Group law on affine conics and applications to cryptography

Group law on affine conics and applications to cryptography

Emanuele Bellini

Technology Innovation Institute, Abu Dhabi, UAE

Nadir Murru

Universitá di Torino, Italy

Antonio J. Di Scala

Politecnico di Torino, Italy

Michele Elia

Politecnico di Torino, Italy

Abstract

In this paper, we highlight that the point group structure of elliptic curves, over finite or infinite fields, may be also observed on reducible cubics with an irreducible quadratic component. Starting from this, we introduce in a very general way a group's structure over any kind of conic. In the case of conics over finite fields, we see that the point group is cyclic and lies on the quadratic component. Thanks to this, some applications to cryptography are described, considering convenient parametrizations of the conics. We perform an evaluation of the complexity of the operations involved in the parametric groups and consequently in the cryptographic applications. In the case of the hyperbolas, the Rédei rational functions can be used for performing the operations of encryption and decryption, and the More's algorithm can be exploited for improving the time costs of computation. Finally, we provide also an improvement of the More's algorithm.

Keywords: Algorithms, Rational functions, Finite Fields, Public key cryptography, Groups over curves.

1. Introduction

Curves having a group's structure are classical and very important tools in cryptography. The main example is provided by elliptic curves over finite fields, whose use in cryptography was introduced, independently, by Koblitz [12] and Miller [17]. Moreover, curves with a group's structure, usually cubics or conics, can be exploited for constructing RSA–like schemes (see, e.g., [4, 8, 13, 19, 20, 22, 23, 24]) for improving the performances in the decryption procedures and having also more security than RSA in some contexts, like broadcast scenarios. Many of

Email addresses: eemanuele.bellini@gmail.com (Emanuele Bellini), nadir.murru@gmail.com (Nadir Murru), antonio.discala@polito.it (Antonio J. Di Scala), michele.elia7@gmail.com (Michele Elia)

these cryptosystems were studied exploiting the properties of the Pell's hyperbola that is the set of solutions in a field $\mathbb{F}$ of the famous Pell's equation $x^2 - Dy^2 = 1$, with $D \in \mathbb{F}^*$, like in [5], where the authors exhibited an RSA-like cryptosystem over the Pell's hyperbola exploiting multi-factor moduli. The Pell's hyperbola and its group's structure have been widely studied not only for the cryptographic applications, but also for the natural interest that inspires, see, e.g., [2] and [11].

In this paper, we first focus on the group's structure of the Pell's hyperbola, highlighting the similarity with that of the elliptic curves. Starting from this, we introduce in a very general way a group's structure over any kind of conics (section 2). Then, we focus on conics defined over finite fields, studying their structure as cyclic groups. This allows to use them in cryptographic applications, especially considering convenient parametrizations (section 3). Hence, we perform an evaluation of the complexity of the operations involved in the parametric groups, providing also an improvement of a specific algorithm (section 4).

2. Group's structure of conics

In the following, we will refer with the term *product*, in symbols $\otimes$, for the operation between two points of a conic and we use the term *addition*, in symbols $\oplus$, for the operation over elliptic curves.

Given two points $\mathbf{A} = (x, y)$ and $\mathbf{B} = (w, z)$ of the Pell's hyperbola in the affine plane, their product is obtained as

$$\mathbf{A} \otimes \mathbf{B} = (xw + yzD, xz + yw) \tag{1}$$

i.e.,

$$(x + y\sqrt{D})(w + z\sqrt{D}) = (xw + yzD) + (xz + yw)\sqrt{D}$$

that is, from the product of elements in $\mathbb{F}(\sqrt{D})$. This product is usually known as the Brahmagupta product and it can be also introduced in a geometric way [25]. In fact, let $\mathfrak{O} = (1, 0)$ be a fixed point, the product of two points $\mathbf{A}$ and $\mathbf{B}$ is defined as the intersection $\mathbf{A} \otimes \mathbf{B}$, with the Pell's hyperbola, of the line through $\mathfrak{O}$ which is parallel to the line through $\mathbf{A}$ and $\mathbf{B}$. Let us note that $\mathfrak{O}$ plays the role of the identity. This construction is the same that gives the operation between points of elliptic curves, even if it appears slightly different, as we will point out below.

Given two points $\mathbf{A}$ and $\mathbf{B}$ of an elliptic curve (of equation $y^2z = x^3 + axz^2 + bz^3$) in the projective plane, we consider the intersection $\mathbf{C}$, with the elliptic curve, of the line through $\mathbf{A}$ and $\mathbf{B}$. Then $\mathbf{A} \oplus \mathbf{B}$ is the symmetric of $\mathbf{C}$ with respect to the x–axis or, in other words, $\mathbf{A} \oplus \mathbf{B}$ is the intersection between the elliptic curve with the line through $\mathbf{C}$ and the identity, which is, in this case, the point at the infinity.

The above geometric construction on the Pell's hyperbola gives the product (1). To see that we have to think the Pell's hyperbola as the quadratic component of the projective cubic $\mathcal{E} : zx^2 - Dy^2z = z^3$, where $[x : y : z]$ are the projective coordinates. Thus, given two points $\mathbf{A}$ and $\mathbf{B}$ of the Pell hyperbola, the line through them intersects the cubic $\mathcal{E}$ in a point $\mathbf{C}$ that can be only a point on the line at infinity $z = 0$. Then, considering the intersection between the Pell's hyperbola with the line through $\mathbf{C}$ and the identity, which is in this case $\mathfrak{O} = (0, 1)$, we get $\mathbf{A} \otimes \mathbf{B}$. See Figure 1.

As a matter of fact, this hyperbola point-group structure is simply another way to see the operation over a degenerated cubic with two components. Therefore, we can consider a more general situation where Pell's hyperbola is substituted by any quadric, i.e. hyperbola, ellipsis, or parabola, furthermore, as we will see, the identity point can be any point of the conic.

