1.3 Linear Congruential Generators

As a first important class of elementary—"classical"—pseudorandom number generators we consider one-step recursive formulas that use linear congruences. They are very fast, have long periods, and their quality is easily analyzed due to their plain structure.

This simple formula generates a sequence of pseudorandom numbers:

$$x_n = ax_{n-1} + b \bmod m. \tag{1}$$

The recursive sequence $(x_n)_{n\in\mathbb{N}}$ depends on four integer parameters:

- the **module** m where $m \geq 2$,
- the **multiplier** $a \in [0 \dots m-1]$,
- the **increment** $b \in [0 \dots m-1]$,
- the **initial value** $x_0 \in [0 \dots m-1]$.

We call this recursive formula a **linear congruential generator**, in the case $b = 0$ also a **multiplicative generator**, in the case $b \neq 0$, a **mixed congruential generator**. Furthermore we call

$$s : \mathbb{Z}/m\mathbb{Z} \longrightarrow \mathbb{Z}/m\mathbb{Z}, \quad s(x) = ax + b \bmod m.$$

the **generating function** of the generator. Formula (1) then becomes

$$x_n = s(x_{n-1}).$$

Programming a linear congruential pseudorandom generator is extremely easy, even in assembler languages; for Sage see Sage sample 1.1. The algorithm works very fast. Moreover the pseudorandom numbers are statistically good *if the parameters m, a, b are suitably chosen.* In contrast the choice of the initial value is unrestricted. This freedom allows a reasonable variation of the generated pseudorandom numbers.

Use of the pseudorandom sequence as a bitstream for XOR encryption requires at least that we consider the initial value x_0, or the complete parameter set (m, a, b, x_0), as effective key, and keep it secret, cf. Figure 1.5.

Remarks and Examples

1. Since x_n may assume only m different values the sequence is periodic with a period length $\leq m$; including a possible preperiod.
2. Choosing $a = 0$ obviously doesn't make sense. Also for $a = 1$ we get a useless sequence, namely $x_0, x_0 + b, x_0 + 2b, x_0 + 3b, \dots$, that also mod m contains several regular subsequences.

Sage Example 1.1 Generating pseudorandom numbers by a linear congruential random generator

```
def lcg(m,a,b,s,n):
  x = s
  outlist = []
  for i in range (0,n):
    y = (a*x + b) % m
    outlist.append(y)
    x = y
  return outlist
```

3. For $m = 13$, $a = 6$, $b = 0$, $x_0 = 1$ we get the sequence

$$6, 10, 8, 9, 2, 12, 7, 3, 5, 4, 11, 1$$

of period length 12 that looks like a fairly random permutation of the integers 1 to 12, despite the small module.

4. Choosing the multiplier $a = 7$ instead of 6 we get a much less sympathic sequence:

$$7, 10, 5, 9, 11, 12, 6, 3, 8, 4, 2, 1.$$

5. If a and m are coprime, then the sequence is purely periodic (no preperiod). For $a \bmod m$ is invertible, hence $ac \equiv 1 \pmod{m}$ for some c. Thus always $x_{n-1} = cx_n - cb \bmod m$. If $x_{\mu+\lambda} = x_\mu$ with $\mu \geq 1$, then also $x_{\mu+\lambda-1} = x_{\mu-1}$ etc., finally $x_\lambda = x_0$.

6. By induction we immediately get

$$x_k = a^k x_0 + (1 + a + \cdots + a^{k-1}) \cdot b \bmod m \tag{2}$$

for all k—a definite warning about the poor randomness of the sequence: Formula (2) allows direct access to any element of the sequence. Note that the coefficient of b is $(a^k - 1)/(a - 1)$ where the division is $\bmod m$.

7. Let $m = 2^e$ and a be even. Then

$$x_k = (1 + a + \cdots + a^{e-1}) \cdot b \bmod m$$

for all $k \geq e$, hence, after a certain preperiod, the period has length 1. More generally common divisors of a and m reduce the period. We want to avoid this effect.

8. Let d be a divisor of m. Then the sequence $y_n = x_n \bmod d$ is the analogous congruential sequence for the module d, generated by the formula $y_n = ay_{n-1} + b \bmod d$. Hence the sequence (x_n), if considered mod d, has a period $\leq d$ that might be very short.

9. This effect is especially inconvenient in the case of a power $m = 2^e$: Then the least significant bit of x_n has a period of length at most 2, hence alternates between 0 and 1, or is constant. And the k least significant bits together have a period of at most 2^k.

10. The innocuously looking example $m = 2^{32}$, $a = 4095 = 2^{12} - 1$, $b = 12794$ exhibits an extremely bad choice of parameters: From $x_0 = 253$ we get $x_1 = 1048829$ and $x_2 = 253 = x_0$.

Preferred modules are

- $m = 2^{32}$ that exhausts the 32 bit range and moreover is computationally efficient,
- $m = 2^{31} - 1$ that is the maximum 32 bit integer, and computationally almost as efficient as a power of 2. Another advantage: This number is prime (claimed by MERSENNE in 1644, proved by EULER in 1772), and this enhances the quality of the pseudorandom sequence. More generally these arguments apply to FERMAT primes $2^k + 1$ and MERSENNE primes $2^k - 1$. The next prime of this kind is $2^{61} - 1$.

Table 1.1 shows the first 100 members of a sequence that is generated with the module $m = 2^{31} - 1 = 2147483647$, the multiplier $a = 397204094$, the increment $b = 0$, and the initial value $x_0 = 58854338$, Sage code sample 1.2. Figure 1.6 gives a visual impression of this information. We see that the sequence doesn't follow any obvious rules. However it is clear that such a visual impression is not a sufficient criterion for the quality of a pseudorandom sequence.

Sage Example 1.2 Using a linear congruential random generator

```
sage: mm = 2**31 - 1
sage: aa = 397204094
sage: bb = 0
sage: seed = 58854338
sage: seq = lcg(mm,aa,bb,seed,100); seq
```

Table 1.1: 100 members of a linear congruential sequence

1292048469	319941267	173739233	1992841820
345565651	2011011872	31344917	592918912
1827933824	1691830787	857231706	1416540893
1184833417	145217588	589958351	1776690121
1330128247	558009026	1479515830	1197548384
1627901332	929586843	19840670	1268974074
1682548197	760357405	666131673	1642023821
787305132	1314353697	167412640	1377012759
963849348	971229179	247170576	1250747100
703109068	1791051358	1978610456	1746992541
177131972	1844679385	1328403386	1811091691
1586500120	1175539757	74957396	753264023
468643347	821920620	1269873360	963348259
1698955999	139484430	30476960	1327705603
1266305157	1337811914	1808105128	640050202
37935526	1185470453	2111728842	380228478
808553600	934194915	824017077	881361640
1492263703	414709486	298916786	1883338449
771128019	558671080	1935988732	798347213
120356246	1378842534	37149011	272238278
1190345324	1006355270	1161592162	1079789655
220609946	1918105148	791775291	979447727
1160648370	779600833	1170336930	1271974642
375813045	1089009771	280197098	1144249742
1236647368	1729816359	650188387	1714906064

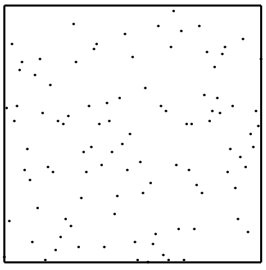

Figure 1.6: A linear congruential sequence. Horizontal axis: counter from 0 to 100, vertical axis: size of the integer from 0 to $2^{31} - 1$.