Pell hyperbolas in DLP–based cryptosystems

Gessica Alecci, Simone Dutto, Nadir Murru

Abstract

We present a study on the use of Pell hyperbolas in cryptosystems with security based on the discrete logarithm problem. Specifically, after introducing the group structure over generalized Pell hyperbolas (and also giving the explicit isomorphisms with the classical Pell hyperbolas), we provide a parameterization with both an algebraic and a geometrical approach. The particular parameterization that we propose appears to be useful from a cryptographic point of view because the product that arises over the set of parameters is connected to the Rédei rational functions, which can be evaluated in a fast way. Thus, we exploit these constructions for defining three different public key cryptosystems based on the ElGamal scheme. We show that the use of our parameterization allows to obtain schemes more efficient than the classical ones based on finite fields.

1 Introduction

The Pell hyperbola over a field $\mathbb{K}$ is a curve defined for a fixed non-zero element $d \in \mathbb{K}$ as

$$\mathcal{C}_d(\mathbb{K}) = \big\{(x, y) \in \mathbb{K} \times \mathbb{K} \,|\, x^2 - dy^2 = 1\big\},$$

whose name comes from the famous Pell equation. It is well-known [1] that a group structure over the Pell hyperbola can be obtained by considering the Brahmagupta product that, given two points $(x_1, y_1), (x_2, y_2) \in \mathcal{C}_d(\mathbb{K})$, is

$$(x_1, y_1) \otimes_d (x_2, y_2) = (x_1x_2 + dy_1y_2, x_1y_2 + y_1x_2). \tag{1}$$

The Pell equation can be introduced considering $\mathbb{K}[t]/(t^2 - d)$, where the norm of an element $x + yt$ is $\mathcal{N}_d(x + ty) = x^2 - dy^2$ and the Brahamagupta product essentially coincides with the product in this ring. Thus, in the following, we will also write $\mathcal{N}_d(x, y) = x^2 - dy^2$ and we can observe that the Pell hyperbola is closed under $\otimes_d$ because the multiplicative property of the norm, i.e., $\mathcal{N}_d((x_1, y_1) \otimes_d (x_2, y_2)) = \mathcal{N}_d(x_1, y_1) \cdot \mathcal{N}_d(x_2, y_2)$. Moreover, the identity element of $\otimes_d$ is the vertex of the hyperbola with coordinates $(1, 0)$ and the inverse of a point (x, y) is $(x, -y)$, see [1].

If $\mathbb{K}$ is a finite field of order q, with q odd prime or power of an odd prime, then we denote it by $\mathbb{F}_q$ and the group over the Pell hyperbola is cyclic of order $q - \chi_q(d)$ (see, e.g., [17]), where $\chi_q(d)$ is the quadratic character of $d \in \mathbb{F}_q$, i.e.

$$\chi_q(d) = \begin{cases} 0 & \text{if } d = 0, \\ 1 & \text{if } d \text{ is a square in } \mathbb{F}_q, \\ -1 & \text{if } d \text{ is a non–square in } \mathbb{F}_q. \end{cases}$$

All Pell hyperbolas with same value of $\chi_q(d)$ are isomorphic. In particular, if $\chi_q(d) = \chi_q(d')$, then $d' = ds^2$ for some $s \in \mathbb{F}_q^\times$ and the group isomorphism is

$$\begin{aligned} \delta_{d,d'} : \big(\mathcal{C}_d(\mathbb{F}_q), \otimes_d\big) &\xrightarrow{\sim} \big(\mathcal{C}_{d'}(\mathbb{F}_q), \otimes_{d'}\big), \\ (x, y) &\longmapsto (x, y/s), \end{aligned} \tag{2}$$

because

$$\mathcal{N}_{d'}(x, y/s) = x^2 - d'(y/s)^2 = x^2 - ds^2y^2/s^2 = x^2 - dy^2 = 1.$$

The group structure of the Pell hyperbola was used in various cryptosystems, especially for constructing RSA-like schemes. In [13], an analogue of the RSA scheme over the Pell hyperbola is introduced. However, it requires to send twice as many bits per message with respect to classical RSA without increasing the security. For overcoming this issue, other works exploited parameterization of the Pell hyperbola, obtaining RSA-like schemes having a two times faster decryption procedure than RSA [3, 20, 21]. Moreover, recently, several works have exploited these schemes in different contexts [4, 6, 23, 25].

Since the group of the Pell hyperbola is cyclic, we can think to exploit it also in Public-Key Encryption (PKE) schemes whose security is based on the Discrete Logarithm Problem (DLP), such as the ElGamal PKE scheme [7]. In its classical version, the ElGamal scheme is based on the cyclic multiplicative group of a finite field $\mathbb{F}_p$, with p prime, and its security is guaranteed by the hardness of the DLP, which can be solved only in sub-exponential time by the index calculus algorithm. Later, an analogue of the ElGamal scheme over elliptic curves was implemented [12]. Over the years, the ElGamal scheme was modified in order to speed up the execution time and to increase the efficiency of the scheme [8, 24]. In [9], authors presented an ElGamal-like cryptosystem based on the matrices over a groupring and, in [14, 15], the authors studied a PKE scheme, called the MOR cryptosystem, similar to the ElGamal scheme over a non-abelian finite group and a group of outer automorphism. Moreover, in [18], the Digital Signature Algorithm (DSA) has been adapted exploiting the solution space of the Pell equation comparing it with the classical implementations over finite fields and elliptic curves. The author showed that the signature based on the Pell equation is more efficient than the analogue with elliptic curves and it has the same level of security of conventional DSA. Further studies and variant can be found in [10, 11, 16, 22].

As we will highlight in the next sections, there is a strict link between the group law of the Pell hyperbola and the elliptic curves, indeed the operation can be geometrically introduced with a similar construction, but with a more simple algebraic expression in the case of the Pell hyperbola. Thus, it seems interesting to translate the ElGamal scheme over the Pell hyperbola, exploiting also some specific parameterization of this curve that provides some benefits from a computational point of view. In particular, in Section 2 we will study a generalization of the group structure on Pell hyperbolas defined by the equation $x^2 - dy^2 = c$, writing the group operation in terms of Brahmagupta products and providing an explicit isomorphisms with the classical Pell hyperbolas. In Section 3, we will provide a parameterization for Pell hyperbolas exploiting an algebraic approach and connecting it also to a geometrical interpretation. The particular parameterization that we propose appears to be very useful from a cryptographic point of view because the product that arises over the set of parameters is connected to Rédei rational functions, as shown in Section 4. Then Section 5 will be devoted to the presentation of three new PKE schemes based on Pell hyperbolas and in Section 6 we will present numerical results.

2 Generalized Pell hyperbolas

In this section, we study the group structure over a generalized Pell hyperbola and we give an explicit isomorphism between a classical Pell hyperbola and a generalized one.

