2.10 Summary

In Sections 2.1 to 2.8 we developed a prediction method whose overall workflow is depicted in Figure 2.4.

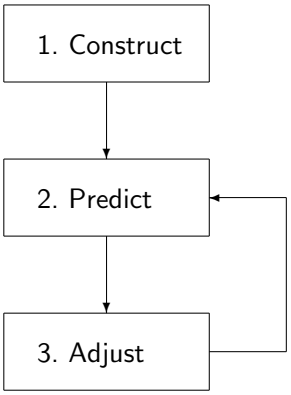

Figure 2.4: Workflow for prediction

1. By guessing plaintext the cryptanalyst finds subsequences of the key stream until she succeeds in constructing a linear relation for the state vectors (Noetherian principle).
2. Using this linear relation she predicts some more key bits.
3. If the predicted key bits are false (the plaintext ceases from making sense), then the cryptanalyst has to guess some more plaintext and to use it to adjust the parameters. Then she continues predicting bits.

This procedure is efficient for the "classical" pseudorandom generators, in particular for congruential generators—even with unknown module—and for feedback shift registers—even nonlinear ones. "Efficient" means that the computational cost is low, and also implies that the needed amount of known or correctly guessed plaintext is small.

One lesson learnt from these results is that for cryptographically secure pseudorandom generation we never should directly use the state of the generator as output. Rather we should insert a transformation between state and output that conceals the state—the output function of Figure 2.1. Section 2.9 illustrates that simply suppressing some bits —"truncating" or "decimating" the output—might be to weak as an output transformation. In the following section we'll learn about better output transformations.

There is a large twilight zone between pseudorandom generators that promise advantage to the cryptanalyst, and pseudorandom generators that

put the cipher designer's mind at ease. In any case we should prefer pseudorandom generators for which both of the procedures

- state transition,
- output function,

are nonlinear. The twilight zone where we don't know useful results on security contains (among others) quadratic congruential generators with slightly truncated output.

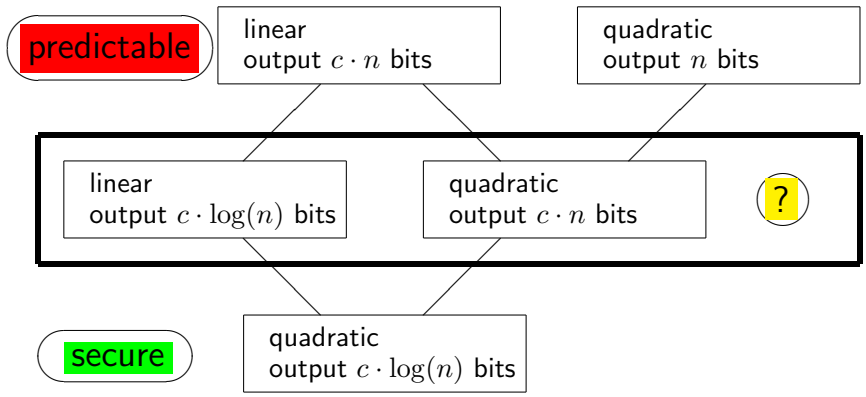

Figure 2.5: Predictable and secure congruential pseudorandom generators for n-bit integers (c a constant factor)

The following chapters present two approaches that are believed to lead to secure pseudorandom generators:

- combinations of LFSRs with a nonlinear output transformation (Chapter 3),
- nonlinear congruential generators with substantially truncated output (Chapter 4).