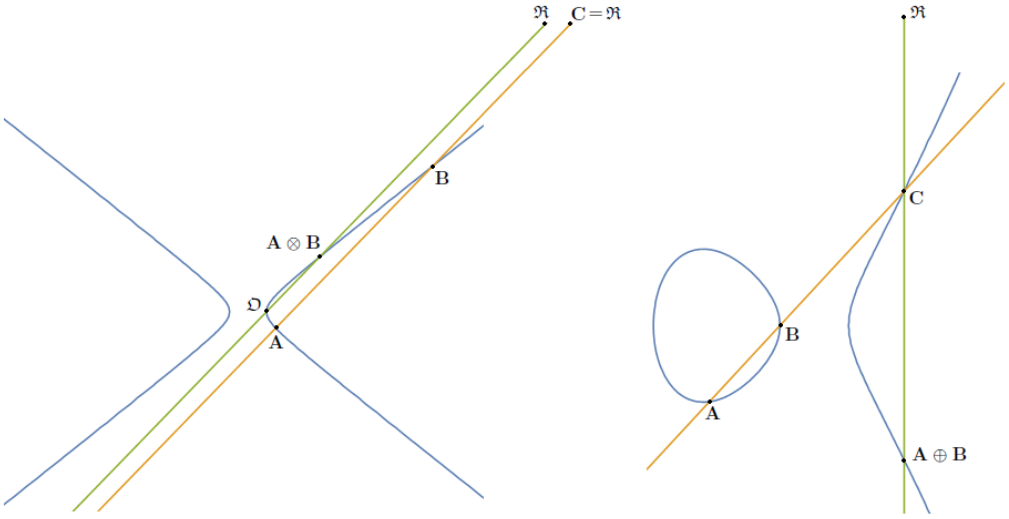

Figure 1: On the left, the geometric construction of the product between two points of the Pell's hyperbola, where $\mathfrak{R}$ is the point at the inifnity; on the right the same construction on an elliptic curve.

Let $C[\mathbb{F}]$ be a projective cubic with a quadratic component $Q[\mathbb{F}]$ of affine equation

$$ex^2 + 2gxy + fy^2 + dx + hy + k = 0.$$

Fix a point $\mathfrak{O} = (\alpha, \beta)$ on $Q[\mathbb{F}]$, then the product of two points **A** and **B** on the quadric is $\mathbf{A} \otimes \mathbf{B}$ defined as follows. Let **C** be the intersection between $C[\mathbb{F}]$ and the line through **A** and **B** (that is a point on the line at infinity). The product $\mathbf{A} \otimes \mathbf{B}$ is the intersection between the line through **C** and $\mathfrak{O}$ and the quadric $Q[\mathbb{F}]$. The geometric view immediately shows that the operation $\otimes$ is commutative with $\mathfrak{O}$ as the identity, as well as the existence of inverses.

The associativity can be proved geometrically or algebraically. In this second instance, it is convenient to refer to quadrics in canonical, or reduced, form:

$$\begin{array}{ll} x^2 - Dy^2 = \ell & \text{for hyperbolas and ellipses, i.e., } g^2 - ef \neq 0, \\ y = ex^2 + k & \text{for parabolas, i.e., } g^2 - ef = 0. \end{array}$$

The proof is indirect by showing first that we have a parametrization, and thus showing that the set of parameters admits of a group structure.

Case $g^2 - ef \neq 0$. It is necessary and convenient to distinguish two further cases.

When $\ell = u^2$, let m be a parameter, and rewrite the equation $x^2 - Dy^2 = u^2$ as follows

$$\frac{Dy}{x-u} = \frac{x+u}{y} = m$$

thus solving for x and y we have

$$x = \frac{u(m^2+D)}{m^2-D} \quad y = \frac{2um}{m^2-D} \ . \tag{2}$$

The product of the two points $\mathbf{A} = (x(m_A), y(m_A))$ and $\mathbf{B} = (x(m_B), y(m_B))$ is the new point $\mathbf{P} = (x(m_P), y(m_P))$ where

$$m_P = m_A \odot m_B = \frac{m_A m_B + D}{m_A + m_B} \quad . \tag{3}$$

Using this expression, the parameter characterizing the sum of three points $\mathbf{A}, \mathbf{B}, \mathbf{C}$ is

$$\frac{m_A m_B m_C + D(m_A + m_B + m_C)}{m_A m_B + m_B m_C + m_A m_C + D}$$

the symmetry proves that the product of points is associative.

When $\ell \neq u^2$, let m be a parameter and $\mathfrak{O} = (\alpha, \beta)$ be the fixed point on the conic (which plays the role of group identity). Consider the line through $\mathfrak{O}$ with slope m, that is of equation $y - \beta = m(x - \alpha)$, which meets the hyperbola in a second point $\mathbf{B}$ of coordinates

$$x = \alpha + 2 \cdot \frac{\alpha - \beta D m}{D m^2 - 1} \quad y = \beta + 2m \cdot \frac{\alpha - \beta D m}{D m^2 - 1} \quad .$$

The product of the two points $\mathbf{A} = (x(m_A), y(m_A))$ and $\mathbf{B} = (x(m_B), y(m_B))$ is the point $\mathbf{P} = (x(m_P), y(m_P))$ where

$$m_P = m_A \odot m_B = \frac{(m_A m_B + D)\alpha - (m_A + m_B)\beta}{-(m_A m_B + D)\beta + (m_A + m_B)\alpha} \quad . \tag{4}$$

Again, the symmetry occurring in the product of three points proves that the product of points is associative.

Case $g^2 - ef = 0$. Let m be a parameter and $\mathfrak{O} = (\alpha, \beta)$ be the fixed point on the parabola, the second intersection of the line through $\mathfrak{O}$ is a point of coordinates

$$x = \frac{-\alpha e + m}{e} \quad , \quad y = \frac{\alpha^2 e^2 - 2\alpha e m + ek + m^2}{e} \quad .$$

The product of two points $\mathbf{A} = (x(m_A), y(m_A))$ and $\mathbf{B} = (x(m_B), y(m_B))$ is given by the point $\mathbf{P} = (x(m_P), y(m_P))$, where

$$m_P = m_A \odot m_B = -2\alpha e + m_A + m_B. \tag{5}$$

Again, the symmetry occurring in the product of three points proves that the product of points is associative.

