2.5 Eine allgemeine Vorhersagemethode

Das Verfahren von PLUMSTEAD (die später unter dem Namen BOYAR publiziert hat) ist weitgehend durch das Verfahren von BOYAR/KRAWCZYK verallgemeinert worden: auf Rekursionsvorschriften, die sich durch eine Linearkombination *irgendwelcher* bekannten Funktionen ausdrücken lassen. Man beschreibt es zunächst wieder besonders passend in der Sprache der kommutativen Algebra, also durch Ringe und Moduln.

Sei also R ein kommutativer Ring (mit $1 \neq 0$), und X, Z seien R-Moduln. Gegeben sei eine Familie von Abbildungen

$$\Phi^{(i)} : X^i \longrightarrow Z \text{ für } i \geq h,$$

die wir uns als bekannt denken, und eine lineare Abbildung

$$\alpha : Z \longrightarrow X,$$

die als geheim angesehen wird (also als interner Parameter des zu beschreibenden Zufallsgenerators). Damit wird eine Folge $(x_n)_{n\in\mathbb{N}}$ in X erzeugt:

- $x_0, \ldots, x_{h-1} \in X$ werden als Startwerte gesetzt.
- Sind $x_0, \ldots, x_{n-1}$ schon erzeugt für $n \geq h$, so sei

$$\begin{aligned} z_n &:= \Phi^{(n)}(x_0, \ldots, x_{n-1}) \in Z, \\ x_n &:= \alpha(z_n) \in X. \end{aligned}$$

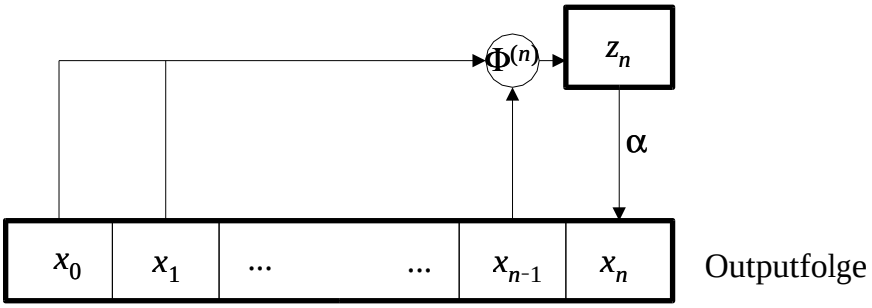

Abbildung 4: Ein allgemeiner Generator

Hier kann also sogar, allgemeiner als bisher, jedes Folgenglied von *allen* vorhergehenden, also von der gesamten „Vergangenheit“ abhängen. Damit ein solches Verfahren sinnvoll zur Zufallserzeugung eingesetzt werden kann, müssen die $\Phi^{(i)}$ natürlich effizient berechenbar sein – im Fall $R = \mathbb{Z}/m\mathbb{Z}$ und $X = R^k$ etwa soll der Aufwand höchstens polynomial mit $\log(m)$ und k wachsen.

Beispiele

1. Der lineare Kongruenzgenerator: $R = \mathbb{Z}/m\mathbb{Z} = X$, $Z = R^2$, $h = 1$, $x_n = ax_{n-1} + b$,

$$\Phi^{(i)}(x_0, \ldots, x_{i-1}) = \begin{pmatrix} x_{i-1} \\ 1 \end{pmatrix},$$

$$\alpha \begin{pmatrix} s \\ t \end{pmatrix} = as + bt.$$

2. Der linear-inversive Kongruenzgenerator: R, X, Z, h, α wie oben, $x_n = ax_{n-1}^{-1} + b$,

$$\Phi^{(i)}(x_0, \ldots, x_{i-1}) = \begin{pmatrix} x_{i-1}^{-1} \bmod m \\ 1 \end{pmatrix}.$$