The equation of the Pell hyperbola is a particular case of the canonical form of hyperbolas and ellipses that, over a finite field, is given by

$$\mathcal{C}_{c,d}(\mathbb{F}_q) = \big\{(x,y) \in \mathbb{F}_q \times \mathbb{F}_q \,|\, \mathcal{N}_d(x,y) = x^2 - dy^2 = c\big\}.$$

Now, we would like to construct a product over this hyperbola that generalizes the Brahmagupta product. Moreover, we are able to define a product where the identity point is a chosen point (a,b) that can be any point in $\mathcal{C}_{c,d}(\mathbb{F}_q)$ (instead of the point $(1,0)$ as in the classical case). We construct this new product exploiting the classical Brahmagupta product $\otimes_d$ in the following way:

$$(x_1,y_1) \otimes_{a,b,c,d} (x_2,y_2) = \frac{1}{c}(a,-b) \otimes_d (x_1,y_1) \otimes_d (x_2,y_2), \tag{3}$$

where the subscripts a,b are used for highlighting the chosen identity point (a,b) and the subscripts c,d for specifying the hyperbola where the product works. Given the definition of this product, we can observe, first of all, that the generalized Pell hyperbola is closed under it since

$$\begin{aligned}\mathcal{N}_d\big((x_1,y_1) \otimes_{a,b,c,d} (x_2,y_2)\big) &= \mathcal{N}_d\left(\frac{1}{c}(a,-b) \otimes_d (x_1,y_1) \otimes_d (x_2,y_2)\right)\\ &= \frac{1}{c^2} \cdot \mathcal{N}_d(a,-b) \cdot \mathcal{N}_d(x_1,y_1) \cdot \mathcal{N}_d(x_2,y_2) = \frac{c^3}{c^2} = c.\end{aligned}$$

Moreover, the identity point for $\otimes_{a,b,c,d}$ is the chosen point (a,b) since

$$\begin{aligned}(a,b)\otimes_{a,b,c,d}(x,y) &= \frac{1}{c}(a,-b)\otimes_d (a,b)\otimes_d (x,y)\\ &= \frac{1}{c}(a^2-db^2,0)\otimes_d (x,y)\\ &= (1,0)\otimes_d (x,y) = (x,y),\end{aligned}$$

and the product is clearly commutative. Finally, the inverse of a point (x,y) is the point

$$(x,y)^{-1} = \frac{1}{c}(a,b)\otimes_d (a,b)\otimes_d (x,-y),$$

since

$$\begin{aligned}(x,y)^{-1}\otimes_{a,b,c,d}(x,y) &= \frac{1}{c}(a,-b)\otimes_d \left(\frac{1}{c}(a,b)\otimes_d (a,b)\otimes_d (x,-y)\right)\otimes_d (x,y)\\ &= \frac{1}{c}(a^2-db^2,0)\otimes_d (a,b)\otimes_d \frac{1}{c}(x^2-dy^2,0)\\ &= (1,0)\otimes_d (a,b)\otimes_d (1,0) = (a,b).\end{aligned}$$

Thus, we have that $\big(\mathcal{C}_{c,d}(\mathbb{F}_q),\otimes_{a,b,c,d}\big)$ is a commutative group (note that the associativity is derived from the associativity of the Brahamagupta product). When $c=1$ and the chosen identity point is $(a,b)=(1,0)$, the product $\otimes_{a,b,c,d}$ coincides with the classical Brahmagupta product $\otimes_d$ from Eq. (1).

In the following theorem we find the explicit isomorphism between $\mathcal{C}_d(\mathbb{F}_q)$ and $\mathcal{C}_{c,d}(\mathbb{F}_q)$. Then, thanks to this map, we can also write the isomorphism between generalized Pell hyperbolas.

Theorem 1. *Given $c,d\in\mathbb{F}_q^\times$ and a point $(a,b)\in\mathcal{C}_{c,d}(\mathbb{F}_q)$, the following map*

$$\begin{aligned}\varphi_{c,d}^{a,b} : \big(\mathcal{C}_d(\mathbb{F}_q),\otimes_d\big) &\xrightarrow{\sim} \big(\mathcal{C}_{c,d}(\mathbb{F}_q),\otimes_{a,b,c,d}\big),\\ (x,y) &\longmapsto (a,b)\otimes_d (x,y),\end{aligned}$$

is a group isomorphism whose inverse is

$$\begin{aligned}(\varphi_{c,d}^{a,b})^{-1} : \big(\mathcal{C}_{c,d},\otimes_{a,b,c,d}\big) &\xrightarrow{\sim} \big(\mathcal{C}_d,\otimes_d\big),\\ (x,y) &\longmapsto (1,0)\otimes_{a,b,c,d}(x,y).\end{aligned}$$

Proof. First of all, let us observe that $\varphi_{c,d}^{a,b}(\mathcal{C}_d(\mathbb{F}_q))\subseteq\mathcal{C}_{c,d}(\mathbb{F}_q)$ since, for any $(x,y)\in\mathcal{C}_d(\mathbb{F}_q)$,

$$\mathcal{N}_d\big((a,b)\otimes_d (x,y)\big) = \mathcal{N}_d(a,b)\cdot\mathcal{N}_d(x,y) = c.$$

Moreover, for any $(x_1, y_1), (x_2, y_2) \in \mathcal{C}_d(\mathbb{F}_q)$, we have

$$\begin{aligned}
\varphi_{c,d}^{a,b}\big((x_1, y_1) \otimes_d (x_2, y_2)\big) &= (a, b) \otimes_d (x_1, y_1) \otimes_d (x_2, y_2) \\
&= \frac{(a, -b) \otimes_d (a, b)}{c} \otimes_d (a, b) \otimes_d (x_1, y_1) \otimes_d (x_2, y_2) \\
&= \frac{1}{c}(a, -b) \otimes_d ((a, b) \otimes_d (x_1, y_1)) \otimes_d ((a, b) \otimes_d (x_2, y_2)) \\
&= \varphi_{c,d}^{a,b}(x_1, y_1) \otimes_{a,b,c,d} \varphi_{c,d}^{a,b}(x_2, y_2).
\end{aligned}$$

Thus, the map $\varphi_{c,d}^{a,b}$ is a well defined group homomorphism. Finally, we obtain that it is bijective since, for any $(x, y) \in \mathcal{C}_d(\mathbb{F}_q)$, we have

$$\varphi_{c,d}^{a,b}(x, y) = (a, b) \Leftrightarrow (x, y) = (1, 0),$$

and, given any $(w, z) \in \mathcal{C}_{c,d}(\mathbb{F}_q)$, its pre–image is $(1, 0) \otimes_{a,b,c,d} (w, z) \in \mathcal{C}_d(\mathbb{F}_q)$:

$$\begin{aligned}
\varphi_{c,d}^{a,b}\big((1, 0) \otimes_{a,b,c,d} (w, z)\big) &= \varphi_{c,d}^{a,b}\left(\frac{1}{c}(a, -b) \otimes_d (1, 0) \otimes_d (w, z)\right) \\
&= \frac{1}{c}(a, b) \otimes_d (a, -b) \otimes_d (1, 0) \otimes_d (w, z) \\
&= \frac{(a, b) \otimes_d (a, -b)}{c} \otimes_d (w, z) = (w, z),
\end{aligned}$$

where $(1, 0) \otimes_{a,b,c,d} (w, z) \in \mathcal{C}_d(\mathbb{F}_q)$ since it has unitary norm. □

We can obtain the explicit isomorphism between two generalized Pell hyperbolas with same parameter d by applying $(\varphi_{c,d}^{a,b})^{-1}$ and $\varphi_{c',d}^{a',b'}$. This isomorphism can be explicitly written, after some calculations, in the following form:

$$\begin{aligned}
\big(\mathcal{C}_{c,d}(\mathbb{F}_q), \otimes_{a,b,c,d}\big) &\xrightarrow{\sim} \big(\mathcal{C}_{c',d}(\mathbb{F}_q), \otimes_{a',b',c',d}\big), \\
(x, y) &\longmapsto (a', b') \otimes_{a,b,c,d} (x, y).
\end{aligned} \tag{4}$$

