Another Attempt to Sieve With Small Chips—
Part II: Norm Factorization

Rainer Steinwandt
Florida Atlantic University

(joint work with Willi Geiselmann, Fabian Januszewski, Hubert Köpfer and Jan Pelzl)

Setting the Scene

Sieving step of the NFS:

yields (a,b)-pairs such that two integers

$$N_{a/r}(a,b)$$

homogeneous polynomials

have “good chances to be smooth”.

Challenge:

special purpose designs like TWIRL produce (a,b)-candidates at a high rate

➡ **fast smoothness test (+ factoring)**

Motivation

One Solution (e.g., TWIRL):

log "big" prime factors during sieving

➡ adds to the—anyway non-trivial—complexity of a device like TWIRL

Alternative idea:

Could an ECM device do all the smoothness tests + factorizations "in real time", hence eliminating the need for logging primes?

Objective

More ambitious hope:

Could an ECM device even replace a complete algebraic TWIRL device?

More modest goal here:

Is an ECM post-processing for a 1024-bit TWIRL feasible?

- TWIRL's diary logic could be avoided
- … and performance would suffice for the device from Willi's talk

NFS Parameters

No asymptotic claim, values as for TWIRL:

Sieving region:

$-A < a \leq A$, $0 < b \leq B$ with $A = 5.5 \cdot 10^{14}$, $B = 2.7 \cdot 10^{8}$

Rational side:

- $\deg(N_r) = 1$
- smoothness bound: $3.5 \cdot 10^{9}$
- two large primes $\leq 4 \cdot 10^{11}$

Algebraic side:

- $\deg(N_a) = 5$
- smoothness bound: $2.6 \cdot 10^{10}$
- two large primes $\leq 6 \cdot 10^{11}$

Required Performance

- **Expected TWIRL output (1 GHz):**

  ≈655 (a,b)-candidates per second

  (a few more at the beginning)

- **Numbers to be tested and factored:**

  N_r(a,b): no more than 216 bit

  N_a(a,b): no more than 350 bit

Design goal: ECM engine to cope with 1000 such (a,b)-pairs per second

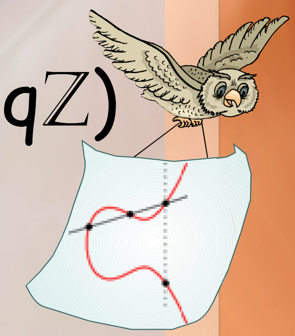
Elliptic Curve Method

Basic idea to factor n:

(a) Pick random elliptic curve $E(\mathbb{Z}/n\mathbb{Z})$ & $P \in E$

(b) For some $k = p_1^{e_1} \cdot \ldots \cdot p_r^{e_r}$, compute $k \cdot P$

(c) … hope that n has prime divisors p,q:
$k \cdot P = \mathcal{O}$ in $E(\mathbb{Z}/p\mathbb{Z})$ and $k \cdot P \neq \mathcal{O}$ in $E(\mathbb{Z}/q\mathbb{Z})$



gcd computation yields divisor of n

Here: prime bound 402 ($r=79$, $e_i = \lfloor \log_{p_i}(530) \rfloor$)

Choice of Curves

Atkin/Morain '93:

For $S:=(12:40)\in\mathbb{Q}^2$ on E: $Y^2=X^3-8X-32$ we have

- $\mathrm{ord}(S) = \infty$, and
- each $r\cdot S$ yields a curve E_r with $16\,|\,\#E_r$.