We conclude this section observing that it is a well-known fact for complex algebraic geometers that smooth curves with a law group have genus $g = 1$. This is so because the law group allows to define a non vanishing vector field (just by left translations), i.e., the tangent bundle is trivial. Thus the Euler characteristic $\chi = 2 - 2g$ vanishes, hence $g = 1$. Any conic has genus 0, so ones wonder what is going wrong. The point is that in this case our conics are affine curves. That is to say, the argument with the Euler characteristic works for compact curves (so called projective curves).

It is also a common practice to introduce the law group on a plane cubic by using the classic secant-tangent construction. That is to say, on smooth projective cubic we take a flex $\mathfrak{R}$ as neutral element and introduce the group law $*$ by using the secant-tangent construction i.e. by using the linear series of divisors $|3\mathfrak{R}|$:

$$\mathbf{P} + \mathbf{Q} + \mathbf{P}^{-1} * \mathbf{Q}^{-1} \underset{\text{lin}}{\equiv} 3\mathfrak{R}$$

The converse of this procedure is not widely known. Namely,

Theorem 1. *Let C be a complete smooth curve which is also an algebraic group $(C, *)$. Then C has genus $g = 1$ and $(C, *)$ is abelian. Moreover, for arbitrary $\mathbf{P}, \mathbf{Q} \in C$:*

$$\mathbf{P} + \mathbf{Q} + \mathbf{P}^{-1} * \mathbf{Q}^{-1} \underset{\text{lin}}{\equiv} 3\mathfrak{R}$$

where $\mathfrak{R}$ is the neutral element of $*$ and $\mathbf{P}^{-1}$ the inverse. Thus $3\mathfrak{R}$ is a very ample divisor and the law group $*$ comes from the classical secant-tangent construction by using the embedding into $\mathbb{P}^2$ given by the complete linear series $|3\mathfrak{R}|$.

For the proof of this Theorem we use the following results from [7].

Theorem 2. *Any abelian variety is a commutative group and a projective variety.*

Theorem 3 (Theorem of the square)**.** *Let D be a divisor on an abelian variety A defined over a field $\mathbf{k}$. For any $x, y \in A(\mathbf{k})$, we have*

$$(D + x) + (D + y) \underset{lin}{\equiv} D + (D + x + y).$$

Proof. of Theorem 1. Here we intend a complete curve as in [9, Chapter IV]. Denote by $\mathbf{P} * \mathbf{Q}$ the law operation and by $\mathbf{P}^{-1}$ the inversion of $\mathbf{P}$. As we explained above, $g = 1$, because using left multiplication we get a never vanishing differential $\mathrm{d}z$, i.e. the canonical bundle=cotangent bundle is trivial. So any other differential, say α, is a multiple of $\mathrm{d}z$:

$$\alpha = f\mathrm{d}z$$

where f is a regular function hence constant since C is complete. Thus $g = 1$.

That $*$ is abelian is a special case of Theorem 2. By putting $D = \mathbf{P} * \mathbf{Q}$, $x = \mathbf{Q^{-1}}$, $y = \mathbf{P^{-1}}$ in Theorem 3 we get

$$\mathbf{P} + \mathbf{Q} \underset{\text{lin}}{\equiv} \mathfrak{R} + \mathbf{P} * \mathbf{Q}$$

where $\mathfrak{R}$ is the neutral element of $*$. Since this is true for arbitrary points we get:

$$\mathbf{P} * \mathbf{Q} + \mathbf{P}^{-1} * \mathbf{Q}^{-1} \underset{\text{lin}}{\equiv} 2\mathfrak{R}$$

hence for arbitrary $\mathbf{P}, \mathbf{Q}$ we get

$$\mathbf{P} + \mathbf{Q} + \mathbf{P}^{-1} * \mathbf{Q}^{-1} \underset{\text{lin}}{\equiv} 3\mathfrak{R}$$

So $|3\mathfrak{R}|$ is a complete g^2_3, i.e. a linear system $2 = \dim|3\mathfrak{R}|, 3 = \deg(3\mathfrak{R})$, and it is very ample as follows from [9, pag. 307]. Then C is embedded in $\mathbb{P}^2$ as a cubic by $|3\mathfrak{R}|$. The equation

$$\mathbf{P} + \mathbf{Q} + \mathbf{P}^{-1} * \mathbf{Q}^{-1} \underset{\text{lin}}{\equiv} 3\mathfrak{R}$$

tells us that the law group $*$ is given by the secant-tangent construction. □

Remark 1. *The binary operation $*$ on the conic can be extended to all pairs $\mathbf{P}, \mathbf{Q}$ of the singular projective cubic with exclusion of the pair $\{\mathbf{S}_1, \mathbf{S}_2\}$ of two singular points intersection of the conic with the line at infinity. Here is the explicit formula*

$$[x : y : u] * [w : z : v] = [xw + yzD : xz + yw : uv].$$

We conclude this section summarizing the parameters and products corresponding to each conic.

Conic	Parameter	Product
$x^2 - Dy^2 = \ell,\ \ell = u^2$	$m = \dfrac{x+u}{y}$	$m_A \odot m_B = \dfrac{m_A m_B + D}{m_A + m_B}$
$x^2 - Dy^2 = \ell,\ \ell \neq u^2$	$m = \dfrac{y-\beta}{x-\alpha}$	$m_A \odot m_B = \dfrac{(Dm_A m_B + 1)\alpha - (m_A + m_B)\beta D}{(-(Dm_A m_B + 1)\beta + (m_A + m_B)\alpha)D}$
$y = ex^2 + k$	$m = (x+\alpha)e$	$m_A \odot m_B = -2\alpha e + m_A + m_B$

Table 1: Parameters and products corresponding to the conics, where (α, β) is the identity.