Finally, if $\big(\mathcal{C}_{c,d}(\mathbb{F}_q), \otimes_{a,b,c,d}\big)$ and $\big(\mathcal{C}_{c',d'}(\mathbb{F}_q), \otimes_{a',b',c'd'}\big)$ have $\chi_q(d) = \chi_q(d')$ and $d' = ds^2$, then the composition of $(\varphi_{c,d}^{a,b})^{-1}$, $\delta_{d,d'}$ and $\varphi_{c',d'}^{a',b'}$ is a group isomorphism between the two generalized Pell hyperbolas given explicitly by

$$\begin{aligned}
(\varphi_{c',d'}^{a',b'} \circ \delta_{d,d'} \circ (\varphi_{c,d}^{a,b})^{-1})(x, y) = \frac{1}{c}\big(&a'(ax - dby) + d'b'(ay - bx)/s, \\
&a'(ay - bx)/s + b'(ax - dby)\big).
\end{aligned}$$

From a computational point of view, it is not useful to exploit a generalized Pell hyperbola in a DLP–based cryptosystem since the security would remain unchanged while the computational costs would arise. However, in Section 5, we will exploit the isomorphism between Pell hyperbolas with different d but same $\chi_q(d)$ to obtain an alternative PKE scheme based on the ElGamal scheme.

3 Parameterization

In this section, we describe and study a parameterization for Pell hyperbolas with both an algebraic and a geometrical interpretation.

When considering the following quotient

$$\mathcal{R}_{d,q} = \mathbb{F}_q[t]/(t^2 - d) = \big\{x + ty \,|\, x, y \in \mathbb{F}_q,\, t^2 = d\big\},$$

for any two elements $x_1 + ty_1, x_2 + ty_2 \in \mathcal{R}_{d,q}$, the product naturally induced is

$$(x_1 + ty_1)(x_2 + ty_2) = (x_1x_2 + dy_1y_2) + t(x_1y_2 + y_1x_2),$$

which is essentially the classical Brahmagupta product. In order to introduce a parameterization for $\mathcal{C}_d(\mathbb{F}_q)$, we denote the invertible elements of $\mathcal{R}_{d,q}$ as $\mathcal{R}_{d,q}^{\otimes_d}$, so that there are two possible cases:

1. if $d \in \mathbb{F}_q^\times$ is a non–square, then

$$\mathcal{R}_{d,q}^{\otimes_d} = \mathcal{R}_{d,q} \smallsetminus \{0\};$$

2. if $d \in \mathbb{F}_q^\times$ is a square and $s \in \mathbb{F}_q^\times$ is a square root of d, then there is the decomposition $t^2 - d = (t - s)(t + s)$, so that

$$\mathcal{R}_{d,q}^{\otimes_d} = \mathcal{R}_{d,q} \smallsetminus \{0, \pm sy + yt \,|\, y \in \mathbb{F}_{\shortparallel}\}.$$

Now, we can consider the quotient $\mathcal{P}_{d,q} = \mathcal{R}_{d,q}^{\otimes_d}/\mathbb{F}_q^\times$, which yields to a parameterization for the Pell hyperbola. The elements of $\mathcal{P}_{d,q}$ are the classes of equivalence of the elements $m + nt \in \mathcal{R}_{d,q}^{\otimes_d}$, i.e., they are

$$[m + nt] = \big\{\lambda(m + nt) \,|\, \lambda \in \mathbb{F}_q^\times\big\}.$$

Given the definition of $\mathcal{P}_{d,q}$, $m + nt$ is equivalent to $(m + nt)n^{-1}$, where n^{-1} is the inverse of n in $\mathbb{F}_q^\times$. Thus, when $n = 0$ we choose as canonical representative $[1]$ while, in the other cases, we can take $[mn^{-1} + t]$. In this way, we can write

$$\begin{aligned}\mathcal{P}_{d,q} &= \begin{cases}\big\{[a + t]|\, a \in \mathbb{F}_q\big\} \cup \{[1]\}, & \text{if } d \text{ is a non–square,}\\ \big\{[a + t] \,|\, a \in \mathbb{F}_q \smallsetminus \{\pm s\}\big\} \cup \{[1]\}, & \text{otherwise}\end{cases}\\ &\sim \begin{cases}\mathbb{F}_q \cup \{\alpha\}, & \text{if } d \text{ is a non–square,}\\ \mathbb{F}_q \smallsetminus \{\pm s\} \cup \{\alpha\}, & \text{otherwise,}\end{cases}\end{aligned} \tag{5}$$

where α denotes an element not in $\mathbb{F}_q$ and it can be seen as the point at infinity in the set of parameters. The product in $\mathcal{P}_{d,q}$, which we denote by $\odot_d$, is induced by the quotient and it is given by

$$[m_1 + t] \odot_d [m_2 + t] = [m_1m_2 + (m_1 + m_2)t + t^2] = [m_1m_2 + d + (m_1 + m_2)t],$$

since $t^2 = d$ in $\mathcal{R}_{d,q}$. Therefore, we can also write

$$[m_1 + t] \odot_d [m_2 + t] = \begin{cases} \left[\frac{m_1 m_2 + d}{m_1 + m_2} + t\right], & \text{if } m_1 + m_2 \neq 0, \\ [1], & \text{if } m_1 + m_2 = 0. \end{cases}$$

In the following, we will mainly use the second notation in (5) for denoting the elements of $\mathcal{P}_{d,q}$ and in this case we can write the product as

$$m_1 \odot_d m_2 = \begin{cases} m_1, & \text{if } m_2 = \alpha, \\ m_2, & \text{if } m_1 = \alpha, \\ \frac{m_1 m_2 + d}{m_1 + m_2}, & \text{if } m_1 + m_2 \neq 0, \\ \alpha, & \text{otherwise.} \end{cases} \tag{6}$$

In the following theorem, we prove that $\mathcal{P}_{d,q}$ is isomorphic to the classical Pell hyperbola and thus it can be used as a parameterization for it.

Theorem 2. *Given $d \in \mathbb{F}_q^\times$, the following map is a group isomorphism*

$$\begin{aligned} \pi_d : \big(\mathcal{P}_{d,q}, \odot_d\big) &\xrightarrow{\sim} \big(\mathcal{C}_d(\mathbb{F}_q), \otimes_d\big), \\ m &\longmapsto \frac{(m,1)^{\otimes_d 2}}{\mathcal{N}_d(m,1)} = \left(\frac{m^2 + d}{m^2 - d}, \frac{2m}{m^2 - d}\right), \\ \alpha &\longmapsto (1,0), \end{aligned}$$

and the inverse is

$$\begin{aligned} \pi_d^{-1} : \big(\mathcal{C}_d(\mathbb{F}_q), \otimes_d\big) &\xrightarrow{\sim} \big(\mathcal{P}_{d,q}, \odot_d\big), \\ (1,0) &\longmapsto \alpha, \\ (-1,0) &\longmapsto 0, \\ (x,y) &\longmapsto \frac{x+1}{y}. \end{aligned}$$

Proof. It is immediate to check that the map is well–defined.

