Compact Representations: Applications and Recent Results

Michael J. Jacobson, Jr.
University of Calgary

Joint work with Laurent Imbert, Renate Scheidler, Alan Silvester, Hugh Williams

Alberta Number Theory Days 2016

The Schäffer Equation

Schäffer (1956) considered the following Diophantine equation:

$$y^q = 1^k + 2^k + \cdots + x^k, \quad k \geq 1, q > 1$$

Theorem

Finitely many solutions, unless $(k, q) \in \{(1,2), (3,2), (3,4), (5,2)\}$

Conjecture

Except for $(x, y) = (24, 70)$ when $k = q = 2$, the only solution for (k, q) not in the above set is $x = y = 1$.

A Computational Approach

Pintér, Walsh (around 2000): computational method for $q = 2$, k even

- every solution corresponds to a solution of

  $$b^2X^4 - dY^2 = 1$$

  for integers b and d from some sets depending on k
- find all solutions to each such quartic by:
  - find minimal solution $\varepsilon = X_1 + Y_1\sqrt{d}$ of $X^2 - dY^2 = 1$
  - find smallest k such that $\varepsilon^k = X_k + Y_k\sqrt{d}$ has $b \mid X_k$
  - check whether X_k/b is a square (test modulo small primes)
- verify that these solutions yield only trivial solutions of $y^2 = 1^k + 2^k + \cdots + x^k$

Computational Problems

Pintér (2000): all solutions for $k \in \{2, 4, 6, 8, 10, 14\}$

Problem: $X_1 + Y_1\sqrt{d}$ can be very large (order of $e^{\sqrt{d}}$ in general)

- for $k = 12$, there are 63 different d values, largest is $d = 1886430 = 2 \cdot 3 \cdot 5 \cdot 7 \cdot 13 \cdot 691$
- for $k = 70$, there are 511 different d values, largest has over 50 decimal digits

Question: can we compute X_k mod p efficiently without explicitly computing X_k?

Compact Representations

$\mathbb{Q}(\sqrt{\Delta}) = \{x + y\sqrt{\Delta} \mid x, y \in \mathbb{Q}\}$ — real quadratic field, discriminant $\Delta > 0$

- h_Δ — ideal class number
- ε_Δ — fundamental unit
- $R_\Delta = \log \varepsilon_\Delta$ — regulator

Lagarias (1979) and Cohen (1993):

- represent $\theta \in \mathbb{Q}(\sqrt{\Delta})$ (eg. ε_Δ) as a power-product

Formalized by Buchmann, Thiel, and Williams (1991)

- size polynomial in $O(\log \Delta)$ (instead of $O(\sqrt{\Delta})$)
- compute using arithmetic of reduced principal ideals, given $\log \theta$

Applications

Proof that computing h_Δ is in $NP \cap coNP$ (assuming GRH)

- i.e., there is a short (size polynomial in $\log \Delta$) certificate for h_Δ

Use for efficient, explicit arithmetic with large elements of $\mathbb{Q}(\sqrt{\Delta})$ (norm, multiplication, coefficients mod p, ...).

- J., Pintér, Walsh (2003): no non-trivial solutions of Schäffer Equation with $q = 2$, k even and
  - $2 \leq k \leq 58$ (unconditionally)
  - $60 \leq k \leq 70$ (assuming the generalized Riemann hypothesis)

Result relied heavily on computations of powers of ε_Δ modulo various integers m

Compact Representation: Idea

"Binary exponentiation" to find principal ideal $\mathfrak{a} = (\theta)$

Write $\lfloor \log \theta \rfloor = b_0 2^l + b_1 2^{l-1} + \cdots + b_l$

Define $s_0 = 1$, $s_j = 2s_{j-1} + b_j = \sum_{i=0}^{j} b_i 2^{j-i}$, $s_l = \lfloor \log \theta \rfloor$

Iteratively compute $\mathfrak{a}_j = (\pi_j)$ such that $\log \pi_j \approx 2s_{j-1} + b_j = s_j$:

