A Factorization Algorithm Using Continued Fractions

**Aim:** Find a proper factor of the odd integer $n$ (known to be composite).

**Step 0:** (Initialization) Put $b_{-1} = 1, b_0 = a_0 := [\sqrt{n}], x_0 = \sqrt{n} - a_0$ and fix a bound $L$.

**Step 1:** For $i = 1, \ldots, L$ compute

$$a_i = [1/x_i], \quad x_i = 1/x_{i-1} - a_i, \quad b_i \equiv a_ib_i + b_{i-2} \pmod{n}.$$

**Step 2:** Compute and factor $m_i := \text{mods}(b_i^2, n)$, for $1 \leq i \leq L$.

**Step 3:** Let $B = \{-1, p_2, \ldots, p_{h-1}\}$, where the $p_i$’s are all the primes which occur more than once the $m_k$’s; i.e. either $p_i$ occurs in two different $m_k$’s or to an even power in one $m_k$.

**Step 4:** List all the $m_i$’s which are $B$-numbers, together with the B-vectors $[m_i] \in \mathbb{F}_2^h$ and find relations among these vectors.

**Step 5:** If there are no relations, increase $L$ and redo steps 1 – 4. Otherwise, for each relation $\sum \varepsilon_i[m_i] = \vec{0}$ in $\mathbb{F}_2^h$, determine:

$$c := \sqrt{\prod m_i^{\varepsilon_i}} \in \mathbb{Z} \quad \text{and} \quad b := \prod b_i^{\varepsilon_i} \pmod{n}.$$

Check that $b \not\equiv \pm c \pmod{n}$. (Note that $b^2 \equiv c^2 \pmod{n}$.)

**Step 6:** If $b \equiv \pm c \pmod{n}$ for all relations, then increase $L$ and go back to step 1.

Otherwise we have $b \not\equiv \pm c \pmod{n}$ and $b^2 \equiv c^2 \pmod{n}$ for some $b, c$ and then $d = \gcd(b + c, n)$ is a proper factor.