Algorithm for generating primes p and q such that q divides $p^4 \pm p^3 + p^2 \pm p + 1$

Maciej Grześkowiak

Adam Mickiewicz University
Poznań, Poland

10th Central European Conference on Cryptology
Będlewo 2010

Overview and motivations

The Gong-Harn Public Key System (GH), 1999

- Computation in $\mathbf{F}_p$
- Security in $\mathbf{F}_{p^3}$

The XTR Public Key System, 2000

- Computation in $\mathbf{F}_{p^2}$
- Security in $\mathbf{F}_{p^6}$

Overview and motivations

The Giuliani-Gong Public Key System, 2003

- Computation in $\mathbf{F}_p$
- Security in $\mathbf{F}_{p^5}$

Primes of special form

We are interested in generating large primes p and q

$$q|p^4+p^3+p^2+p+1=\Phi_5(p)$$

or

$$q|p^4-p^3+p^2-p+1=\Phi_{10}(p)$$

Approaches

Naive method

- Choose randomly p
- Compute $p^4 + p^3 + p^2 + p + 1$
- Check $p^4 + p^3 + p^2 + p + 1 = qs$?

Lenstra's method

- Choose randomly $q \equiv 1 \pmod 5$
- Compute r root of $X^4 + X^3 + X^2 + X + 1 \pmod q$
- Find $p = qk + r$

Lenstra's method

- Choose randomly r
- Check $r^4 + r^3 + r^2 + r + 1 = q$?
- Find $p = qk + r$

Proposed method

Theorem

- *Let* $f(x,y) = x^2 + xy - y^2$.
- *Let* $q = |f(a,b)|$ *be a prime,*
  where $a \equiv 1 \pmod{20}$, $b \equiv 11 \pmod{20}$.
- *Then* $\Phi_5(r_i) \equiv 0 \pmod{q}$.
- *Moreover* $r_{1,2} \equiv (\pm z_0 - a)(-2b))^{-1} \pmod{q}$,
  where $z_0 \equiv (a^2 - 4b^2)^{(q+1)/4} \pmod{q}$
- $r_{3,4} \equiv (\pm z_0' - a)(2b)^{-1} \pmod{q}$,
  where $z_0' \equiv ((a+b)^2 - 4b^2)^{(q+1)/4} \pmod{q}$.

The Algorithm

Procedure FindPrimeQ

INPUT: $f(x, y) = x^2 + xy - y^2$
While not $IsPrime(q)$ **do**

1. Choose randomly $a \equiv 1 \pmod{20}$
2. Choose randomly $b \equiv 11 \pmod{20}$
3. Compute $q = |a^2 + ab - b^2|$

Return (a, b, q)

OUTPUT:

$$q = f(a, b)$$

The Algorithm

Procedure FindRootModuloPrimeQ

INPUT: (a, b, q)

1. Compute $z = (a^2 - 4b^2)^{(q+1)/4} \pmod{q}$
2. Compute $r = (z - a)(2b)^{-1} \pmod{q}$

Return r

OUTPUT:

$$\Phi_5(r) \equiv 0 \pmod{q}$$

The Algorithm

Procedure FindPrimeP

INPUT: r, q

While not $IsPrime(p)$ **do**

1. Choose randomly k
2. Compute $p = qk + r$

Return (p)

OUTPUT:

