Research Article

Luigi Accardi and Massimo Regoli

On a class of strongly asymmetric PKA algorithms

Abstract: In the papers [1, 2] a new scheme to produce public key agreement (PKA) algorithms was proposed and some examples based on polynomials (toy models) were discussed. In the present paper we introduce a non-commutative realization of the above mentioned scheme and prove that non-commutativity can be an essential ingredient of security in the sense that, in the class of algorithms constructed, under some commutativity assumptions on the matrices involved, we can find a breaking strategy, but dropping these assumptions we can not, even if we assume, as we do in all the attacks discussed in the present paper, that discrete logarithms have zero cost.

Keywords: PKA algorithm, asymmetric algorithm

MSC 2010: 94A60

Luigi Accardi, Massimo Regoli: Centro Vito Volterra, Università di Roma Tor Vergata, Roma, Italy, e-mail: accardi@volterra.uniroma2.it, regoli@uniroma2.it

Communicated by: Kaoru Kurosawa

1 Strongly asymmetric algorithm-3 (SAA-3)

Public key agreement (PKA) algorithms play an important role in contemporary asymmetric cryptography. The term asymmetry here refers to the difference of information between sender and receiver: each of them ignores the secret key of the other one. However, in the most used asymmetric algorithms the operations executed by sender and receiver are very similar. The main idea of strongly asymmetric cryptography is to further break this symmetry introducing multiple secret and public keys for the sender. This allows to split public information into several pieces, thus making attacks more difficult.

In the present paper we will keep the distinction between public parameters and public keys even if in PKA algorithms this distinction is not sharp because any public parameter can become a public key of one of the interlocutors.

1.1 Public parameters

The public parameters of the algorithm are

- a natural integer \( d \in \mathbb{N} \),
- a large prime number \( p \in \mathbb{N} \),
- the finite field \( \mathbb{F} \) (typically \( \mathbb{F} := \mathbb{Z}_p \)),
- a fixed matrix \( \alpha \in M(d; \mathbb{F}) := \{d \times d \text{ square matrices with entries in } \mathbb{F}\} \),
- a finite set \( I \subset \mathbb{N} \).

In the following, all scalar multiplications (in particular exponentiations) are meant in \( \mathbb{F} \) and we use the convention

\[ 0^x := 0 \quad \text{for all } x \in \mathbb{F}. \]

The term matrix will be used as a synonym of element of \( M(d; \mathbb{F}) \); all matrix multiplications are meant in the standard sense while matrix exponentiations are meant in the Schur sense, i.e. elementwise: if \( c \) is either an
element of $\mathbb{F}$ or a matrix $c = (c_{ij})$ and $M = (M_{ab})$ is a matrix, the symbol $c^M$ denotes the matrix

$$(c^M)_{ab} := \begin{cases} c_{M_{ab}} & \text{scalar case}, \\ c_{M_{ab}} & \text{matrix case}, \end{cases} \quad a, b \in \{1, \ldots, d\}.$$

**Secret keys of $B$.** The main secret key of $B$ is

$$x_{B,3} \in M(d; \mathbb{F}).$$

Additional secret keys of $B$ are

$$\{A_j \in M(d; \mathbb{F}) : j \in I\}, \quad N_{B,3} \in M(d; \mathbb{F}), \quad c \in \mathbb{F}.$$

The conditions to be satisfied by these secret keys of $B$ are

1. \[ [x_{B,3}, \alpha] = 0, \]
2. \[ [x_{B,3}, N_{B,3}^{-1} \log y_A] = 0, \]
3. \[ [N_{B,3}^{-1} \log y_A, A_j] \neq 0 \quad \text{for all } j \in I, \]
4. \[ [\log y_{B,3}, N_{B,3}] \neq 0 \quad \text{for all } j \in I, \]
5. \[ [\log y_{B,3}, A_j] \neq 0 \quad \text{for all } j \in I, \]
6. \[ [x_{B,3}, A_j] \neq 0 \quad \text{for all } j \in I. \]

Here and in the following, if $x \equiv (x_{ij}) \in M(d; \mathbb{F})$, then $\log x$ denotes the Schur logarithm of $x$, i.e.

$$(\log x)_{ij} := \log x_{ij}.$$

Clearly, $N_{B,3}$ must be invertible and, for the reasons explained in Section 1.4, it is convenient to choose the $A_j$ non-invertible.