In addition, it is a group homomorphism since, for any $m_1, m_2 \in \mathcal{P}_{d,q}$, such that $m_1, m_2 \neq \alpha$ and $m_1 + m_2 \neq 0$, we have

$$\begin{aligned} \pi_d(m_1) \otimes_d \pi_d(m_2) &= \frac{(m_1,1)^{\otimes_d 2} \otimes_d (m_2,1)^{\otimes_d 2}}{\mathcal{N}_d(m_1,1)\mathcal{N}_d(m_2,1)} = \frac{(m_1 m_2 + d, m_1 + m_2)^{\otimes_d 2}}{(m_1^2 - d)(m_2^2 - d)} \\ &= \frac{\left(\frac{m_1 m_2 + d}{m_1 + m_2}, 1\right)^{\otimes_d 2}}{\frac{m_1^2 - d}{m_1 + m_2} \cdot \frac{m_2^2 - d}{m_1 + m_2}} = \frac{\left(\frac{m_1 m_2 + d}{m_1 + m_2}, 1\right)^{\otimes_d 2}}{\frac{(m_1 m_2 + d)^2}{(m_1 + m_2)^2} - d} \\ &= \frac{\left(\frac{m_1 m_2 + d}{m_1 + m_2}, 1\right)^{\otimes_d 2}}{\mathcal{N}_d\left(\frac{m_1 m_2 + d}{m_1 + m_2}, 1\right)} = \pi_d(m_1 \odot_d m_2). \end{aligned}$$

Similarly, it can be proved that the map is a group homomorphism for the remaining cases.

We can observe directly by the definition of π_d that $\alpha \in \ker(\pi_d)$. Moreover $\left(\frac{m^2+d}{m^2-d}, \frac{2m}{m^2-d}\right) \neq (1,0)$ for any $m \in \mathbb{F}_q$, so that $\ker \pi_d = \{\alpha\}$ and the map is injective. Then, the surjectivity follows directly from $|\mathcal{C}_d(\mathbb{F}_q)| = |\mathcal{P}_{d,q}|$.

Finally, it is straightforward to see that the pre–image of $(1,0)$ and $(-1,0)$ are α and 0, respectively while, for $(x,y) = \left(\frac{m^2+d}{m^2-d}, \frac{2m}{m^2-d}\right) \in \mathcal{C}_d(\mathbb{F}_q)$ with $y \neq 0$, the pre–image is $m = \frac{x+1}{y} \in \mathcal{P}_{d,q}$. Indeed, we have

$$x + 1 = \frac{m^2+d}{m^2-d} + 1 = \frac{2m^2}{m^2-d} = my.$$

In conclusion, π_d is a group isomorphism whose inverse is explicitly given like in the statement. □

Remark 1. *Considering the formula of π_d^{-1} given in the statement of the previous theorem, the set of parameters $\mathcal{P}_{d,q}$ of the Pell hyperbola can be obtained considering the lines $y = \frac{1}{m}(x+1)$ for m varying in $\mathbb{F}_q$ or $m = \alpha$ (having the sense of the point at the infinity).*

Since the definition of $\left(\mathcal{P}_{d,q}, \odot_d\right)$ is independent of the choice of the identity point (a,b) and of the constant c, the parameterization can be adapted for generalized Pell hyperbolas, leading with a proof analogous to Theorem 2 to the group isomorphism

$$\begin{aligned} \pi^{a,b}_{c,d} : \left(\mathcal{P}_{d,q}, \odot_d\right) &\xrightarrow{\sim} \left(\mathcal{C}_{c,d}(\mathbb{F}_q), \otimes_{a,b,c,d}\right), \\ m &\longmapsto \begin{cases} \left(2\frac{am+bd}{m^2-d}m - a, 2\frac{am+bd}{m^2-d} + b\right), & \text{if } m \neq \alpha, \\ (a,b), & \text{otherwise,} \end{cases} \end{aligned}$$

with inverse

$$\begin{aligned} (\pi^{a,b}_{c,d})^{-1} : \left(\mathcal{C}_{c,d}(\mathbb{F}_q), \otimes_{a,b,c,d}\right) &\xrightarrow{\sim} \left(\mathcal{P}_{d,q}, \odot_d\right), \\ (x,y) &\longmapsto \begin{cases} \frac{x+a}{y-b}, & \text{if } y \neq b, \\ -\frac{bd}{a}, & \text{if } (x,y) = (-a,b), \\ \alpha, & \text{if } (x,y) = (a,b). \end{cases} \end{aligned} \tag{7}$$

This parameterization and its inverse can be used as an alternative way to obtain the isomorphism in Eq. (4).

From a geometrical point of view, the parameter m of a point (x,y) is the slope of the line through (x,y) and $(-a,b)$ written considering x variable with y. This is very interesting when related to the geometric interpretation of the Brahmagupta product, which can be introduced in a very similar way to the one of the elliptic curves. Indeed, given two points P and Q of an elliptic curve, their sum $P \oplus Q$ is obtained by considering the point R, intersection between the elliptic curve and the line through P and Q, so that $P \oplus Q$ is the intersection between the elliptic curve and the line through R and the identity point, that is the point at infinity. This construction works also considering two points P

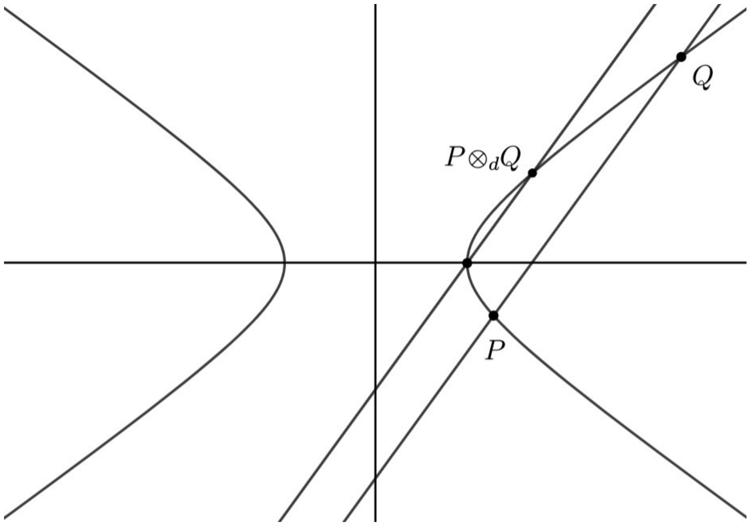

Figure 1: Geometric interpretation of the Brahmagupta product.

and Q of the Pell hyperbola, with the difference that the line through P and Q intersects the hyperbola at the point R that is, in this case, the point at infinity. Consequently, the product $P \otimes_d Q$ is the intersection between the hyperbola and the line through R (point at infinity) and the identity, that is, in the case of the classical Brahmagupta product, the point $(1, 0)$, i.e., the line parallel to the line through P and Q (see Fig. 1).

It is quite easy to check that, from this geometrical construction of $\otimes_d$, we obtain the algebraic expression described in Eq. (1), see, e.g., [26, pp. 231–232]. Indeed, given two points $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ on the Pell hyperbola, it is sufficient to check that the slope of the line through P and Q is equal to that of the line through the points $(x_1x_2 + dy_1y_2, x_1y_2 + y_1x_2)$ and $(1, 0)$. Thanks to this geometrical approach, we can also observe that the identity point can be an arbitrary point of the Pell hyperbola, and this choice leads to the generalized Brahmagupta product in Eq. (3).