3. Group and group order

When $\mathbb{F}$ is a finite field, the point product defines a finite group on the quadrics which depends on a single parameter, thus, it is expected that these groups are cyclic. That is the case is proved along with the determination of group order.

Let $q = p^m$ be an odd prime power, and consider the curve $x^2 - Dy^2 - z^2 = 0$ in the projective plane. The points at infinity have coordinates $[\pm\sqrt{D} : 1 : 0]$, where $\sqrt{D}$ belongs to $\mathbb{F}_q$ if D is a square in the field, otherwise it belongs to the extension field $\mathbb{F}_{q^2}$, i.e., it is not an $\mathbb{F}_q$-point. Considering the Pell's equation written as $Dy^2 = x^2 - 1$, and rising both sides to the power exponent $\frac{q-1}{2}$ we have

$$(x^2 - 1)^{\frac{q-1}{2}} = \begin{cases} 1 & \text{if } D \text{ is a square and } y \neq 0 \\ -1 & \text{if } D \text{ is not a square and } y \neq 0. \end{cases}$$

Therefore, considering q prime we have the following results, (see, e.g., [16, Theorem 5]).

- D not square: since $\left((x^2 - 1)^{\frac{q-1}{2}}\right)^{q+1} = 1$ if $x^2 - 1 \neq 0$, the total number of points on the curve is $q + 1$ including the two points $(\pm 1, 0)$.
- D square: since $(x^2 - 1)^{\frac{q-1}{2}} = 1$ if $x^2 - 1 \neq 0$, the total number of points is $q - 1$ including the two points $(\pm 1, 0)$, but in this case we must also count the two points at infinity, that in projective coordinate are $[\pm\sqrt{D} : 1 : 0]$, hence in total the group has still order $q + 1$.

This count also holds for any square ℓ, in which case the coordinates x and y are changed by the same factor u, with $u^2 = \ell$.
If ℓ is not a square in the field, the set of solutions can be obtained as the product

$$(u + \sqrt{D}v)(x_o + \sqrt{D}y_o)$$

where $u + \sqrt{D}v$ is any solution of the Pell equation and $x_o + \sqrt{D}y_o$ is a fixed solution of the equation $x^2 - Dy^2 = \ell$.

The group of the parabola of equation $y = ex^2 + k$ is cyclic of order $q + 1$. Clearly the equation identifies q points, the further point is the point at infinity, which is characterized by the homogeneous equation $yz = ex^2 + kz^2$. By setting $z = 0$, we obtain $x = 0$, thus the point at infinity has homogeneous coordinates $(0, 1, 0)$.

In many algorithms for cryptographic applications, the use of the arithmetic of algebraic curves typically requires the evaluation of a multiple $n\mathbf{A}$ or a power $\mathbf{A}^n$ of a point on the curve with large n. The use of quadrics does not avoid this computation, however the computational

cost may be significantly smaller, although maintaining the same strength against cryptanalytic attacks. Furthermore, the point product is a complete operation, which means that the formulas are defined for all pairs of input points on the quadric, with no exceptions for doubling, for the neutral element, or for negatives, and the output is always a point on the curve [6]. In particular, in the case of quadrics, it appears convenient to perform the operations on the corresponding set of parameters, like, e.g., in [4, 20], where the authors only focused on the Pell's hyperbola, while here we have showed that it is possible to work on more general conics. We give a sketch of the RSA-like cryptosystem on general conics.

- **Key generation.**
  - Take two large prime numbers p and q and evaluate $N = pq$
  - Take an integer ϵ coprime with $(p+1)(q+1)$ and evaluate $\delta \equiv \epsilon^{-1} \pmod{(p+1)(q+1)}$

  The couple (N, ϵ) is the public key, the triple (p, q, δ) is the private key.

- **Encryption.**
  Let $M_x, M_y \in \mathbb{Z}_N^*$ be two plaintexts.
  - Find the conic where (M_x, M_y) lies (for instance, in the case of the Pell hyperbola, evaluate $D = (M_x^2 - 1)/M_y^2$ in $\mathbb{Z}_N$; D identifies the Pell hyperbola where the point lies)
  - Find the parameter m corresponding to (M_x, M_y) (for instance, in the case of the Pell hyperbola $m = (1 + M_x)/M_y$ in $\mathbb{Z}_N$)
  - Evaluate $c = m^{\odot\epsilon} \pmod{N}$, where powers are evaluated with respect to $\odot$ that is the operation over the set of parameters (i.e., the operation described by equations (3), (4), (5), depending on the type of conic).

  The encrypted message is c (which will be sent to the receiver together to D in the case, e.g., of the Pell's hyperbola).

- **Decryption.**
  Let c be the ciphertext.
  - Evaluate $c^{\odot\delta} \pmod{N}$ that returns m
  - Find the point corresponding to the parameter m, i.e., the plaintexts (M_x, M_y) (for instance, in the case of the Pell hyperbola evaluate $(m^2+D)/(m^2-D)$ and $2m/(m^2-D)$ in $\mathbb{Z}_N$).

See [4] for details on the behaviour of the Pell's hyperbola over $\mathbb{Z}_N$. We provide an example of the above scheme using the Pell's hyperbola.

Example 1.