...can't do better
(Mazur '76)

Note: Unlike E, we can transform the curves E_r into Montgomery/Chudnovsky form $By^2=x^3+Ax^2+x$

Precomputation

84 curves E_r should suffice—estimated loss of "good" (a,b)-candidates <0.5%:

Precompute & store 84 rational βs from which needed curves & points mod n can be derived:

- numerators & denominators ≤17 kbit
- mod n reduction for new curve in parallel to ECM computation on available curves

Montgomery ladder
for point multiplication

2^{nd} Phase of ECM

Here: Improved Standard Continuation (Montgomery/Brent):

- (Large) primes considered: $402<q<9680$
- For each q we have r,s: $q=2(st\pm r) + 1$

t=30

Check $q\cdot Q \stackrel{?}{=} \mathcal{O}$ via $(2st+1)\cdot Q \stackrel{?}{=} \pm 2r\cdot Q$

With $\nu\cdot Q = (X_\nu : - : Z_\nu)$, test all qs at once via

$$\gcd(\Pi_{r,s} (X_{2r}Z_{2st+1}-X_{2st+1}Z_{2r}), n) \neq 1 ?$$

Composite Divisors

Here: Identified factors of n are not necessarily prime

Use "product tree" to split

$$T_{0,0} := \Pi_{r,s}\,(X_{2r}Z_{2st+1} - X_{2st+1}Z_{2r})$$

$T_{1,0} :=$ $\Pi_{\text{"some } r,s\text{"}}\,(X_{2r}Z_{2st+1} - X_{2st+1}Z_{2r})$