- compute $\mathfrak{a}_{j-1}^2 = (\pi_{j-1}^2)$, given $\mathfrak{a}_{j-1} = (\pi_{j-1})$ with $\log \pi_{j-1} \approx s_{j-1}$
- reduce: $\mathrm{red}(\mathfrak{a}_{j-1}^2) = (\pi_{j-1}^2 \gamma_j)$, for $\gamma_j \in \mathbb{Q}(\sqrt{\Delta})$
- adjust using "baby steps": $\mathfrak{a}_j = \rho^k(\mathrm{red}(\mathfrak{a}_{j-1}^2)) = (\pi_j) = (\pi_{j-1}^2 \gamma_j \beta_j)$, $\beta_j \in \mathbb{Q}(\sqrt{\Delta})$, with $\log \pi_j = 2 \log \pi_{j-1} + \log \gamma_j + \log \beta_j \approx 2s_{j-1} + b_j$
- store $\lambda_j = \gamma_j \beta_j$

Compact Representation: Definition and Remarks

Compact representation of θ given by $(\lambda_0, \lambda_1, \ldots, \lambda_l)$ where

$$\theta = \pi_l = \prod_{i=0}^{l} \lambda_i^{2^{l-i}}$$

Notes:

- requires only arithmetic with *reduced* ideals (small coefficients)
- does *not* compute the π_j, only approximations of $\log \pi_j$
- computes a power-product representation of each π_j using $\pi_j = \pi_{j-1}^2 \lambda_j$

Example: $\Delta = 193$

$\varepsilon_{193} = 1764132 + 126985\sqrt{193}$

$R_{193} = \log \varepsilon_{193} \approx 15.08$

Write $\lfloor R_{193} \rfloor = 15 = b_0 2^3 + b_1 2^2 + b_2 2 + b_3$ with $b_0 = b_1 = b_2 = b_3 = 1$

$\underline{j = 0,\ (s_0 = 1)}$

$\mathfrak{a}_0 = (1)$ with $\lambda_0 = 1$

- $\mathfrak{a}_0 = (\pi_0)$ with $\pi_0 = \lambda_0 = 1$ and $\log \pi_0 = 0 < s_0$

$\underline{j = 1,\ (s_1 = 2s_0 + b_1 = 3)}$

$\mathfrak{a}_1 = \rho(\mathrm{red}(\mathfrak{a}_0^2)) = 6\mathbb{Z} + \frac{13+\sqrt{193}}{2}\mathbb{Z}$ with $\lambda_1 = \frac{13+\sqrt{193}}{2}$

- $\mathfrak{a}_1 = (\pi_1)$ with $\pi_1 = \pi_0^2\lambda_1 = \lambda_0^2\lambda_1$ and $\log \pi_1 \approx 2.56 < s_1$

Example: $\Delta = 193$ (cont.)

$j = 2$, $(s_2 = 2s_1 + b_2 = 7)$

$\mathfrak{a}_2 = \rho(\mathrm{red}(\mathfrak{a}_1^2)) = 4\mathbb{Z} + \frac{7+\sqrt{193}}{2}\mathbb{Z}$ with $\lambda_2 = \frac{179+13\sqrt{193}}{72}$

- $\mathfrak{a}_2 = (\pi_2)$ with $\pi_2 = \pi_1^2\lambda_2 = \lambda_1^2\lambda_2$ and $\log \pi_2 \approx 6.81 < s_2$

$j = 3$, $(s_3 = 2s_2 + b_3 = 15)$

$\mathfrak{a}_3 = \rho(\mathrm{red}(\mathfrak{a}_2^2)) = 1\mathbb{Z} + \frac{13+\sqrt{193}}{2}\mathbb{Z}$ with $\lambda_3 = \frac{69+5\sqrt{193}}{32}$

- $\mathfrak{a}_3 = (\pi_3)$ with $\pi_3 = \pi_2^2\lambda_3 = \lambda_1^4\lambda_2^2\lambda_3$ and $\log \pi_3 \approx 15.08$

