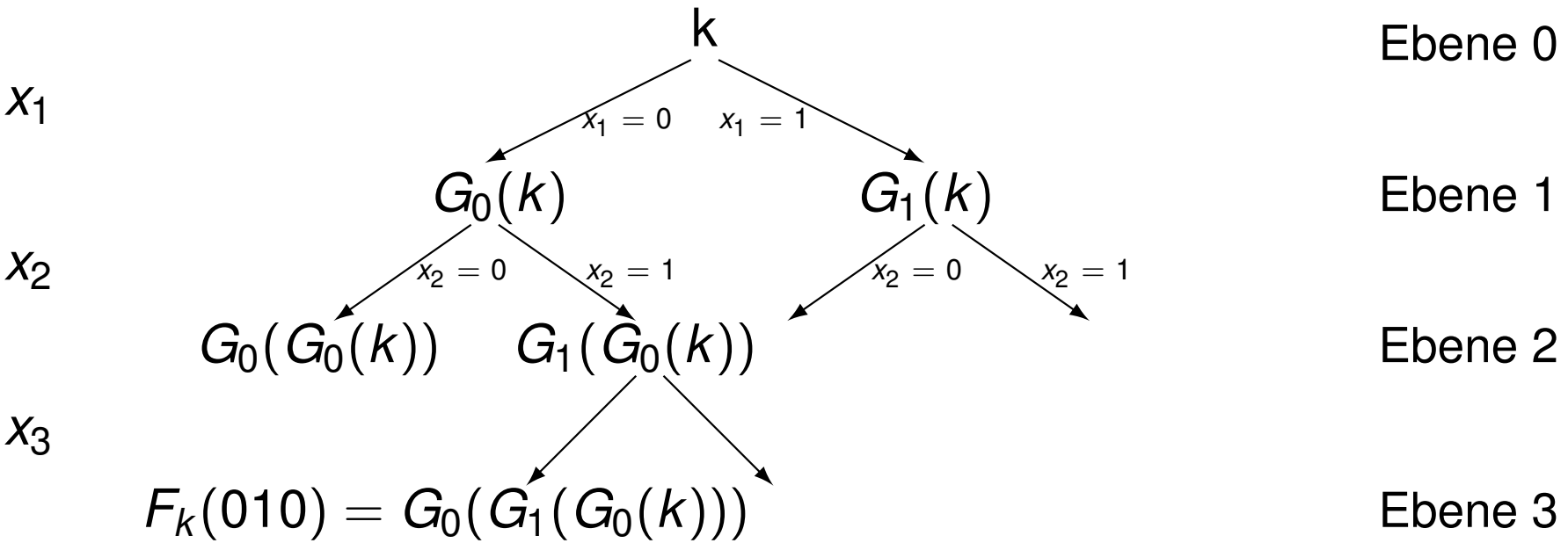
Mögliche Ideen für die Konstruktion von F_k:

- 1-bit PRF: $F_k(b) := G_b(k)$ für $b \in \{0,1\}$.
- Verallgemeinerung auf n bits?
  $F_k(x) := (G_{x_1}(k), \ldots, G_{x_n}(k))$ für $x = (x_1, \ldots, x_n) \in \{0,1\}^n$
  Aber: F_k ist keine PRF!

Konstruktion von PRF aus PRG

Algorithmus Pseudozufällige Funktion F_k

Für $k \in \{0,1\}^n$ und $x = (x_1, \ldots, x_n) \in \{0,1\}^n$ definiere
$F_k(x) = G_{x_n}(G_{x_{n-1}}(\ldots(G_{x_1}(k))\ldots))$.



- Behauptung: F_k ist eine Pseudozufallsfunktion.

Beweis der PRF Eigenschaft

Definiere Hybride $H^{(i)}$ so dass von Ebene 1 bis Ebene i uniforme Schlüssel $k_{x_1 \ldots x_i}$ verwendet werden:

$$H^{(i)}(x) := G_{x_n}(\ldots G_{x_{i+1}}(k_{x_1 \ldots x_i}) \ldots),$$

mit $k_{z_1 \ldots z_j} \in_R \{0,1\}^n$ für alle $z = z_1 \ldots z_j \in \{0,1\}^j$ und $0 \leq j \leq i$.

Dann gilt:

- $H^{(n)}(x) = f(x)$ für zufälliges $f \in_R Func_n$.
- $H^{(0)}(x) = F_k(x)$

Sei $\mathcal{D}$ ein erfolgreicher Angreifer gegen die PRF, der maximal $q(n)$ Orakelanfragen macht, für ein Polynom $q(n)$.
Zu zeigen: es existiert ein erfolgreicher Angreifer $\mathcal{A}$ gegen den PRG G.