4 Exponentiation with Rédei rational functions

In this section, we show an efficient algorithm for the exponentiation on the Pell hyperbola that exploits the parameterization introduced in the previous one. Moreover, we compare it with the square–multiply algorithm with the classical Brahmagupta product.

When dealing with cryptosystems whose security is based on the DLP, the computational bottleneck is the evaluation of the exponentiation, which is usually implemented with a square–multiply algorithm, eventually enhanced with

```
Brahmagupta($x_P, y_P, e, d, q$):
(x, y) = (1, 0)
$bin\_e$ = binary($e$)
for $bit$ in $bin\_e$ do
    $x = x^2 + dy^2 \in \mathbb{F}_q$
    $y = 2xy \in \mathbb{F}_q$
    if $bit = 1$ then
        $x = xx_P + dyy_P \in \mathbb{F}_q$
        $y = xy_P + yx_P \in \mathbb{F}_q$
    end if
end for
return $(x, y)$
```

```
Modified_More($m, e, d, q$):
$N, D = 1, 0$
$bin\_e$ = binary($e$)
for $bit$ in $bin\_e$ do
    $N = N^2 + dD^2 \in \mathbb{F}_q$
    $D = 2ND \in \mathbb{F}_q$
    if $bit = 1$ then
        $N = Nm + dD \in \mathbb{F}_q$
        $D = N + Dm \in \mathbb{F}_q$
    end if
end for
return $N/D \in \mathbb{F}_q$
```

Figure 2: square–multiply algorithm with the Brahmagupta product (left) and modified More algorithm for Rédei polynomials (right).

a pre-computation phase. Therefore the total time is determined by the speed of the single product, which is required in both square and multiply steps.

The Brahmagupta product described explicitly in (1) requires 5 products and 2 additions, while the product on the parameters obtained in (6) requires 1 inversion, 2 products and 2 additions. The inversion is largely more expensive than the additional 3 products required in the Brahmagupta product. Therefore, in a comparison of square–multiply implementations, the first one is the most efficient. However, when d is a non–square, the product (6) can be evaluated exploiting the Rédei rational functions. They are introduced over the real numbers by means of the powers

$$(m + \sqrt{d})^n = A_n(m, d) + B_n(m, d)\sqrt{d},$$

which define two sequences of polynomials whose ratios provide the Rédei rational functions

$$Q_n(m, d) = \frac{A_n(m, d)}{B_n(m, d)},$$

with $m, d \in \mathbb{Z} \setminus \{0\}$ and d non–square. Observing that

$$Q_1(m, d) = m \quad \text{and} \quad Q_{n_1+n_2}(m, d) = Q_{n_1}(m, d) \odot_d Q_{n_2}(m, d),$$

we have

$$m^{\odot_d e} = Q_e(m, d).$$

Clearly, the above definition can be easily adapted over finite fields. In [19], the author proposed an algorithm for evaluating the Rédei rational function $Q_n(m, d)$ with complexity $O(\log n)$ considering additions and multiplications over a ring, and in [5], the authors improved the performance of this algorithm. The obtained algorithm is detailed in Fig. 2, where it is compared with the

```
KeyGen($n$):
  1: $q \leftarrow_{\$} \{0,1\}^n$ power of a prime
  2: $d \leftarrow_{\$} \mathbb{F}_q$ with $\chi_q(d) = -1$
  3: $G \leftarrow_{\$} \mathcal{C}_d(\mathbb{F}_q)$ of order $q+1$
  4: $sk \leftarrow_{\$} \{2, \dots, q\}$
  5: $H = G^{\otimes_d sk} \in \mathcal{C}_d(\mathbb{F}_q)$
  6: $pk = (q, d, G, H)$
  7: return $pk, sk$
```

```
Encrypt($msg, pk$):
Require: $msg < q$
  1: $y \leftarrow msg$
  2: $x = \sqrt{1 + d\,y^2} \in \mathbb{F}_q$
  3: $r \leftarrow_{\$} \{2, \dots, q\}$
  4: $C_1 = G^{\otimes_d r} \in \mathcal{C}_d(\mathbb{F}_q)$
  5: $C_2 = H^{\otimes_d r} \otimes_d (x, y) \in \mathcal{C}_d(\mathbb{F}_q)$
  6: return $C_1, C_2$
Decrypt($C_1, C_2, pk, sk$):
  1: $(x, y) = (C_1^{\otimes_d sk})^{-1} \otimes_d C_2 \in \mathcal{C}_d(\mathbb{F}_q)$
  2: $msg \leftarrow y$
  3: return $msg$
```

Figure 3: ElGamal PKE scheme with the cyclic group $\big(\mathcal{C}_d(\mathbb{Z}_q), \otimes_d\big)$ of order $q+1$. Exponentiations are realized with the square–multiply algorithm with the Brahmagupta product.

square–multiply algorithm with the classical Brahmagupta product. In particular, the two algorithms share the quantity of operations at each step except for steps 7/8, where the Brahmagupta version requires an additional product, and step 11, where the modified More algorithm requires a final inversion.

Thus, from an efficiency point of view, the two algorithms are comparable. The main advantage in adopting the parameterization and the modified More algorithm is that the size of the data is halved because they are elements of $\mathbb{F}_q$ and not of $\mathcal{C}_d(\mathbb{F}_q)$.

5 Public-key encryption with the Pell hyperbola

In this section, we present and compare three different PKE schemes based on the ElGamal scheme with Pell hyperbolas over a finite field $\mathbb{F}_q$. In particular, we exploit the group on the Pell hyperbola $\mathcal{C}_d(\mathbb{F}_q)$ with d non–square and the Brahmagupta product with $(1, 0)$ as identity, as well as the parameterization presented in Section 3. In the following, it is useful to take $q = p$ prime and $p = 2p' - 1$ with p' prime in order to avoid small subgroups.

5.1 ElGamal with the Pell hyperbola

The first algorithm is detailed in Fig. 3. Given q power of a prime n bits long (step 1), the order of the cyclic group is $q - \chi_q(d) = q + 1$. After choosing, in step 2, $d \in \mathbb{F}_q$, a random generator G of $\mathcal{C}_d(\mathbb{F}_q)$ is taken in step 3. Then the algorithm proceeds by taking a random exponent sk (step 4) and obtaining a public point $H \in \mathcal{C}_d(\mathbb{F}_q)$ through the square–multiply algorithm with the Brahmagupta product (step 5). The public key contains q, d and the points G and H, while the secret key is the exponent sk used to obtain H from G. In the

```
KeyGen($n$):
  1: $q \leftarrow_\$ \{0,1\}^n$ power of a prime
  2: $d \leftarrow_\$ \mathbb{F}_q$ with $\chi_q(d) = -1$
  3: $g \leftarrow_\$ \mathcal{P}_{d,q}$ of order $q+1$
  4: $sk \leftarrow_\$ \{2,\ldots,q\}$
  5: $h = g^{\odot_d sk} \in \mathcal{P}_{d,q}$
  6: $pk = (q,d,g,h)$
  7: return $pk, sk$
```

```
Encrypt($msg, pk$):
Require: $msg < q$
  1: $r \leftarrow_\$ \{2,\ldots,q\}$
  2: $c_1 = g^{\odot_d r} \in \mathcal{P}_{d,q}$
  3: $c_2 = h^{\odot_d r} \odot_d msg \in \mathcal{P}_{d,q}$
  4: return $c_1, c_2$
Decrypt($c_1, c_2, pk, sk$):
  1: $msg = -c_1^{\odot_d sk} \odot_d c_2 \in \mathcal{P}_{d,q}$
  2: return $msg$
```