$T_{1,1} :=$ $T_{0,0}/T_{1,0}$

… … … …

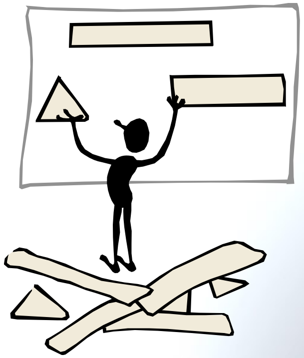
Product Tree

...**depth 3 sufficient** for our purposes

... allows **recovery of multiple divisors** at once

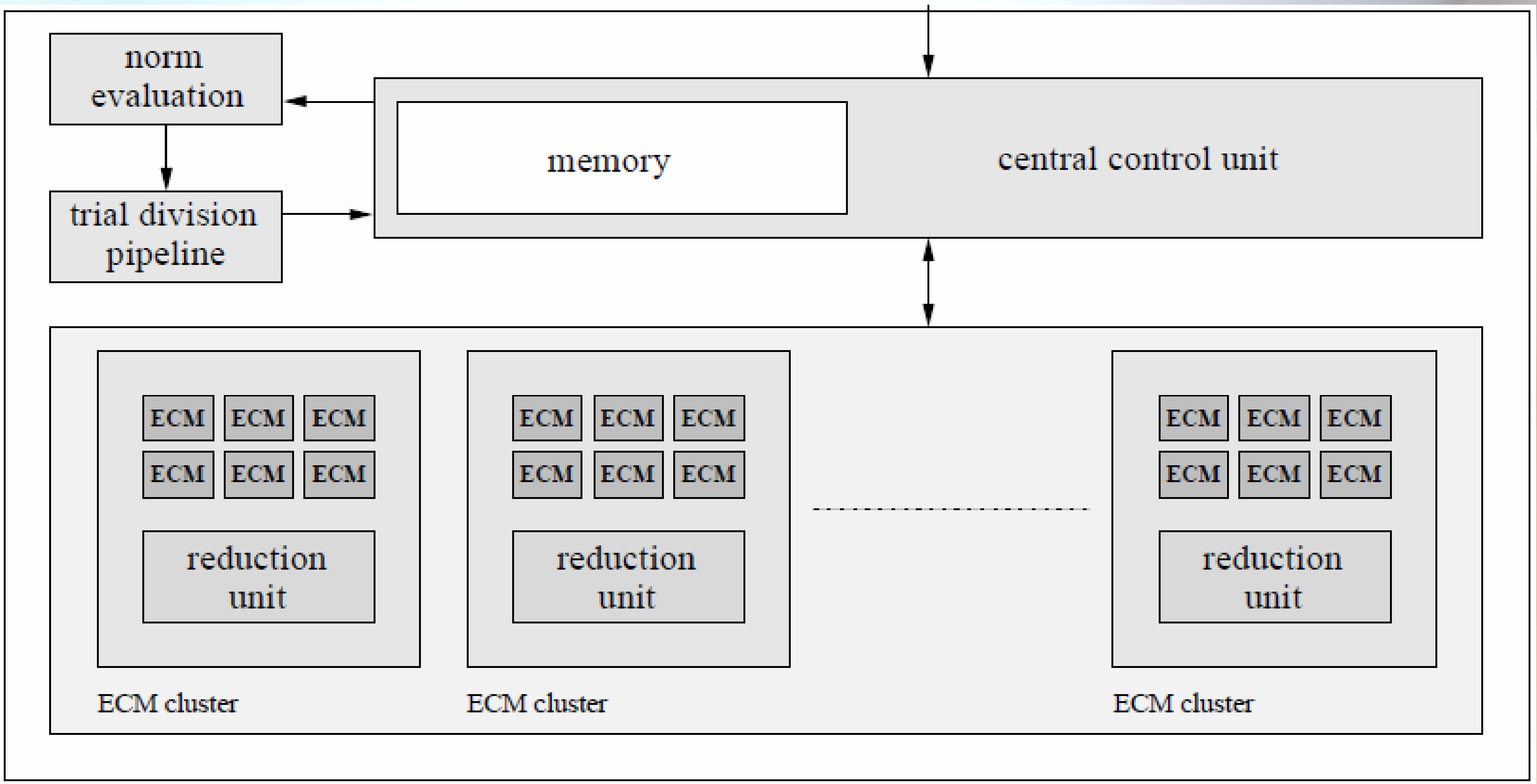
... serves as **primality test**

... **most encountered divisors will be prime** (preceding trial division for factors $<10^5$)



- 9592 primes, 108 prime powers,
- pipelined structure with 10 division circuits

Structure of a Factorization Unit



Rational & Algebraic Factorization

Rational side:

- norm comput. ≈ evaluate affine polynomial
- av. factor size after trial division: ≈ 200 bit

(a,b)-candidates passing rational tests
(along with factors)

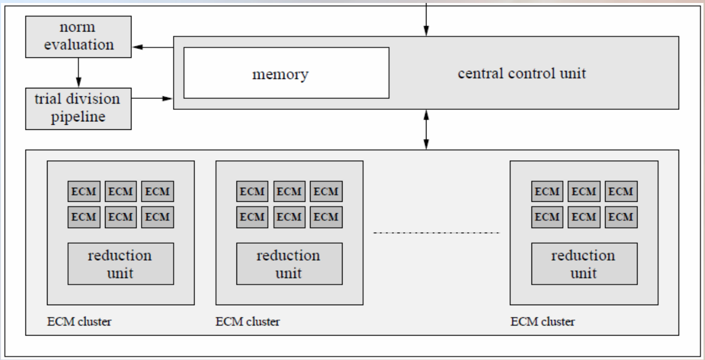
Algebraic side:

- norm comput. ≈ 5 mult. + 5 add.
- basic structure as on the rational side, identical set of elliptic curves
- operands are larger, but fewer candidates

Area Estimate

Existing work on fast arithmetic applies (Montgomery, Tenca/Koç, Pelzl et al.,...)

- With 0.13μm CMOS, one "ECM cluster" (6 ECM units + reduction unit):
  $\approx$ 4.92 mm^2 (rational)
  $\approx$ 5.76 mm^2 (algebraic)



- Trial division pipeline:
  $\approx$ 0.17 mm^2 (rational)
  $\approx$ 0.21 mm^2 (algebraic)

- Central control unit: $\leq$ 1 mm^2

Coping with TWIRL...

... for chips of size 147 mm^2 ($\approx$ Pentium 4) we can group 29 rational or 25 algebraic ECM clusters on one chip

... **Software simulations for factoring norms:**

- **Rational side:**
  on average 61 curves (→ 0.8 s per norm)

240 MHz

- **Algebraic side:**
  on average 70 curves (→ 2 s per norm)

---

5 rational + 6 algebraic chips should suffice

Conclusions

… for a 1024-bit TWIRL, logging the prime factors does not seem to be necessary: post-processing with ECM seems realistic

… coping with the (a,b)-candidates output by the device in Willi's talk should be doable

More details:

ICISC '06 proceedings, pp. 118-135
(Springer LNCS 4296)