$$q | p^4 + p^3 + p^2 + p + 1$$

Computational Complexity

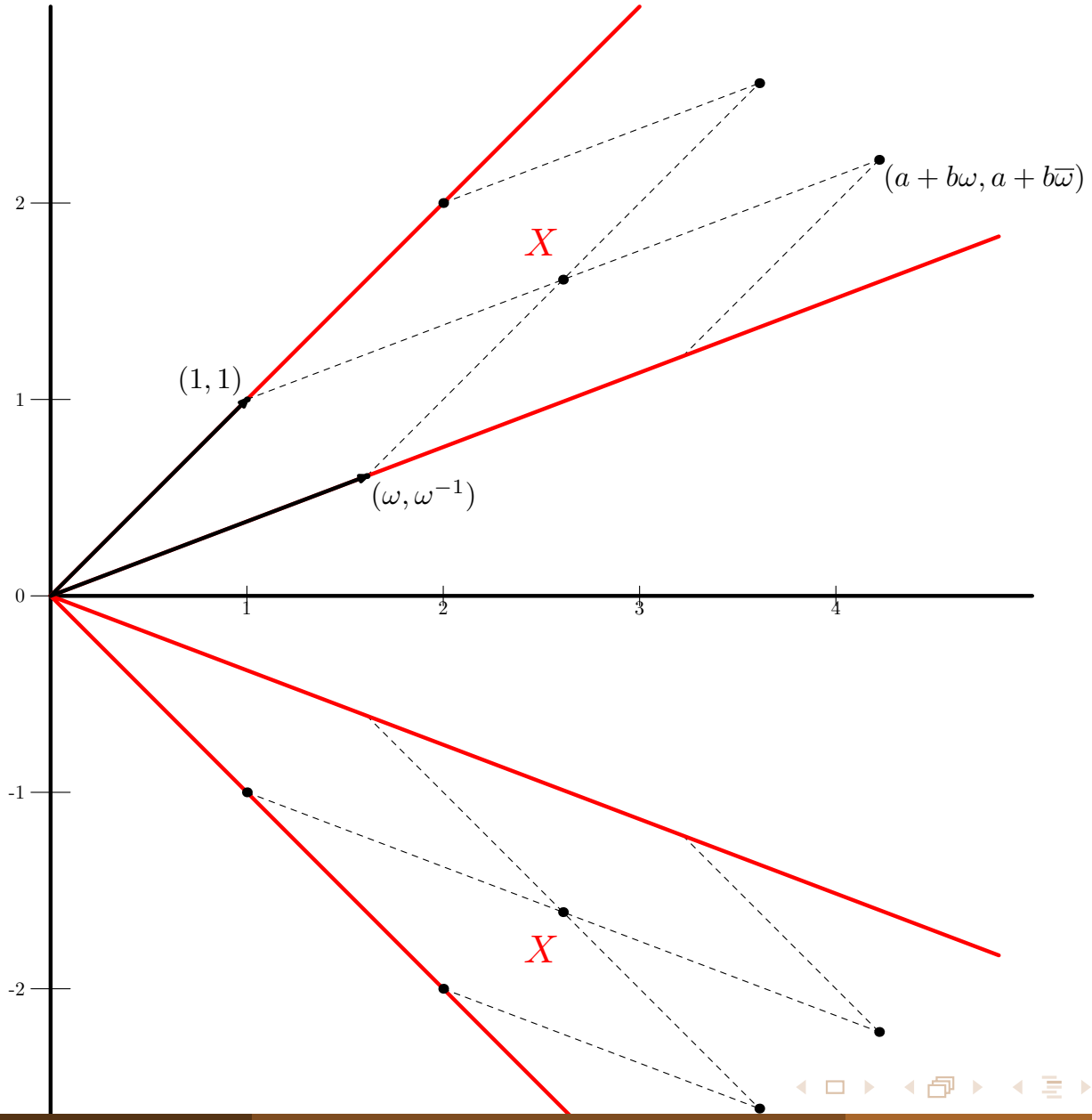
Basic notation

- Let $K = \mathbf{Q}(\omega)$, $\mathcal{O}_K = \{a + b\omega : a, b \in \mathbf{Z}\}$,$\omega = (1 + \sqrt{5})/2$,
- $N(a + b\omega) = (a + b\omega)(a + b\overline{\omega}) = a^2 + ab - b^2 = f(a, b)$

Fundamental domain $\mathcal{X}$ of $\mathbf{Q}(\omega)$

1. Consists of $A = (a + b\omega, a + b\overline{\omega})$, $a + b\omega > 0$, $a + b\overline{\omega} \neq 0$
2. $l(A) = \xi(1, 1) + \xi_1(\log \omega, -\log \omega)$, $0 \leq \xi_1 < 1$, $\xi, \xi_1 \in \mathbf{R}$

Geometric representation



The number of representations of n

Let $\alpha = a + b\omega \in \mathcal{O}_K, a \equiv 1 \pmod{20}, b \equiv 11 \pmod{20}$
Define

$$r_f(n) = \sharp\{\alpha \in \mathcal{O}_K, N(\alpha) = n, (\alpha, \overline{\alpha}) \in \mathcal{X}\} \tag{1}$$

Then

$$\sum_{n \leq x} r_f(n) \leq \frac{x \log \omega}{\sqrt{5}} + O(\sqrt{x}) \tag{2}$$

Computational Complexity

The number of primes

Let $\alpha = a + b\omega \in \mathcal{O}_K, a \equiv 1 \pmod{20}, b \equiv 11 \pmod{20}$
Define

$$\pi_f(x) = \sharp\{p \leq x, \exists \alpha \in \mathcal{O}_K, N(\alpha) = p, (\alpha, \overline{\alpha}) \in \mathcal{X}\} \tag{3}$$

Then

$$\pi_f(x) = \frac{1}{8}\frac{x}{\log x} + O\left(\frac{x}{\log^2 x}\right).$$

Conclusions

Procedure FindPrimeQ

INPUT: $f(x, y) = x^2 + xy - y^2$
While not *IsPrime*(q) **do**

1. Choose randomly $a \equiv 1 \pmod{20}$, $b \equiv 11 \pmod{20}$
2. Compute $q = |a^2 + ab - b^2|$

Return (a, b, q)

- There exist constants $c > 0$, n_0 such that for every $n \geq n_0$ and an arbitrary $\lambda \geq 1$
- the procedure finds $\alpha = a + b\omega \in \mathcal{O}_K$, $(\alpha, \overline{\alpha}) \in \mathcal{X}$ such that $N(\alpha) = q \leq x$
- with probability greater than or equal to $1 - e^{-\lambda}$
- after repeating no less than $[c \log x]$ steps of the procedure

Thank you for your attention