- ***Key generation.***
  Let us consider $p = 2027$, $q = 3061$, *and* $N = pq = 6204647$. *We choose as the encryption*

exponent $\epsilon = 2^{16} + 1 = 65537$*, which is also a standard choice also for the RSA scheme, because it has an efficient binary representation. Then we evaluate*

$$\delta \equiv \epsilon^{-1} \pmod{(p+1)(q+1)} = 44249.$$

The couple $(6204647, 65537)$ *is the public key and* $(2027, 3061, 44249)$ *is the private key.*

- ***Encryption.***
  Now we would like to encrypt, e.g, the plaintext

$$(M_x, M_y) = (1098585, 5538173) \in \mathbb{Z}_N^* \times \mathbb{Z}_N^*.$$

For doing this, we evaluate the Pell hyperbola $x^2 - Dy^2 = 1$ *where* (M_x, M_y) *lies and we find*

$$D \equiv (M_x^2 - 1)M_y^{-2} \pmod{N} = 4993512,$$

since we want to find D such that $M_x^2 - DM_y^2 \equiv 1 \pmod{N}$*. Thus, we evaluate the parameter corresponding to the point* (M_x, M_y) *by*

$$m \equiv (1 + M_x)M_y^{-1} \pmod{N} = 1310780,$$

see first row of Table 1 with $u = 1$*. Then, we encrypt m using the encryption exponent* ϵ*:*

$$c \equiv m^{\odot\epsilon} \pmod{N} = 1263767,$$

where the powers are evaluated with respect to the product (3) *and c is the ciphertext corresponding to the initial plaintext* (M_x, M_y)*.*

- ***Decryption.***
  Given the ciphertext $c = 1263767$ *and* $D = 4993512$*, the receiver can use the private ket* δ *for recovering the parameter m corresponding to the plaintext* (M_x, M_y)*:*

$$c^{\odot\delta} \pmod{N} = 1310780.$$

Once the parameter $m = 1310780$ *is recovered, we are able to retrieve the plaintext evaluating the point over the Pell hyperbola* $x^2 - Dy^2 = 1$ *corresponding to this parameter:*

$$M_x \equiv (m^2 + D)(m^2 - D)^{-1} \pmod{N}, \quad M_y \equiv 2m(m^2 - D)^{-1} \pmod{N}$$

where we have used equation (2)*.*

Remark 2. *If we construct the previous cryptosystem using a parabola of equation* $y = ex^2 + k$*, with identity* (α, β)*, then the cryptosystem appears to be weak, since the exponentiation with respect to the operation described by* (5) *has a closed form:*

$$m^{\odot\epsilon} = -(2\epsilon - 2)\alpha e + \epsilon \pmod{N}$$

where ϵ, α, e *are known quantities. However, previously, we have studied the group's structure also for the parabola for the completeness of the discussion.*

4. Complexity of the computations

In this section, we evaluate the complexity of the operations involved in cryptosystems constructed on conics, referring to the parametric representation and considering the equations (3), (4), and (5). We study also the case given by the parabola just for completeness. The complexity is expressed in terms of number of arithmetical operations in $\mathbf{F}_q$, i.e., number of multiplications, additions, and divisions, or inversions, being well known that inversion in finite fields is an expensive operation. In particular, we will focus the attention on the operation over hyperbolas, since the main cryptosystems in the literature are developed on these curves. We will present a direct method of evaluation of the exponentiation over the set of the parameters for all the conics (subsection 4.1). Moreover, for the exponentiation over the set of parameters of hyperbolas, we evaluate the complexity of More's algorithm (subsection 4.2) and of an improvement of it that we propose in subsection 4.3.

In the following, let $n = \sum_{i=1}^{L} c_i 2^{i-1}$, with $c_i \in \{0,1\} \subset \mathbb{N}$, and $L = \lfloor \log_2 n \rfloor + 1$, be the binary representation of n, and denote with $w(n) = \sum_{i=1}^{L} c_i$ the number of symbols 1 in the binary representation of n. Three different algorithms, all exploiting the square-and-multiply method, for computing $\mathbf{A}^{\otimes n}$, with $\mathbf{A}$ a point of a conic, are described and compared. The computation scheme is the following

i) Find the parameter value m of $\mathbf{A}$

ii) Compute $m^{\odot n}$ that is the parameter of $\mathbf{A}^{\otimes n}$

iii) Find the coordinates of the resulting point $\mathbf{A}^{\otimes n}$

Since the computations at steps i) and iii) are common to every method, they are not counted in this complexity evaluation. The product of a point with itself is called doubling.

4.1. Direct Algorithm

The procedure computes and stores L doublings, denoted by x_j, $j = 1, \ldots, L$, of the initial m using the equations (3), (4), and (5), i.e., set $x_0 = m$ and $x_j = x_{j-1} \odot x_{j-1}$. Then iteratively evaluates the sum $\sum_{j=0}^{L} c_j x_j$ by means of the same equations. For doing these computations we recall the equations (3) and (4), highlighting also the doubling:

$$m_A \odot m_B = \frac{m_A m_B + D}{m_A + m_B} \quad m^{\odot 2} = \frac{m^2 + D}{m + m} \tag{6}$$

$$\begin{cases} m_A \odot m_B = \dfrac{(m_A m_B + D) - (m_A + m_B)\frac{\beta}{\alpha}}{-(m_A m_B + D)\frac{\beta}{\alpha} + (m_A + m_B)} \\ m^{\odot 2} = \dfrac{(m^2 + D) - (m + m)\frac{\beta}{\alpha}}{-(m^2 + D)\frac{\beta}{\alpha} + (m + m)} \end{cases} \tag{7}$$

respectively.

The algorithm is detailed in Figure 2. Notice that during the pre-computation phase $L - 1$ squares, $L - 1$ products, $L - 1$ doubling, $L - 1$ additions and $L - 1$ inversions are performed. Similarly, during the exponentiation phase, $2w(n)$ products, $2w(n)$ additions, and $w(n)$ inversions are performed.