3. Kongruenzgeneratoren höheren Grades: $R = \mathbb{Z}/m\mathbb{Z} = X$, $Z = R^{d+1}$, $h = 1$, $x_n = a_d x_{n-1}^d + \cdots + a_0$,

$$\Phi^{(i)}(x_0, \ldots, x_{i-1}) = \begin{pmatrix} x_{i-1}^d \\ \vdots \\ x_{i-1} \\ 1 \end{pmatrix},$$

$$\alpha \begin{pmatrix} t_0 \\ \vdots \\ t_d \end{pmatrix} = a_d t_0 + \cdots + a_0 t_d.$$

4. Beliebige Kongruenzgeneratoren: $R = \mathbb{Z}/m\mathbb{Z}$, $x_n = s(x_{n-1})$, $h = 1$. Ist m prim, so lässt sich jede Funktion $s : R \longrightarrow R$ als Polynom vom Grad $< m$ schreiben. Ist m zusammengesetzt, so verwendet man eben statt der Basis aus den Monomen die Basis $\{e_0, \ldots, e_{m-1}\}$ mit $e_i(j) = \delta_{ij}$ von R^R. Die Basis-Darstellung ist $s = \sum_{i=0}^{m-1} s(i)e_i$. Man nimmt $X = R$, $Z = R^m$ und

$$\Phi^{(i)}(x_0, \ldots, x_{i-1}) = \begin{pmatrix} e_0(x_{i-1}) \\ \vdots \\ e_{m-1}(x_{i-1}) \end{pmatrix},$$

$$\alpha \begin{pmatrix} t_0 \\ \vdots \\ t_{m-1} \end{pmatrix} = s(0)t_0 + \cdots + s(m-1)t_{m-1}.$$

Dass die $\Phi^{(i)}$ effizient berechenbar sein sollen, kann hier, egal welche Basis verwendet wird, nur bedeuten, dass eine Familie s_m von Funktionen auf $\mathbb{Z}/m\mathbb{Z}$ gegeben ist, die sich einheitlich als Linearkombination

einer Teilmenge der Basis beschreiben lassen, die höchstens polynomial mit $\log(m)$ wächst.

5. Mehrstufige Kongruenzgeneratoren werden natürlich auch erfasst, wenn man h gleich der Rekursionstiefe nimmt.

6. Auch nichtlineare Schieberegister sind Beispiele, siehe den nächsten Abschnitt 2.6.

Für die Kryptoanalyse nimmt man wie gesagt an, dass die $\Phi^{(i)}$ bekannt sind, aber α unbekannt ist. (Später wird im Fall $R = \mathbb{Z}/m\mathbb{Z}$ auch noch m als unbekannt angenommen.) Die Frage ist: Kann man aus einem Anfangsstück $x_0, \ldots, x_{n-1}$ $(n \geq h)$ der Folge das nächste Glied x_n bestimmen?

Dazu betrachtet man die aufsteigende Kette $Z_h \subseteq Z_{h+1} \subseteq \ldots \subseteq Z$ von Untermoduln mit

$$Z_n = Rz_h + \cdots + Rz_n.$$

Falls $Z_n = Z_{n-1}$, ist $z_n = t_h z_h + \cdots + t_{n-1} z_{n-1}$ mit $t_h, \ldots, t_{n-1} \in R$ und daher

$$x_n = t_h x_h + \cdots + t_{n-1} x_{n-1}$$

aus $x_0, \ldots, x_{n-1}$ bestimmbar ohne Verwendung von α. Ist Z ein noetherscher R-Modul, so wird nach endlich vielen Schritten der stationäre Zustand erreicht: $Z_n = Z_l$ für $n \geq l$. Ab dieser Stelle ist die Folge der x_n komplett vorhersagbar nach folgendem „Algorithmus“:

- Bilde $z_n = \Phi^{(n)}(x_0, \ldots, x_{n-1})$.
- Finde eine Linearkombination $z_n = t_h z_h + \cdots + t_{n-1} z_{n-1}$.
- Setze $x_n = t_h x_h + \cdots + t_{n-1} x_{n-1}$.