**Public keys of $B$.** The public keys of $B$ are given by the finite set of matrices

$$\{y_{B,3}, y_{B,3,j} \in M(d; \mathbb{F}) : j \in I\},$$

constructed using the secret keys of $B$, as follows. For all $j \in I$ and $a, b \in \{1, \ldots, d\}$ set

$$y_{B,3,a,b} := e^{(N_{B,3}A_j)_{ab}} = (e^{N_{B,3}A_j})_{ab},$$
$$y_{B,3,a,b} := e^{(A_j x_{B,3})_{ab}} = (e^{A_j x_{B,3}})_{ab}.$$

Notice that, once given the $(A_j)_{j \in I}$ and any matrix $x \in M(d; \mathbb{F})$, the polynomial $Q \equiv (A_j)_{j \in I}$,

$$Q_a(x^x) := \sum_{j \in I} A_j (a^x)^j,$$

of degree $|I|$ in the matrix $a^x$ is uniquely determined.

**Secret key of $A$.** A matrix $x_A \in M(d; \mathbb{F})$.

**Public key of $A$.** The public key of $A$ is given by the matrix $y_A := (y_{A,a,g}) \in M(d; \mathbb{F})$ constructed as follows. For each $a, g \in \{1, \ldots, d\}$ set

$$y_{A,a,g} = e^{[N_{B,3}Q_a a^x]_{ag}} = (e^{N_{B,3}Q_a a^x})_{ag}.$$

$y_A$ can be computed uniquely in terms of the public keys of $B$, the public parameter $\alpha$ and the secret key of $A$ as follows. For each $a, g \in \{1, \ldots, d\}$,

$$y_{A,a,g} = \prod_{j \in I} \prod_{a,b \in \{1, \ldots, d\}} (y_{B,3,a,b})^{|a^x|}_{ag} = \prod_{j \in I} \prod_{a,b \in \{1, \ldots, d\}} e^{[N_{B,3}A_j a^x]_{ag}} = e^{[N_{B,3}A_j a^x]_{ag}} = e^{[N_{B,3}Q_a a^x]_{ag}} = (e^{N_{B,3}Q_a a^x})_{ag},$$
SSK. The matrix
\[ \kappa_{a,g} := c^{(Q_{a}(\alpha^{x}x_{3})x_{2})_{a,g}} = (c^{Q_{a}(\alpha^{x}x_{3})})_{a,g}, \quad a, g \in \{1, \ldots, d\}. \]

\( B \) computes the SSK using the public key of \( A \) and his own secret keys.

**First step.** \( B \) uses his secret key \( N_{B,i} \) to clean the noise, calculating, for each \( a, g \in \{1, \ldots, d\} \),
\[
\prod_{b \in \{1, \ldots, d\}} (y_{B,b,g})^{(N_{B,b})_{a,g}} = \prod_{b \in \{1, \ldots, d\}} (c^{(N_{B,b})_{a,g}})_{a,g} = \prod_{b \in \{1, \ldots, d\}} (c^{N_{B,b}}_{a,g})_{a,g} = (c^{N_{B,b}})_{a,g} = (c^{Q_{a}(\alpha^{x}x_{3})})_{a,g} = \kappa_{a,g}.
\]