Figure 4: ElGamal PKE scheme with the cyclic group $\left(\mathcal{P}_{d,q}, \odot_d\right)$ of order $q+1$ of the parameters of $\mathcal{C}_d(\mathbb{F}_q)$. Exponentiations are realized with the modified More algorithm.

first step of the encryption algorithm, the message determines the y coordinate of a point, while in the second step the corresponding x is chosen under the condition $(x, y) \in \mathcal{C}_d(\mathbb{F}_q)$. Since such a point could not exist, some bits of y can be kept variable by reducing the maximum length of the message. The ciphertext consists of two points C_1 and C_2. After taking a random exponent r in step 3, it is used in step 4 to obtain C_1 through the exponentiation with the Brahmagupta product with base the public generator G. In step 5, the point C_2 is determined as the Brahmagupta product of $H^{\otimes_d r}$ with the point (x, y) representing the message. During the decryption, the point (x, y) is retrieved as the Brahmagupta product of the inverse of $C_1^{\otimes_d sk}$ with C_2 (step 1). From the obtained y coordinate, the original message is recovered in step 2. In particular, this implementation is formally equivalent to the ElGamal PKE scheme with a cyclic subgroup of order $q+1$ of $\mathbb{F}_{q^2}$ [17].

5.2 ElGamal with the group of parameters

The second algorithm is described in Fig. 4 and consists of the ElGamal PKE scheme with the cyclic group $\left(\mathcal{P}_{d,q}, \odot_d\right)$. A power of a prime q of n bits is taken in step 1 and the order of the cyclic group is still $q - \chi_q(d) = q + 1$. In step 2, a random non–square $d \in \mathbb{F}_q$ is taken. After choosing a generator $g \in \mathcal{P}_{d,q}$ in step 3 and a random exponent sk in step 4, a parameter $h = g^{\odot_d sk}$ is evaluated in step 5 with the modified More algorithm. The public key consists of q, d and the parameters g, h, while the secret key is the exponent sk. The encryption considers the message as a parameter $msg \in \mathcal{P}_{d,q}$. Step 1 takes a random exponent r, which is used in step 2 to obtain the parameter c_1 through the modified More algorithm for the exponentiation. The second parameter c_2 is the result of the parameter product between $h^{\odot_d r}$ and msg. Finally, the ciphertext is the pair of parameters (c_1, c_2) and consequently it requires half of the space than in the previous algorithm. The decryption is straightforward. It retrieves the message as the parameter product between the inverse of $c_1^{\odot_d sk}$ (which is simply

```
KeyGen($n$):
  1: $q \leftarrow_{\$} \{0,1\}^n$ power of a prime
  2: $d \in \mathbb{F}_q$ minimum with $\chi_q(d) = -1$
  3: $g \leftarrow_{\$} \mathcal{P}_{d,q}$ of order $q+1$
  4: $sk \leftarrow_{\$} \{2, \dots, q\}$
  5: $h = g^{\odot_d sk} \in \mathcal{P}_{d,q}$
  6: $pk = (q, d, g, h)$
  7: return $pk, sk$
```

```
Encrypt($msg, pk$):
Require: $msg \leq (q-1)^2$
  1: $(x, y) \leftarrow msg$
  2: $d' = \frac{x^2-1}{y^2} \in \mathbb{F}_q$ with $\chi_q(d') = -1$
  3: $m = \pi_{d'}^{-1}(x, y) = \frac{x+1}{y} \in \mathcal{P}_{d',q}$
  4: $r \leftarrow_{\$} \{2, \dots, q\}$
  5: $s = \sqrt{d'/d} \in \mathbb{F}_q$
  6: $c_1 = (gs)^{\odot_{d'} r} \in \mathcal{P}_{d',q}$
  7: $c_2 = (hs)^{\odot_{d'} r} \odot_{d'} m \in \mathcal{P}_{d',q}$
  8: return $c_1, c_2, d'$
```

```
Decrypt($c_1, c_2, d', pk, sk$):
  1: $m = (-c_1^{\odot_{d'} sk}) \odot_{d'} c_2$
  2: $msg \leftarrow \pi_{d'}(m) = \left(\frac{m^2+d'}{m^2-d'}, \frac{2m}{m^2-d'}\right)$
  3: return $msg$
```

Figure 5: Alternative ElGamal PKE scheme with the cyclic group $(\mathcal{P}_{d',q}, \odot_{d'})$ of order $q+1$ and d' part of the ciphertext. Exponentiations are realized with the modified More algorithm.

its opposite) and c_2. Because of the comparison between the exponentiation algorithms in Section 4, the computational time is comparable with that of the ElGamal scheme with the points of the Pell hyperbola. However, the public key and the ciphertext require less space because they contain parameters instead of coordinates.

5.3 ElGamal with the isomorphisms

The third algorithm is an alternative version of ElGamal PKE scheme with the use of the parameters. The differences are due to the exploitation of the explicit isomorphisms between Pell hyperbolas with different d. The algorithms are described in Fig. 5. The key generation is analogous to the previous one, except for the smallest non–square d taken in step 3, which is used for the exponentiation in step 6 and then included in the public key. The main differences are in the encryption: the maximum length of the message can be doubled with respect to the previous algorithms, because it is used in step 1 to obtain the coordinates of a point $(x, y) \in \mathbb{F}_q \times \mathbb{F}_q$, e.g., by mapping the first half of msg in x and the second half in y. From this point, step 2 searches for a non–square $d' \in \mathbb{F}_q$ such that $(x, y) \in \mathcal{C}_d(\mathbb{F}_q)$. If necessary, some of the bits of x can be kept variable so that such a d' can be found. Then, in step 3, the parameter m related to the point is obtained through the parameterization from Eq. (7). In step 4 a random exponent r is chosen. Now, since the public key contains parameters of points of $\mathcal{C}_d(\mathbb{F}_q)$, the isomorphism between Pell hyperbolas $\delta_{d,d'}$ is exploited. In particular, the version between the groups of parameters is required. Thus,

Sec.	FFC	ECC	$\mathcal{C}_d(\mathbb{F}_q)$	$\mathcal{P}_{d,q}$	$\pi_{d'}, \delta_{d,d'}$
80	1024	160	512	512	512
112	2048	224	1024	1024	1024
128	3072	256	1536	1536	1536
192	7680	384	3840	3840	3840
256	15360	512	7680	7680	7680

Table 1: Field size in bits for different DLP–based cryptosystems depending on the cyclic group and the classical security strength in bits.

supposing $d' = ds^2$, $\delta_{d,d'}(x, y) = (x, y/s)$ gives the explicit group isomorphism

$$\begin{aligned} \big(\mathcal{P}_{d,q}, \odot_d\big) &\xrightarrow{\sim} \big(\mathcal{P}_{d',q}, \odot_{d'}\big), \\ m &\longmapsto ms, \end{aligned} \tag{8}$$

which is used to obtain gs and hs to be used for the exponentiations with $\odot_{d'}$. In step 5, the factor $s = \sqrt{d'/d} \in \mathbb{F}_q$ can always be evaluated because d' and $1/d$ are non–squares and their product is a square. Steps 6-7 evaluate the parameters c_1, c_2 as in the previous algorithm but with the basis obtained from the isomorphism with the factor s. The ciphertext contains c_1, c_2 and also the non–square d' used in the calculations. The setting of the decryption is analogous to the previous case but, after evaluating the product between the inverse of $c_1^{\odot_{d'} sk}$ and c_2 (step 1), the message must be retrieved from the point related to the obtained parameter (step 2). The main advantage in adopting this algorithm is the doubled length of the message with a cost of less than double operations and also the decreased keys and ciphertext sizes.