Direct(m,n)

if $m=0$ **return** ∞

Set L, c_j s.t. $n=\sum_{j=1}^{L} c_j 2^{j-1}$

// Pre-computation:

$x_1 = m$

for $j=2,\ldots,L$

  $x_j = x_{j-1}^{\odot 2}$

// Exponentiation:

$y_1 = \infty$

for $j=1,\ldots,L$

  if $c_j = 1$ $\quad y_{j+1} = y_j \odot x_j$

  else $\quad y_{j+1} = y_j$

return y_{L+1}

More(m,n)

if $m=0$ or $n=0$ **return** ∞

Set L, c_j s.t. $n=\sum_{j=1}^{L} c_j 2^{j-1}$

$R_1 = m$

for $j=1,\ldots,L-1$

  $R_{j+1} = \dfrac{R_j^2+b}{2R_j+a}$

  if $c_{L-j}=1$

    $R_{j+1} = \dfrac{mR_{j+1}+b}{R_{j+1}+m+a}$

return R_{L+1}

Modified_More(m,n)

if $m=0$ or $n=0$ **return** ∞

Set L, c_j s.t. $n=\sum_{j=1}^{L} c_j 2^{j-1}$

$A_1 = m, B_1 = 1$

for $j=1,\ldots,L-1$

  $A_{j+1} = A_j^2 + bB_j$

  $B_{j+1} = 2A_jB_j + aB_j^2$

  if $c_{L-j}=1$

    $A' = A_{j+1}, B' = B_{j+1}$

    $A_{j+1} = mA' + bB'$

    $B_{j+1} = A' + (m+a)B'$

return A_{L+1}/B_{L+1}

Figure 2: Direct exponentiation algorithm (left), More's algorithm (middle), and the modified More's algorithm (right).

4.2. More's algorithm

The operation defined by equation (6) is connected to the Rédei polynomials, see [3, 4]. See also [14] for a general overview on Rédei polynomials. For the ease of the reader, below, we recall the useful definitions and results.

Rédei polynomials are defined as follows:

$$\mathcal{N}_n(D,z) = \sum_{k=0}^{\lceil \frac{n}{2} \rceil} \left(\begin{array}{c} n \\ 2k \end{array} \right) D^k z^{n-2k} \quad \mathcal{D}_n(D,z) = \sum_{k=0}^{\lceil \frac{n}{2} \rceil} \left(\begin{array}{c} n \\ 2k+1 \end{array} \right) D^k z^{n-2k-1}$$

and satisfy the linear recurrences

$$\left\{ \begin{array}{rcl} \mathcal{N}_{n+1}(D,z) & = & z\mathcal{N}_n(D,z) + d\mathcal{D}_n(D,z) \\ \mathcal{D}_{n+1}(D,z) & = & \mathcal{N}_n(D,z) + z\mathcal{D}_n(D,z) \end{array} \right.$$

or, equivalently, satisfy the homogeneous linear recurrence of order two

$$x_{n+2} - 2zx_{n+1} + (z^2 - D)x_n = 0$$

with respective initial conditions

$$\left\{ \begin{array}{lcl} x_0 = \mathcal{N}_0(D,z) = z & , & x_1 = \mathcal{N}_1(D,z) = z^2 + D \\ x_0 = \mathcal{D}_0(D,z) = 1 & , & x_1 = \mathcal{D}_1(D,z) = 2z \end{array} \right.$$

The Rédei rational function is the ratio $Q_n(D,z) = \frac{\mathcal{N}_n(D,z)}{\mathcal{D}_n(D,z)}$ and gives $z^{\odot n}$ (that is the powers of a parameter corresponding to a point of the Pell's hyperbola $x^2 - Dy^2 = 1$). Both Rédei polynomials

and rational functions can be quickly evaluated via their recurrence relations. Many properties can be deduced from the relation

$$\begin{bmatrix} \mathcal{N}_n(D,z) & D\mathcal{D}_n(D,z) \\ \mathcal{D}_n(D,z) & \mathcal{N}_n(D,z) \end{bmatrix} = \begin{bmatrix} z & D \\ 1 & z \end{bmatrix}^n$$

From this representation of Rédei polynomials we immediately have the following relations for the Rédei rational functions

$$Q_{n+m}(D,z) = Q_n(D,z) \odot Q_m(D,z) \quad Q_{nm}(D,z) = Q_n(D, Q_m(D,z)).$$

In [18], More proposed a fast algorithm for evaluating the Rédei rational functions. Though, the algorithm uses 2 inversions at each step. Precisely, in $\mathbb{F}_q$, the number of multiplications required for computing the inverse of an element is $O(\log_2 q)$. Therefore, the actual complexity of More's algorithm is $O(\log_2 n \cdot \log_2 q)$. However, as shown below, the algorithm can be modified to avoid inversions at each step by using more multiplications, and using only one inversion before returning the result.

More's algorithm mimics the square-and-multiply algorithm for evaluating powers, and evaluates the Rédei function $Q_n(x)$ of degree $n \geq 1$ with respect to $t(x) = x^2 - ax - b$ (i.e., for a more general definition of Rédei functions). It consists of

the following steps:

- Initialize $R(x) \leftarrow x$.
- For i from L to 1 updated $R(x)$ as follows:
  - $R(x) \leftarrow \frac{R^2(x)+b}{2R(x)+a}$.
  - If $c_i = 1$, set $R(x) \leftarrow \frac{xR(x)+b}{R(x)+x+a}$.
- Return $R(x)$.

Note that $x + a$ can be pre-computed. The algorithm is detailed in Figure 2.

The output of this procedure is $R(x)$, that is the Rédei rational function $Q_n(x)$, which coincides with $x^{\odot n}$ when the polynomial is $t(x) = x^2 - D$. Noting that expression $2R(x) + a$ can be evaluated as $R(x) + R(x) + a$, each step requires 1 multiplication (i.e. $R^2(x)$), 2 additions, and one division if $b_i = 0$, while if $b_i = 1$ one further multiplication, 2 further additions, and one further division are required. Note that a division can be done using one inversion and one multiplication. Notice that, the for cycle never checks the most significant bit of n. In summary, the algorithm requires $2(w(n) + L - 1)$ multiplications, $2(w(n) - 1) + 3(L - 1)$ additions ($m + a$ can be pre-computed), and $w(n) - 1 + L - 1$ inversions.

In the following section an algorithm which uses only one inversion is described and its complexity estimated.