**Second step.** \( B \) inserts his main secret key, calculating, for each \( a, g \in \{1, \ldots, d\} \),
\[
\prod_{j \in I} \prod_{b \in \{1, \ldots, d\}} (y_{B,b,j})^{(A_{j}x_{3})_{a,g}} = \prod_{j \in I} \prod_{b \in \{1, \ldots, d\}} (c^{(A_{j}x_{3})_{a,g}})_{a,g} = \prod_{j \in I} \prod_{b \in \{1, \ldots, d\}} (c^{A_{j}x_{3}})_{a,g} = \kappa_{a,g}.
\]

\( A \) computes the SSK using the public key of \( B \) and her own secret key, and calculating, for each \( a, g \in \{1, \ldots, d\} \),
\[
\prod_{j \in I} \prod_{b \in \{1, \ldots, d\}} (y_{B,b,j})^{(A_{j}x_{3})_{a,g}} = \prod_{j \in I} \prod_{b \in \{1, \ldots, d\}} (c^{(A_{j}x_{3})_{a,g}})_{a,g} = \prod_{j \in I} \prod_{b \in \{1, \ldots, d\}} (c^{A_{j}x_{3}})_{a,g} = \kappa_{a,g}.
\]

Since (1) implies \( x_{B,j}^{(A_{j}x_{3})} = (\alpha^{x}x_{3})^{(A_{j}x_{3})} \) for each \( j \in I \), this becomes equal to
\[
\prod_{j \in I} (c^{A_{j}x_{3}})_{a,g} = (\kappa_{a,g})_{a,g}.
\]

### 1.2 Attacks

The eavesdropper \( E \) knows the following:

(i) the public parameters:
\[ d \in \mathbb{N}, \quad F, \quad \alpha \in M(d; F), \quad I \subseteq \mathbb{N}, \]

(ii) the public keys of \( B \):
\[ y_{B,i,j} \in M(d; F), \quad j \in I, \]

(iii) the structure of the public keys of \( B \):
\[ y_{B,i,j} := c^{(N_{B,i}A_{j})}_{a,b} = (c^{N_{B,i}A_{j}})_{a,b}, \quad j \in I, a, b \in \{1, \ldots, d\}, \]
\[ y_{B,i,j} := c^{(A_{j}x_{3})}_{a,b} = (c^{A_{j}x_{3}})_{a,b}, \]

(iv) the public key of \( A \):
\[ y_{A} \in M(d; F), \quad y_{A,a,g} = c^{(N_{B,i}A_{j}x_{3})}_{a,g} = (c^{N_{B,i}A_{j}x_{3}})_{a,g}, \]

(v) the constraints (1).
$E$ wants to know the SSK:
\[ \kappa_{ab} := c^{Q(a^x)b}x_{ab} = (c^{Q(a^x)b}x_{ab})_{ab}, \quad a, b \in \{1, \ldots, d\}. \]

Assuming zero cost logarithms, $E$ can compute for each $a, b \in \{1, \ldots, d\}$
\[ \log y_{B,2,j,ab} = (N_{B,3}A_j)_{ab} \log c, \quad j \in I, \]
\[ \log y_{B,3,j,ab} = (A_j x_{B,3})_{ab} \log c, \quad j \in I, \]
\[ \log y_A = (N_{B,3}Q_n(a^{x_A}))_{ab} \log c, \]
\[ \log \kappa = x_{B,3}Q(a^{x_A}) \log c. \]

In addition $E$ knows that the logarithm of the SSK is
\[ \log \kappa_{ab} = (x_{B,3}Q(a^{x_A}))_{ab} \log c, \quad a, b \in \{1, \ldots, d\}. \]

This gives the following matrix equations:
\begin{align*}
\log y_{B,2,j} &= N_{B,3}A_j \log c, \quad j \in I, \tag{5} \\
\log y_{B,3,j} &= A_j x_{B,3} \log c, \quad j \in I, \tag{6} \\
\log y_A &= N_{B,3}Q_n(a^{x_A}) \log c, \tag{7} \\
\log \kappa &= x_{B,3}Q(a^{x_A}) \log c. \tag{8}
\end{align*}