6 Security and performance

In this section some results about security, data size and computational costs are presented.

6.1 Security

Since all the introduced cryptosystems are variants of the classical ElGamal, security against chosen–plaintext and chosen–ciphertext attacks is achieved as in the standard scheme. However, they also remain insecure against adaptive chosen–ciphertext attacks, so that in these scenarios a padding of the message is required.

Looking at the security strength for the DLP–based cryptosystems, it relies on the adopted cyclic group. In particular, since in all the introduced schemes the parameter $d \in \mathbb{F}_q$ is a non–square, there is an explicit group isomorphism between $\big(\mathcal{C}_d(\mathbb{F}_q), \otimes_d\big)$ and the multiplicative subgroup $G \subset \mathbb{F}_{q^2}^{\times}$ of order $q + 1$ [17]. In addition, this is true also for $\big(\mathcal{P}_{d,q}, \odot_d\big)$ through π_d. Thus, the DLP related to the Pell hyperbola can be reduced to that in a finite field that, with

Formulation	par	pk	sk	msg	c_1, c_2
FFC	$2n$	n	n	n	$2n$
	2048	1024	1024	1024	2048
ECC	$6n$	$2n$	n	n	$4n$
	960	320	160	160	640
$\mathcal{C}_d(\mathbb{F}_q)$	$4n$	$2n$	n	n	$4n$
	2048	1024	512	512	2048
$\mathcal{P}_{d,q}$	$3n$	n	n	n	$2n$
	1536	512	512	512	1024
$\pi_{d'}, \delta_{d,d'}$	$2n$	n	n	$2n$	$3n$
	1024	512	512	1024	1536

Table 2: Data–size in bits for ElGamal with FFC, ECC, $\mathcal{C}_d(\mathbb{F}_q)$, $\mathcal{P}_{d,q}$ and the alternative formulation, depending on the size n of q and for 80 bits of security.

respect to the standard security strengths from [2] for ElGamal in Finite Field Cryptography (FFC), has halved size of q.

The comparison of the size of q for FFC, ECC and the introduced cryptosystems based on the Pell hyperbola is detailed in Table 1, where the fourth column refers to the formulation in Fig. 3, the fifth to Fig. 4 and the last to Fig. 5. Despite the sizes of the fields in Pell–based cryptosystems are halved with respect to those in FFC, the sizes in ECC still remain the smallest. Given these security levels, it is possible to compare the proposed cryptosystems with the classical ones in terms of data–size and performance.

6.2 Data–size

Table 2 collects the size of the data involved in the various ElGamal formulations. In particular, the public key is divided in two parts: the public parameters, i.e., the data required for the description of the cyclic group and one of its generators, that can be used by different users, and the actual public key (the element h or the point H). The other collected values are the size of the secret key, the maximum message length and the bit–length of the ciphertext (that for ElGamal is a pair of elements or points). The considered formulations are, in order from top to bottom, the classical ElGamal scheme in FFC and ECC, and the three cryptosystems based on the Pell hyperbola and the group of its parameters. For each case, the table shows the data–size depending on the size n of the cardinality q of the related finite field (taken from Table 1) and the values for 80 bits of security strength.

The formulation with $\mathcal{C}_d(\mathbb{F}_q)$ in the third row has the same size of that in FFC (first row) in terms of parameters, public key and ciphertext, but the maximum message length is halved, so that in a fair comparison its encryption and decryption should be run twice, and the ciphertext length becomes the double of that in the first row. Despite this drawback, a performance comparison could be interesting since q has still halved size with respect to FFC.

Looking at the fourth row, i.e, at the formulation with $\mathcal{P}_{d,q}$, all the sizes are half of those for FFC, except for the bit–length of the parameters which is still smaller. Again, when fixing the same message length, two runs of `Enc` and `Dec` are required so that the size of the ciphertext is doubled and becomes equal to that in the first row. However, with respect to the previous formulation, the public key has half the size and calculations are still faster than in FFC since q is smaller.

Finally, when comparing the formulation in the fifth row with the classical FFC, its parameters and keys require half the bits, and the ciphertext is smaller for the same maximum message length.

Despite the formulation in ECC (second row) is competitive for its smallest data, when fixing the maximum message length, its parameters and keys maintain the smallest size, but the ciphertext length grows as that of the formulation with $\mathcal{C}_d(\mathbb{F}_q)$. In particular, the ratio with the length of the message is 4, while in the first and fourth row the ratio is 2 and the best value is 1.5 corresponding to the last proposal.

In conclusion, the formulation in the last row is the best proposal in terms of information encrypted.

6.3 Performance

The last study concerns the performance of the ElGamal formulations and consists in collecting the elapsed times of a simple implementation in Python of each of the algorithms run on the cluster of the DISMA at Politecnico of Turin, on a single CPU with $46G$ of RAM allocated. For each level of security strength and cryptosystem, the times shown in Table 3 are the means of 10 randomly generated instances, whose times are taken as the minimum of 10 identical runs. The formulations are compared for different security strengths each with maximum message length fixed. Following considerations from the analysis on the data–size, this results in repeating encryption and decryption k times for ECC, where k is the ratio between the maximum message length for FFC and ECC (e.g., for 80 bits of security, $1024/160 = 6.4$ so that $k = 7$), and 2 times when using directly $\mathcal{C}_d(\mathbb{F}_q)$ or $\mathcal{P}_{d,q}$. When working with a finite field $\mathbb{F}_q$, the case with q prime is considered, and the bit–length n of q depends on the standard security strengths, assuming the values obtained in Table 1.

The times for the key generation algorithm do not consider the generation of the public parameters, i.e., the cyclic group and one of its generators, e.g., steps 1-3 in Fig. 3, Fig. 4 and Fig. 5. This because they can be pre–computed and used by different users, so that the collected times consider only the generation of private and public key, i.e., a single exponentiation.

Columns 3 and 5 show that the key generation performs similarly for FFC and ElGamal with $\mathcal{C}_d(\mathbb{F}_q)$, despite the latter is generally a bit faster. The formulation with $\mathcal{P}_{d,q}$ in the sixth column is generally at the third place, beaten by the version that exploits the change of parameter d described in Fig. 5 (column 7). ECC in the fourth column is less efficient at lower security strengths but, thanks to the good scalability of elliptic curves, works better than any other formulation starting from the third level of security strength (128 bits).