Unabhängige Instanzen eines PRGs

Hilfslemma

Sei $G : \{0,1\}^n \to \{0,1\}^{\ell(n)}$ ein PRG und $q(n)$ ein beliebiges Polynom. Dann gilt für alle ppt Algorithmen $\mathcal{A}$:

$$|\mathrm{Ws}[\mathcal{A}(U) = 1] - \mathrm{Ws}[\mathcal{A}(T) = 1]| \leq \mathrm{negl}(n),$$

wobei der Wahrscheinlichkeitsraum definiert ist über die Münzwürfe von $\mathcal{A}$ und

- $U = (U_1, \ldots, U_{q(n)}) \in (\{0,1\}^{\ell(n)})^{q(n)}$ mit $U_j \in_R \{0,1\}^{\ell(n)}$;
- $T = (T_1, \ldots, T_{q(n)}) \in (\{0,1\}^{\ell(n)})^{q(n)}$ mit $T_j = G(s_j)$ für $s_j \in_R \{0,1\}^n$.

Beweis: Übungsaufgabe.

Konstruktion von $\mathcal{D}'$

Nach Lemma reicht zu zeigen: es ex. Angreifer $\mathcal{D}'$, der zwischen

- $R = (R^1, \ldots, R^{q(n)}) \in_R (\{0,1\}^{2n})^{q(n)}$ (zufällig) und
- $R = (R^1, \ldots, R^{q(n)}) \in (\{0,1\}^{2n})^{q(n)}$ mit $R^j = (r_0^j, r_1^j) = G(s_j)$ für $s_j \in_R \{0,1\}^n$ (pseudozufällig)

unterscheiden kann.

Algorithmus Unterscheider $\mathcal{D}'$

EINGABE: 1^n, $R = ((r_0^1, r_1^1), \ldots, (r_0^{q(n)}, r_1^{q(n)})) \in (\{0,1\}^{2n})^{q(n)}$ mit $(r_0^i, r_1^i) = G(s_i)$ oder $R \in_R (\{0,1\}^{2n})^{q(n)}$

1. Setze Zähler $t := 1$ und rate $i \in \{0, \ldots, n-1\}$.
2. Beantworte Anfragen $\mathcal{O}(x)$ für $x = (x_1, \ldots, x_n) \in \{0,1\}^n$ von $\mathcal{D}$ wie folgt:
   - Baue PRF Baum bis Ebene i dynamisch auf, mit $k_{x_1 \ldots x_j} \in_R \{0,1\}^n$ für $0 \leq j \leq i$.
   - Ebene $i+1$: $k_{x_1 \ldots x_{i+1}} := r^t_{x_{i+1}}$. Zähler $t := t + 1$.
   - $\mathcal{O}(x) := G_{x_n}(\ldots(G_{x_{i+2}}(k_{x_1 \ldots x_{i+1}}))\ldots)$

AUSGABE: Ausgabe von $\mathcal{D}$.

Analyse von $\mathcal{D}'$

1. Fall 1: R ist pseudozufällig. Dann

$$\begin{aligned}
\mathrm{Ws}[\mathcal{D}'(R) = 1] &= \sum_{j=0}^{n-1} \mathrm{Ws}[i = j] \cdot \mathrm{Ws}[\mathcal{D}'(R) = 1 \mid i = j] \\
&= \sum_{j=0}^{n-1} \mathrm{Ws}[i = j] \cdot \mathrm{Ws}[\mathcal{D}^{H^{(j)}(\cdot)}(1^n) = 1] \\
&= \frac{1}{n} \sum_{j=0}^{n-1} \mathrm{Ws}[\mathcal{D}^{H^{(j)}(\cdot)}(1^n) = 1]
\end{aligned}$$

2. Fall 2: $R = U$ is zufällig. Dann analog

$$\mathrm{Ws}[\mathcal{D}'(U) = 1] = \frac{1}{n} \sum_{j=0}^{n-1} \mathrm{Ws}[\mathcal{D}^{H^{(j+1)}(\cdot)}(1^n) = 1]$$

Insgesamt gilt für den Vorteil von $\mathcal{D}'$:

$$\begin{aligned}
\mathrm{negl}(n) &= |\mathrm{Ws}[\mathcal{D}'(R) = 1] - \mathrm{Ws}[\mathcal{D}'(U) = 1]| \\
&= \frac{1}{n} \left| \sum_{j=0}^{n-1} \mathrm{Ws}[\mathcal{D}^{H^{(j)}(\cdot)}(1^n) = 1] - \mathrm{Ws}[\mathcal{D}^{H^{(j+1)}(\cdot)}(1^n) = 1] \right| \\
&= \frac{1}{n} \left| \mathrm{Ws}[\mathcal{D}^{H^{(0)}(\cdot)}(1^n) = 1] - \mathrm{Ws}[\mathcal{D}^{H^{(n)}(\cdot)}(1^n) = 1] \right| \\
&= \frac{1}{n} \left| \mathrm{Ws}[\mathcal{D}^{F_k(\cdot)}(1^n) = 1] - \mathrm{Ws}[\mathcal{D}^{f(\cdot)}(1^n) = 1] \right|
\end{aligned}$$

Also: wenn ein erfolgreicher Angreifer $\mathcal{D}$ geg de PRF existiert, dann existiert auch ein erfolgreicher Angreifer $\mathcal{D}'$ gegen den PRG. Somit folgt der Satz.