Knapsack Summary

CS 3000

---

1 Key Generation

Recall that a sequence of integers $\{a_0, a_1, \ldots, a_n\}$ is called superincreasing if and only if for each $i = 2, \ldots, n$, we have that

$$a_0 + a_1 + \cdots + a_{i-1} < a_i$$

That is, each term in the sequence is greater than all the previous terms in the sequence. For the Knapsack encryption scheme, each person needs their own private superincreasing sequence, which should be of a respectable length. This person then chooses some number $M > 2a_n$ to be their modulus, as well as some c, which needs to satisfy $\gcd(c, M) = 1$, then computes c' such that $cc' = 1 \pmod{M}$. For added security, one might choose several such (M, c) pairs in order to hide the private sequence better. Once choosen, we generate a new sequence $\{b_0, b_1, \ldots, b_n\}$ with the rule

$$b_i = ca_i \pmod{M} \quad \text{for all } i = 0, \ldots, n$$

This new sequence is the public key.

2 Encryption and Decryption

To encrypt a message. We first compute it's binary representation according to some scheme. This binary representation is then broken up into blocks of length n, which is the same length as the private and public sequences. We then compute

$$S = \sum_{i=0}^{n} x_i b_i = x_0 b_0 + x_1 b_1 + \cdots + x_n b_n$$

Where x_i is the i^{th} bit in the binary representation of the message. This is done for each block in the message. The number S is what is then sent over what is supposed to be an insecure channel of communication. The reveiver, knowing M and c', computes

$$S' = c'S \pmod{M}$$

Then solves the knapsack problem:

$$S' = \sum_{i=0}^{n} x_i a_i = x_0 a_0 + x_1 a_1 + \cdots + x_n a_n$$

Where the solution $\langle x_0, x_1, \ldots, x_n \rangle$ returns the original block that the sender sent.