$E$ knows that $N_{B,3}$ is invertible and, even not knowing $N_{B,3}^{-1}$, $E$ knows from (7) that
\[ N_{B,3}^{-1} \log y_A = Q(a^{x_A}) \log c \]
must hold. From this and (8), even not knowing $\log \kappa$, $E$ knows that the following relation must hold:
\[ \log \kappa = x_{B,3}N_{B,3}^{-1}(\log y_A). \tag{9} \]

### 1.3 Solutions of the system (5), (6), (9)

In order to study the system (5), (6), (9), let us introduce the simplifying notations
\begin{align*}
Q(a^{x_A}) \log c &=: x, \tag{10} \\
N_{B,3} &=: y, \\
A_j \log c &=: z_j, \quad j \in I, \tag{11}
\end{align*}
for the unknowns to $E$, and
\begin{align*}
\log y_{B,n,j} &=: a_{n,j}, \quad j \in I, \quad n = 2, 3, \tag{12} \\
\log y_A &=: a \tag{13}
\end{align*}
for the known matrices. Then the system (5), (6), (9) becomes equivalent to the system
\begin{align*}
a_{n,j} &= yz_j, \quad j \in I, \tag{14} \\
a_{3,j} &= z_j x_{B,3}, \quad j \in I, \tag{15} \\
\log \kappa &= x_{B,3} y^{-1} a. \tag{16}
\end{align*}

Since no $z_j$ appears in equation (16), the only possibility to solve the system (14), (15), (16) is to deduce $y$ and $x_{B,3}$ from (14), (15) and to replace them in (16). Equations (14), (15) are a system of $2|I|$ quadratic equations in the $|I| + 2$ matrix unknowns (for $E$):
\[ z_j, y, x_{B,3}, \quad j \in I. \]

However, to find the SSK, $E$ does not need to know all the $z_j$. It is sufficient to know one of them, let us call it $z_j$. Clearly, up to a relabeling, $z_j$ can be replaced by any $z_j$ with $j \in I$. Notice that $z_j$ is invertible if and only if $A_j$ is invertible.
1.4 Attacks if $A_1$ is invertible

**First attack.** Since $y$ is invertible, (14) implies that $z_1$ (equivalently $A_1$) is invertible if and only if $a_{2,1}$ is invertible. Suppose that $z_1$ is invertible. Then, in the notations of Section 1.3, we have

$$y = a_{2,1} z_1^{-1},$$

$$x_{B,3} = z_1^{-1} a_{3,1}. \tag{17}$$

Hence, using (17), (18) and (16), $E$ can compute

$$\log \kappa = x_{B,3} y^{-1} a = z_1^{-1} a_{3,1} a_{2,1}^{-1} a.$$ \tag{19}

Therefore, if $E$ knows $z_1$, then she knows $\log \kappa$, hence the SSK $\kappa$.

**Second attack.** $E$ may deduce $z_1$ from (14) obtaining

$$z_1 = y^{-1} a_{2,1}.$$ \tag{20}

Then, using (18) and the invertibility of $a_{2,1}$, she finds

$$x_{B,3} = a_{2,1}^{-1} y a_{3,1}$$

and from this she deduces

$$\log \kappa = x_{B,3} y^{-1} a = a_{2,1}^{-1} y a_{3,1} y^{-1} a.$$ \tag{21}

**Third attack.** From (5) and (6), if $A_1$ is invertible, $E$ obtains

$$(\log y_{B,2;1})^{-1} (\log y_{A})(\log y_{B,3}) = (N_{B,3}^{-1} A_1 \log c)^{-1} (\log y_{A})(A_1 x_{B,3} \log c)$$

$$= (A_1 \log c)^{-1} N_{B,3}^{-1} (\log y_{A})(A_1 \log c) x_{B,3}$$

$$= A_1^{-1} N_{B,3}^{-1} (\log y_{A}) A_1 x_{B,3}. \tag{22}$$

Equivalently, in the notations (10), (11), (12), (13):

$$a_{2,1}^{-1} a a_{3,1} = (y z_1)^{-1} a z_1 x_{B,3} = z_1^{-1} y^{-1} a z_1 x_{B,3}. \tag{23}$$