4.3. Modified More's algorithm

A way to avoid the division at every step is to consider $R(x)$, in the More's algorithm, as the ratio of two polynomials $\frac{A(x)}{B(x)}$ so that we can update the polynomials $A(x)$ and $B(x)$ at each step and only at the end perform the quotient.

The procedure consists of the following steps:

- Initialize $A(x) \leftarrow x$ and $B(x) \leftarrow 1$.
- For i from L to 1 updated $A(x)$ and $B(x)$ as follows:
  - $A(x) = A(x)^2 + bB(x)$, and $B(x) = 2A(x)B(x) + a * B_j^2$.
  - If $c_i = 1$, set $A'(x) \leftarrow A(x)$, $B'(x) \leftarrow B(x)$, $A(x) = xA'(x) + bB'(x)$, and $B(x) = A'(x) + (x + a)B'(x)$.
- Return $R(x) \leftarrow A(x)/B(x)$.

The algorithm is detailed in Figure 2.

Also in this case, the output is $R(x)$, that is the Rédei rational function $Q_n(x)$, which coincides with $x^{\odot n}$ when the polynomial is $t(x) = x^2 - D$. Let us note that, in the iterations, no inversion is needed, and only a final quotient is computed to provide the value of the Rédei rational function.

This modified algorithm needs some additional multiplication and sum at each step, that is

1. 5 multiplications, 3 additions, and zero inversion for $L - 1$ steps:
2. 3 multiplications, 2 additions ($m + a$ can be pre-computed), and zero inversion for $w(n) - 1$ steps.

In summary the total number of multiplications is $5(L - 1) + 3(w(n) - 1)$, the number of additions is $3(L - 1) + 2(w(n) - 1)$, plus a final division.

For the sake of comparison the complexities relative to the three algorithms are summarized in Table 2. Note that an inversion costs about $\log_2 q$ multiplications.

	Direct			More			Modified More		
	P	A	I	P	A	I	P	A	I
(3)	$2(L + w - 1)$	$2(L - 1) + 3w$	$L + w - 1$	$2(L + w - 2)$	$3(L - 1) + 2(w - 1)$	$L + w - 2$	$5(L - 1) + 3(w - 1)$	$3(L - 1) + 2(w - 1)$	1
(4)	$4L + 4w$	$4L + 3w$	$L + w$	–	–	–	–	–	–
(5)	–	$2L + 2w$	–	–	–	–	–	–	–

P = # products, squares A= # additions, doublings I = # inversions

Table 2: Complexity of three algorithms to compute the exponentiation $m^{\odot n}$.

Example 2. *In this example, we show the intermediate steps of the three algorithms to compute $m^{\odot n}$ work. In particular, we set $m = 2$, $n = 11 = (1011)_2$, $L = 4$, $D = 2$, $a = 0$, and $b = D$.*

Direct*:*

Pre-computation of x:

$$x_1 = m = 2$$
$$x_2 = x_1 \odot x_1 = \\ = \frac{x_1^2 + D}{2x_1} = \frac{3}{2}$$
$$x_3 = x_2 \odot x_2 = \\ = \frac{x_2^2 + D}{2x_2} = \frac{17}{12}$$
$$x_4 = x_3 \odot x_3 = \\ = \frac{x_3^2 + D}{2x_3} = \frac{577}{408}$$

Exponentiation:

$$y_0 = \infty$$
$$i = 1, c_1 = 1$$
$$y_2 = y_1 \odot x_1 = \\ = \frac{y_1 x_1 + D}{y_1 + x_1} = \frac{2}{1}$$
$$i = 2, c_2 = 1$$
$$y_3 = y_2 \odot x_2 = \\ = \frac{y_2 x_2 + D}{y_2 + x_2} = \frac{10}{7}$$
$$i = 3, c_3 = 0$$
$$c_3 = 0$$
$$y_4 = y_3 = \frac{10}{7}$$
$$i = 4, c_4 = 1$$
$$y_5 = y_4 \odot x_4 = \\ = \frac{y_4 x_4 + D}{y_4 + x_4} = \frac{11482}{8119}$$

More*:*

$$R_1 = m = 2$$
$$i = 1, c_3 = 0$$
$$R_2 = \frac{R_1^2 + b}{2R_1 + a} = \frac{3}{2}$$
$$i = 2, c_2 = 1$$
$$R_3 = \frac{R_2^2 + b}{2R_2 + a} = \frac{17}{12}$$
$$R_3 = \frac{mR_3 + b}{R_3 + m + a} = \frac{58}{41}$$
$$i = 3, c_1 = 1$$
$$R_4 = \frac{R_3^2 + b}{2R_3 + a} = \frac{3363}{2378}$$
$$R_4 = \frac{mR_4 + b}{R_4 + m + a} = \frac{11482}{8119}$$

Modified More*:*

$$A_1 = m = 2$$
$$B_1 = 1$$
$$i = 1, c_3 = 0$$
$$A_2 = A_1^2 + bB_1 = 6$$
$$B_2 = 2A_1B_1 + aB_1^2 = 4$$
$$i = 2, c_2 = 1$$
$$A_3 = A_2^2 + bB_2 = 68$$
$$B_3 = 2A_2B_2 + aB_2^2 = 48$$
$$A' = A_3, B' = B_3$$
$$A_3 = mA' + bB' = 232$$
$$B_3 = A' + (m + a)B' = 164$$
$$i = 3, c_1 = 1$$
$$A_4 = A_3^2 + bB_3 = 107616$$
$$B_4 = 2A_3B_3 + aB_3^2 = 76096$$
$$A' = A_4, B' = B_4$$
$$A_4 = mA' + bB' = 367424$$
$$B_4 = A' + (m + a)B' = 259808$$
$$\frac{A_4}{B_4} = \frac{11482}{8119}$$
$$\frac{A_4}{B_4} = \frac{11482}{8119}$$