Conclusion:

$$\begin{aligned}
\varepsilon_{193} &= \lambda_1^4\lambda_2^2\lambda_3 \\
&= \left(\frac{13+\sqrt{193}}{2}\right)^4 \left(\frac{179+13\sqrt{193}}{72}\right)^2 \left(\frac{69+5\sqrt{193}}{32}\right) \\
&= 1764132 + 126985\sqrt{193}
\end{aligned}$$

Size of a Compact Representation

Example ($\Delta = 193$): compact representation requires 39 bits, standard representation 40 bits

Example ($\Delta_c = 410286423278424$): compact representation requires 1212 bits, standard representation would require 686106 bits

Asymptotically:

- number of terms: $O(\log_2 \log \theta)$
- size of each term: $O(\log \Delta)$
- total: $O((\log_2 \log \theta) \log \Delta)$

Can we do even better?

Improvements (J., Silvester, Williams 2013)

Smaller terms: adjust recursion to accommodate "shortfall" from reduction

- aim for $2s_i + b_{i+1} - h$, where reduction shortfall is $\approx h$
- use binary expansion of $\log\theta + C$ to make up for the h's
- size of resulting compact representation: $O((\log_2 \log\theta)\log\Delta^{3/4})$

Eg. compact representation of ε_{Δ_c} requires 974 bits

Fewer terms: use signed base b expansion of $\log\theta$

- size of resulting compact representation: $O((\log_b \log\theta)\log\Delta^{\frac{b+1}{4}})$
- minimized for b between 3 and 4

Eg. using $b = 3$, size of compact representation of ε_{Δ_c} reduces to 843 bits.

Further Improvements: Better Scalar Recoding?

Seems hard to reduce size of terms further

- Use other exponentiation techniques to reduce number of terms?

Of particular interest: double-base number systems

- represent $\log \theta$ as sum/difference of terms of the form $2^a 3^b$
- number of terms is sublinear in $\log \log \theta$
- challenges: expression not "regular," size of terms varies

Other Settings (Imbert, J., Scheidler (201x))?

$C : y^2 = f(x) \in \mathbb{F}_q[x]$, q odd, f monic, square-free

- $\deg(f) = 2g + 1$ — *imaginary* hyperelliptic curve of genus g
- $\deg(f) = 2g + 2$ — *real* hyperelliptic curve of genus g

$\mathbb{F}_q(C)$ — function field of C

- quadratic extension of rational function field $\mathbb{F}_q(x)$
- similar properties to quadratic fields (ideal class group, non-trivial units when real, etc...)
- C imaginary: Picard group of C is isomorphic to ideal class group of $\mathbb{F}_q(C)$

Results (Imbert, J. Scheidler (201x))

Scheidler (1994): compact representation of $\theta \in \mathbb{F}_q(C)$ real (binary method)

Preliminary work for imaginary case:

- compact representation of $\theta \in \mathbb{F}_q(C)$ for $(\theta) = \mathfrak{a}^n$
- trick to reduce size of terms doesn't apply (unique reduced ideal in each equivalence class)
- using larger base gives improvements, between 3 and 4 is optimal

Application: Bilinear Pairings

Tate-Lichtenbaum pairing (S divisor of $C(\mathbb{F}_q)$, T divisor of $C(\mathbb{F}_{q^k})$):

$$T_n(S,T) = f_S(T)^{\frac{q^k-1}{n}} \in \mu_n \subset \mathbb{F}_{q^k}$$

where $nS = (f_S)$ (S has order n in the Picard group)

Bilinear map — used in *many* cryptographic protocols

Application of compact representations:

- Basic idea (Costello 2010): precompute f_S as (essentially) a compact representation whenever S is fixed (eg. a long-term private key)
- Use our ideas from compact representations to minimize storage costs and/or improve time to evaluate at T

Future Work: Other Settings and Applications

Real hyperelliptic function fields

- improvements to Scheidler's method?
- pairings computation in real hyperelliptic curves?
- applications for units and polynomial Pell equations?

Higher degree number and function fields:

- Done for arbitrary number fields (Thiel 1994) — implementation? improvements?
- Applications (eg. Thue and other norm equations)?