**Fourth attack.** From (21) it follows that, if

$$[z_1, y^{-1} a] = 0,$$ \tag{24}

then

$$\log y_{B,2;1}^{-1} (\log y_{A})(\log y_{B,3;1}) = a_{2,1}^{-1} a a_{3,1} = y^{-1} a x_{B,3} = N_{B,3}^{-1} (\log y_{A}) x_{B,3}. \tag{25}$$

Comparing this with (9), we see that, if in addition to (22) one has (24), then

$$(\log y_{B,2;1})^{-1} (\log y_{A})(\log y_{B,3;1}) = x_{B,3} N_{B,3}^{-1} (\log y_{A})$$

hence $E$ can express the key in terms of the public parameters:

$$\log \kappa = x_{B,3} N_{B,3}^{-1} (\log y_A) = \log \kappa.$$ \tag{26}

In conclusion: if in addition to the invertibility of $A_1$, we also suppose that condition (3) is not satisfied, i.e. that

$$[A_1, N_{B,3}^{-1} \log y_A] = 0,$$ \tag{27}

and that (2) holds, then $E$ can reconstruct the SSK.
Fifth attack. From (5) and (6), if $A_1$ is invertible, $E$ obtains
\[
(\log y_{B,3,1})(\log y_A)(\log y_{B,2,1})^{-1} = (A_1x_{B,3}\log c)(\log y_A)(N_{B,3}A_1\log c)^{-1}
\]
\[
= (A_1x_{B,3}\log c)(\log y_A)(A_1\log c)^{-1}N_{B,3}^{-1}
\]
\[
= A_1x_{B,3}(\log y_A)A_1^{-1}N_{B,3}^{-1}
\]
or, equivalently,
\[
(\log y_{B,3,1})(\log y_A)(\log y_{B,2,1})^{-1} = z_1x_{B,3}a^{-1}N_{B,3}^{-1}.
\]
If
\[
[z_1, x_{B,3}a] = 0,
\]
then the above identity is equivalent to
\[
(\log y_{B,3,1})(\log y_A)(\log y_{B,2,1})^{-1} = x_{B,3}(\log y_A)N_{B,3}^{-1}.
\]
Comparing this with (9), we see that, if in addition to
\[
[A_1, x_{B,3}] = [A_1, \log y_A] = 0
\]
one has
\[
[N_{B,3}, y_A] = 0,
\]
then $E$ can express the SSK in terms of the public parameters:
\[
(\log y_{B,3,1})(\log y_A)(\log y_{B,2,1})^{-1} = x_{B,3}N_{B,3}^{-1}(\log y_A) = \log k.
\]

Sixth attack. Suppose that the $y_{B,2,j}$ are invertible, so that $E$ can form the product
\[
(\log y_{B,2,1})^{-1}(\log y_{B,3,j}) = (\log c)^{-1}A_j^{-1}N_{B,3}^{-1}A_jx_{B,3}(\log c).
\]
If $c$ is a scalar, this is equivalent to
\[
(\log y_{B,2,1})^{-1}(\log y_{B,3,j}) = A_j^{-1}N_{B,3}^{-1}A_jx_{B,3}.
\]
So $E$ can compute
\[
(\log y_{B,2,1})^{-1}(\log y_{B,3,j}) \log y_A = A_j^{-1}N_{B,3}^{-1}A_jx_{B,3} \log y_A.
\]
Thus, if
\[
[A_j, N_{B,3}] = 0,
\]
(23)
this expression can be simplified obtaining
\[
(\log y_{B,2,1})^{-1}(\log y_{B,3,j}) \log y_A = N_{B,3}^{-1}x_{B,3} \log y_A.
\]
Comparing this with (9), we see that, if in addition to (23) one has
\[
[x_{B,3}, N_{B,3}] = 0,
\]
then $E$ can express the SSK in terms of the public parameters:
\[
(\log y_{B,2,1})^{-1}(\log y_{B,3,j}) \log y_A = x_{B,3}N_{B,3}^{-1} \log y_A = \log k.
1.5 The role of commutativity