Das NOETHERsche Prinzip führt zu einer Vorhersage durch eine lineare Relation (die sich aber bei jedem Schritt ändern kann).

Damit aus dem „Algorithmus“ ein Algorithmus wird, muss das Verfahren im zweiten Schritt zum Finden einer Linearkombination algorithmisch durchführbar sein. Der Aufwand für einen vorausgesagten Wert besteht dann im wesentlichen aus der Auflösung eines linearen Gleichungssystems in Z.

In unserem Standard-Beispiel mit (bekanntem) Modul $m = 8397$, $x_0 = 2134$, $x_1 = 2160$, $x_2 = 6905$ ist

$$z_1 = \begin{pmatrix} 2134 \\ 1 \end{pmatrix}, z_2 = \begin{pmatrix} 2160 \\ 1 \end{pmatrix}, z_3 = \begin{pmatrix} 6905 \\ 1 \end{pmatrix}.$$

Der Versuch, z_3 als Linearkombination $t_1 z_1 + t_2 z_2$ zu schreiben, führt auf das Gleichungssystem (in $R = \mathbb{Z}/8397\mathbb{Z}$)

$$\begin{aligned} 2134 t_1 + 2160 t_2 &= 6905, \\ t_1 + t_2 &= 1. \end{aligned} \tag{1}$$

Durch Elimination kommt man auf $26 t_1 = -4745 = 3652$. Das Inverse von 26 mod 8397 ist 323, und daraus ergibt sich $t_1 = 4016$, $t_2 = 4382$. Damit wird $x_3 = 3778$ korrekt vorhergesagt.

Auch der Rest der Folge wird so korrekt vorhergesagt, denn es ist schon $Z_2 = Z$: Da $z_2 - z_1 = \binom{26}{0}$, ist $e_1 = \binom{1}{0} \in Z_2$ und $e_2 = \binom{0}{1} = z_1 - 2134 \cdot e_1 \in Z_2$.

Das Beispiel liefert auch eine Teilantwort auf die Frage, wann die Kette der Z_n stationär wird: Spätestens bei $Z_l = Z$, wenn das überhaupt vorkommt. Im allgemeinen kann man das nicht erwarten. Es folgt im allgemeinen auch nicht notwendig aus $Z_l = Z_{l+1}$, dass die Kette bei Z_l schon stationär ist – sie könnte später wieder ansteigen. Eine Schranke dafür, wie oft ein echter Anstieg möglich ist, gibt Satz 2.

In einem Schleifendurchlauf des Vorhersage-Algorithmus sind zwei Ereignisse möglich:

- $z_n \notin Z_{n-1}$. Dann ist keine Vorhersage für x_n möglich, aber Z_{n-1} wird zu $Z_n = Z_{n-1} + R z_n$ erweitert, und zwar echt.
- $z_n \in Z_{n-1}$. Dann wird x_n korrekt vorhergesagt.

Der Satz besagt, dass das erste dieser Ereignisse höchstens ${}^2\log(\#Z)$-mal vorkommen kann (bzw. $(\operatorname{Dim} Z)$-mal, wenn R ein Körper ist). Bei jedem dieser Vorkommnisse braucht man dann den Zugriff auf das Folgeglied x_n, um weiter zu kommen. Das befriedigt nicht ganz, entspricht bei genauem Hinsehen aber der Situation des Kryptoanalytikers, der beim Brechen einer Verschlüsselung mit einem vermuteten Schlüssel weiterarbeitet, bis sinnloser Text entsteht, dann die nächsten Zeichen zu erraten versucht, seinen vermuteten Schlüssel korrigiert und damit weiter entschlüsselt. Im übrigen kennen wir diese Situation ja schon aus dem vorigen Abschnitt. Bemerkenswert ist, dass der neue Algorithmus recht einfach ist, aber sich auch ganz auf die Vorhersage konzentriert und nicht versucht, die unbekannten Parameter zu bestimmen.