Sec.	Alg.	FFC	ECC	$\mathcal{C}_d(\mathbb{F}_q) \times 2$	$\mathcal{P}_{d,q} \times 2$	$\pi_{d'}, \delta_{d,d'}$
80	`Gen`	0.011079	0.028271	0.011713	0.009781	0.007524
	`Enc`	0.022311	0.393407	0.059983	0.040459	0.028152
	`Dec`	0.012183	0.194531	0.023631	0.020472	0.010203
112	`Gen`	0.074718	0.056586	0.073778	0.056865	0.038527
	`Enc`	0.149400	1.194561	0.364686	0.229299	0.164122
	`Dec`	0.077622	0.567866	0.148194	0.115962	0.057106
128	`Gen`	0.233983	0.075437	0.227347	0.171958	0.112873
	`Enc`	0.467730	1.818186	1.103675	0.689103	0.496599
	`Dec`	0.239429	0.903710	0.454805	0.347872	0.171190
192	`Gen`	3.188959	0.185410	2.811594	2.127992	1.372381
	`Enc`	6.372422	7.454103	13.791595	8.525471	6.291258
	`Dec`	3.218019	3.718247	5.630895	4.273549	2.103753
256	`Gen`	22.874051	0.365562	18.155630	13.841428	9.519104
	`Enc`	45.766954	22.052779	87.457496	55.563741	42.658508
	`Dec`	22.981310	10.965318	36.287580	27.792128	14.464945

Table 3: Average times in seconds for 10 random instances of ElGamal with FFC, ECC, with $\mathcal{C}_d(\mathbb{F}_q)$, $\mathcal{P}_{d,q}$ and the alternative formulation, for fixed message length, depending on the security strength.

This advantage is a bit attenuated for encryption and decryption: in particular, ECC is the less efficient formulation for the first four levels and becomes comparable to the others for 128 bits of security. However, for the highest level, the good scalability of elliptic curves allows to increase the efficiency of the two algorithms, which become the fastest. Among the others, ElGamal with $\mathcal{C}_d(\mathbb{F}_q)$ is generally the worst option, followed by the formulation with $\mathcal{P}_{d,q}$. For the first four levels of security, the formulations in FFC and from Fig. 5 are the most efficient, with encryption slightly better for the former and decryption more efficient for the latter.

In conclusion, the first two new cryptosystems remain interesting from the theoretical point of view and they could be studied further to obtain better performances. On the other hand, the new ElGamal formulation that exploits $\pi_{d'}$ and $\delta_{d,d'}$ introduced in Fig. 5 seems to be a very powerful alternative for DLP–based cryptosystems: its performance is comparable with the classical ElGamal in FFC for all security levels and also with the ECC version up to the fourth level of security, while it maintains the highest ratio of message/ciphertext length.

References

[1] E. J. Barbeau, Pell's Equation, Springer–Verlag, New York - Berlin, 2003.

[2] E. Barker, SP 800-57 Part 1: Recommendation for Key Management, NIST, 2020.

[3] E. Bellini, N. Murru, An efficient and secure RSA-like cryptosystem exploiting Rédei rational functions over hyperbolas, Finite Fields and their Applications, Vol. 39, 179–194, 2016.

[4] E. Bellini, N. Murru, A multifactor RSA-like scheme with fast decryption based on Rédei rational functions over the Pell hyperbola, Lecture Notes in Computer Science, Vol. 11973, 343–357, 2020.

[5] E. Bellini, N. Murru, A.J. Di Scala, M. Elia, Group law on affine hyperbolas and applications to cryptography, Applied Mathematics and Computation, Vol. 409, Article 125537, 2021.

[6] R. M. Daniel, E. B. Rajsingh, S. Silas, A forward secure signcryption scheme with ciphertext authentication for e-payment systems using hyperbola curve cryptography, Journal of King Saud University, Vol. 33, 86–98, 2021.

[7] T. ElGamal, A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, IEEE Transactions on Information Theory, Vol. 31, 469–472, 1985.

[8] H. I. Hussein, W. M. Abduallah, An efficient ElGamal cryptosystem scheme, International Journal of Computers and Applications, Vol. 43, 1088–1094, 2021.

[9] S. Inam, R. Ali, A new ElGamal-like cryptosystem based on matrices over groupring, Neural Comput. and Appl., Vol. 29, 1279–1283, 2018.

[10] K. Kamthawee, B. Chiewthanakul, The Construction of ElGamal over Koblitz Curve, Advanced Materials Research, Vol 931–932, 1441–1446, 2014.

[11] O. Khadir, New Variant of ElGamal Signature Scheme, Int. J. Contemp. Math. Sciences, Vol. 5, 1653–1662, 2010.

[12] N. Koblitz, Elliptic curve cryptosystem, Mathematics of Computation, Vol. 48, 203–209, 1987.

[13] F. Lemmermeyer, Introduction to Cryptography, 2006.

[14] A. Mahalanobis, A Simple Generalization of the ElGamal Cryptosystem to Non-Abelian Groups, Communications in Algebra, Vol. 36, 3878–3889, 2008.

[15] A. Mahalanobis, A Simple Generalization of the ElGamal Cryptosystem to Non-Abelian Groups II, Communications in Algebra, Vol. 40, 3583-3596, 2012.

[16] A. Mahalanobis, The MOR Cryptosystem And Extra-Special p-Groups, Journal of Discrete Mathematical Sciences and Cryptography, Vol. 18, 201–208, 2015.

[17] A. J. Menezes, S. A. Vanstone, A Note on Cyclic Groups, Finite Fields, and the Discrete Logarithm Problem, Applicable Algebra in Engineering, Communication and Computing, Vol. 3, 67–74, 1992.

[18] A. M. Mishra, A Digital Signature Scheme based on Pell Equation, Int. J. Of Innovative Research in Science Engineering and Technology, Vol. 3(1), pp. 8586–8591, 2014.

[19] W. More, Fast evaluation of Rédei functions, Appl. Algebra Engrg. Comm. Comput., Vol. 6, 171–173, 1995.

[20] N. R. Murthy, M.N.S. Swamy, Cryptographic applications of Brahmagupta–Bhaskara equation, IEEE Trans. Circuits Syst. I, Vol. 53, 1565—1571, 2006.

[21] S. Padhye, A public key cryptosystem based on Pell equation, IACR Cryptol. ePrint Arch. 2006 191, 2006.

[22] A. Pandey, I. Gupta, D. K. Singh, Improved cryptanalysis of a ElGamal Cryptosystem Based on Matrices Over Group Rings, Journal of Mathematical Cryptology, Ready Online at https://doi.org/10.1515/jmc-2019-0054, 2020.

[23] K. Rao, P. S. Avadhani, D. L. Bhaskari, K.V.S.S.R.S.S. Sarma, An identity based encryption scheme based on Pell's equation with Jacobi symbol, Int. J. Appl. Sci. Eng. Res., Vol. 1, 17–20, 2013.

[24] R. Ranasinghe, P. Athukorala, A generalization of the ElGamal public-key cryptosystem, Journal of Discrete Mathematical Sciences and Cryptography, Ready online at https://doi.org/10.1080/09720529.2020.1857902, 2021.

[25] C. Thirumalai, S. Mohan, G. Srivastava, An efficient public key secure scheme for cloud and IoT security, Computer Communications, Vol. 150, 634–643, 2020.

[26] O. Veblen, J. W. Young, Projective Geometry, Vol. I, Boston: Ginn & Co., 1938.