*First attack.* If $x_1$ commutes with $a_{3,1}$,

$$[a_{3,1}, x_1] = 0,$$

then in equation (19) the variable $x_1$ disappears and (19) becomes

$$\log \kappa = a_{2,1}a_{2,1}^{-1}a$$

or, equivalently, in view of (10), (11), (12), (13),

$$\log \kappa = \log y_{B,3,1}^{-1} \log y_A,$$

which expresses the SSK in terms of the public parameters.

In conclusion: if in addition to the invertibility of $A_1$ we also suppose that condition (4) is not satisfied, i.e. that

$$[\log y_{B,3,1}, A_1] = 0,$$

then $E$ can reconstruct the SSK.

*Second attack.* Similarly, if $y$ commutes with $a_{3,1}$,

$$[a_{3,1}, y] = 0,$$

then in equation (20) the variable $y$ disappears and (20) becomes

$$\log \kappa = a_{2,1}^{-1}a_{2,1}^{-1}a$$

or, equivalently, in view of (10), (11), (12), (13),

$$\log \kappa = (\log y_{B,2,1}^{-1}) \log y_{B,3,1} \log y_A,$$

which again expresses the SSK in terms of the public parameters.

In conclusion: if in addition to the invertibility of $A_1$ we also suppose that condition (4) is not satisfied, i.e. that

$$[\log y_{B,3,1}, y] = 0,$$

then $E$ can reconstruct the SSK.

**Remark.** Notice that the conditions (24) and (25) are always satisfied in the one-dimensional case.

1.6 Attacks under strong invertibility conditions

In the present section we suppose that the $A_j$ (equivalently all the $z_j$) are invertible for each $j \in I$. Moreover, we fix, as always possible, a numeration of $I$ which identifies it with a set of the form $I \equiv \{1, \ldots, N\}$. Then equations (14), (15) become, respectively,

$$a_{2,j}z_j^{-1} = y \quad \text{for all } j \in I,$$

$$a_{3,j}z_j^{-1} = x_{B,3} \quad \text{for all } j \in I.$$

Equation (26) implies in particular that

$$a_{2,j}z_j^{-1} = \cdots = a_{2,d}z_d^{-1}.$$

If $a_{2,2}$ is invertible, we deduce from

$$a_{2,2}z_2^{-1} = a_{2,2}x_2^{-1}. $$
that (27) is equivalent to
\[ z_2^{-1} = a_{22}^{-1} a_{21} z_1^{-1}. \]
If \( a_{23} \) is invertible, we deduce from
\[ a_{22} z_2^{-1} = a_{23} z_3^{-1} \]
that
\[ z_3^{-1} = a_{23}^{-1} a_{22}^{-1} a_{21} z_1^{-1} = a_{23}^{-1} z_1^{-1}. \]
Suppose that \( a_{j} \) is invertible for each \( j \in I \) and, by induction, that
\[ z_j^{-1} = a_{j,j}^{-1} a_{j,j-1} z_{j-1}^{-1}. \]
Then from
\[ a_{j,j} z_j^{-1} = a_{j,j+1} z_{j+1}^{-1} \]
we deduce that
\[ z_{j+1}^{-1} = a_{j,j+1}^{-1} a_{j,j} z_j^{-1} = a_{j,j+1}^{-1} a_{j,j} a_{j,j} a_{j,j} z_1^{-1} = a_{j,j+1}^{-1} z_1^{-1}. \]
Therefore, by induction (28) holds for each \( j \in I \).
From the same argument, with \( a_{j,j} \) replaced by \( a_{j,j} \), we deduce that
\[ z_j^{-1} = a_{j,j}^{-1} z_j^{-1}. \]
Combining (28) and (29), we obtain the system of \(|I| - 1\) homogeneous linear equations in the unknown matrix \( z_1^{-1} \):
\[ (a_{j,j}^{-1} a_{j,j+1} z_{j+1}^{-1} - a_{j,j}^{-1} a_{j,j+1} z_1^{-1}) z_j^{-1} = 0, \quad j \in I \setminus \{1\} \tag{30} \]