According to Table 2, since $L = 4, w(n) = 3$ we expected

- $2(L-1) + 2w = 12$ *products,* $2(L-1) + 3w = 12$ *additions, and* $L - 1 + w = 6$ *inversions, for the direct method;*
- $2(w+L-2) = 10$ *products,* $2(w-1)+3(L-1) = 13$ *additions, and* $w+L-2 = 5$ *inversions for More's method;*

- $5(L-1)+3(w-1)=21$ *products,* $3(L-1)+2(w-1)=13$ *additions, and 1 inversion for the Modified More's method.*

To conclude, we note that More's method allows to perform one less inversion compared to the Direct method. Though, half of the inversions can be pre-computed in the Direct method. Both methods perform a similar number of multiplications, while the Direct method performs more additions when the exponent has many 1's in its binary representation. Finally, if one wants to avoid inversions at the cost of increasing the number of multiplications and additions, he can perform a modifications of More's algorithm, which postpones the inversion to only the last step. In the case of the rational field, this implies working with larger integers as the algorithms is close to the end.

5. Conclusions

We introduced a general group structure over any kind of conic. For the case of finite field, we also described some applications to cryptography derived from convenient parametrization of the conics. Additionally, we described three different ways of performing the group operation, and we evaluated and compared the complexity of each method.

Acknowledgment

A. J. Di Scala is member of GNSAGA of INdAM and of DISMA Dipartimento di Eccellenza MIUR 2018-2022.

We would like to thank the anonymous referees whose valuable comments allowed us to improve the contents of the paper.

References

[1] E. Bach, J. Shallit, *Algorithmic Number Theory*, MIT, Cambridge Mass., 1996.

[2] E. J. Barbeau, *Pell's equation*, Springer, New York, 2003.

[3] S. Barbero, U. Cerruti, N. Murru, *Solving the Pell equation via Rédei rational functions*. The Fibonacci Quarterly 48. (2010), p. 348–357.

[4] E. Bellini, N. Murru, *An efficient and secure RSA-like cryptosystem exploiting Rédei rational functions over conics*. Finite Fields Appl. 39. (2016), p.179–194.

[5] E. Bellini, N. Murru, *A multi-factor RSA-like scheme with fast decryption based on Rédei rational functions over the Pell hyperbola*, Lecture Notes in Computer Science, Proceedings of the Third International Conference on Numerical Computations: Theory and Algorithms (NUMTA 2019), To Appear.

[6] D. Bernstein, T. Lange, *Faster addition and doubling on elliptic curves*, International Conference on the Theory and Application of Cryptology and Information Security, Springer, (2007), p.29–50.

[7] O. Debarre, *Two or three things I know about abelian varieties*. Preprint available at https://www.math.ens.fr/~debarre/AV.pdf.

[8] N. Demytko, *A new elliptic curve based analogue of RSA*. Advances in Cryptology - EURO-CRYPT '93. (1994), p. 40–49.

[9] R. Hartshorne, *Algebraic Geometry*, Graduated text in Mathematics 52, Springer-Verlag, 1977.

[10] J. Hoffstein, J. Pipher, J.H. Silverman, *An introduction to mathematical cryptography*, Springer, New York, 2008.

[11] M. J. Jacobson, H. C. Williams, *Solving the Pell equation*, CMS Books in Mathematics, Canadian Mathematical Society, 2009.

[12] N. Koblitz, *Elliptic curve cryptosystems*. Math. Comp. 48. (1987), p. 203–209.

[13] K. Koyama, *Fast RSA-type schemes based on singular cubic curves* $y^2+axy \equiv x^3 \pmod{n}$. Advances in Cryptology - EUROCRYPT '95. (1995), p. 329–340.

[14] R. Lidl, G.L. Mullen, G. Turnwald, *Dickson polynomials*, Longman, London, 1993.

[15] A.J. Menezes, P.C. van Oorschot, S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, 1997.
[16] A.J. Menezes, S.A. Vanstone, *A note on cyclic groups, finite fields, and the discrete logarithm problem*, Appl. Algebra Engrg. Comm. Comput. 3 (1992), p. 67–74.
[17] V. Miller, *Use of elliptic curves in cryptography*. Lecture Notes in Computer Science 85 (1985), p. 417–426.
[18] W. More, *Fast evaluation of Rédei functions*. Appl. Algebra Engrg. Comm. Comput. 6 (1995), p. 171–173.
[19] N. R. Murthy, M. N. S. Swamy, *Cryptographic applications of Brahmagupta-Bhaskara equation*. IEEE Trans. Circuits Syst. 53 (2006), p. 1565–1571.
[20] S. Padhye, *A public key cryptosystem based on Pell equation*. IACR Cryptol. ePrint Arch. (2006).
[21] J. Pieprzyk, T. Hardjono, J. Seberry, *Fundamentals of Computer Security*, Springer, New York, 2003.
[22] K. Rao, P. S. Avadhani, D. L. Bhaskari, K. Sarma, *An identity based en-cryption scheme based on Pell's equation with Jacobi symbol*. Int. J. Appl. Sci. Eng. Res. 1 (2013), p. 17–20.
[23] K. Sarma, P. S. Avadhani, *Public key cryptosystem based on Pell's equation using the Gnu Mp library*. Int. J. Comput. Sci. Eng. 3 (2011), p. 739–743.
[24] T. C. Segar, R. Vijayaragavan, *Pell's RSA key generation and its security analysis*, 2013 Fourth International Conference on Computing, Communications and Networking Technologies (IC-CCNT), IEEE (2013), p. 1–5.
[25] O. Veblen, J. W. Young, *Projective Geometry, Vol. I*, Boston: Ginn & Co., 1938.
[26] J. von zur Gathen, J. Gerhard, *Modern Computer Algebra*, Cambridge Univ. Press, 1999.
[27] H.C. Williams, A modification of the RSA public-key encryption procedure, *IEEE Trans. on Inform. Th.*, IT-26(6), November (1980), p.726–729.