**Remark.** We know a priori that at least one matrix solution of the system (30) exists. However, if \(|I| < d^2 + 1\), then \((d^3 + 1) - |I|\) coefficients of this solution are indeterminate.

### 1.7 Computational complexity

**Computation of \( y_A \)**

- Computation of \( \alpha^{-x_A} \):
  - \( x_A \)-scalar: \( \log x_A \) matrix multiplications: each matrix multiplication, \( d^3 \) scalar multiplications.
  - \( x_A \)-matrix: \( d^3 \) scalar exponentiations.
- Computation of \( (\alpha^{-x_A})^j \) (no difference between \( x_A \)-scalar or \( x_A \)-matrix): \( \log j \) matrix multiplications for each \( j \in \{1, \ldots, |I|\} \).
  - Total number of matrix multiplications:
    \[ \sum_{j=1}^{|I|} \log j = \log(|I|!) \sim |I| \log |I|. \]

In conclusion, we have \((\log x_A + |I| \log |I|)d^3 \) multiplications.

- Number of exponentiations: \( d^4 \).
  - Multiplications for each entry of \( y_A \): \(|I| \cdot d \).

In total, we have \( d^2 ((\log x_A + |I| \log |I|)d^3 + |I|d) \) multiplications to produce \( y_A \), plus \( d^4 \) exponentiations.

The complexity of the number of exponentiations is of order \( \log e \), where \( e \) is the exponent. This gives \((\log e)d^4 \) multiplications.

**Computation of \( y_{B,2} \)**. One matrix multiplication, i.e. \( d^3 \) multiplications, and \( d^3 \) exponentiations, i.e. an order of \( d^3 \log e \) multiplications.

In total, we have \( d^3 + d^3 \log e \) multiplications.

**Computation of \( y_{B,3} \)**. The same order of complexity as \( y_{B,2} \).
**Computation of the SSK: B.** For a single entry of the SSK, we have \( d^4 \) exponentiations, i.e. order of \( d^4 \log e \) multiplications plus \( d \) multiplications, and additional \( d^4 \) exponentiations for a total of \( d^4 \log e \) multiplications plus \( d \) multiplications.

In total, we have \( 2d^4 \log e + d \) multiplications for a single entry of the SSK.
For the whole SSK, we have \( d^2(2d^4 \log e + d) \) multiplications.

**Computation of the SSK: A.** We have \( d^4 \) exponentiations and \( 2d \) multiplications for each entry in the SSK.
In total, we have \( d^2(d^4 \log e + 2d) \) multiplications.

## 2 Conclusions

Strongly asymmetric cryptography is a program, rather than a single algorithm, based on an idea briefly described in Section 1. The realization of this program depends on the construction of concrete algorithms based on this idea. In the present paper the structure of one of these algorithms is discussed in detail and some possible attacks, and constructive complexity estimates are described. The software implementation of this algorithm, as well as the construction of new algorithms based on the same idea are now under investigation. The authors hope to come back to this point soon.

**Funding:** Luigi Accardi acknowledges support by the RSF grant 14-11-00687.

## References


Received April 13, 2015; revised July 20, 2015; accepted August 17, 2015.