Which new RSA Signatures can be Computed from RSA Signatures, Obtained in a Specific Interactive Protocol?

Jan-Hendrik Evertse [1]
Department of Mathematics and Computer Science, University of Leiden,
P.O. Box 9512, 2300 RA Leiden, The Netherlands.

Eugène van Heyst
CWI Centre for Mathematics and Computer Science,
Kruislaan 413, 1098 SJ Amsterdam, The Netherlands.

Abstract. We consider certain interactive protocols, based on RSA. In these protocols, a signature authority $\mathcal{Z}$ (which chooses the RSA-modulus N that is kept fixed) issues a fixed number of RSA-signatures to an individual $\mathcal{A}$. These RSA-signatures consist of products of rational powers of residue classes modulo N; some of these residue classes are chosen by $\mathcal{Z}$ and the others can be chosen freely by $\mathcal{A}$. Thus, $\mathcal{A}$ can influence the form of the signatures that he gets from $\mathcal{Z}$. $\mathcal{A}$ wants to choose his residue classes in such a way that he can use the signatures he gets from $\mathcal{Z}$ to compute a signature of a type not issued by $\mathcal{Z}$.
In previous literature, some special cases of our protocols were considered, namely that only $\mathcal{A}$ chooses the residue classes ([Dav82],[Denn84],[DO85]) and that only $\mathcal{Z}$ chooses the residue classes [EvH92]. The results in our paper are used under the following assumptions:

- $\mathcal{A}$ cannot compute RSA-roots on randomly chosen residue classes modulo N.
- In his computations, $\mathcal{A}$ uses only multiplications and divisions modulo N.

Our main result gives a necessary and sufficient condition under which $\mathcal{A}$ is able to influence the signatures he gets from $\mathcal{Z}$ in such a way that he can use these RSA-signatures to compute a signature of a type not issued by $\mathcal{Z}$. It turns out that this condition is equivalent to the solvability of a particular quadratic equation in integral matrices. We also study a particular case of this problem in more detail.

1. Introduction

A challenging problem in cryptology is to study the security of certain classes of interactive protocols. To this end, one must investigate certain classes of attacks on these protocols.

For instance, consider the interactive *ping-pong protocols* (cf. [EGS85]). In such a

[1] This research has been made possible by a fellowship of the Royal Netherlands Academy of Arts and Sciences (K.N.A.W.)

R.A. Rueppel (Ed.): Advances in Cryptology - EUROCRYPT '92, LNCS 658, pp. 378-389, 1993.

protocol (which consists of several moves), one party generates a secret message, applies a sequence of operators to it, and sends it to the other party. This party also applies a sequence of operators to the message received, and sends back the result. In each move of the protocol, one of the parties applies a sequence of operators to the last message it received, and sends it back. The question is, whether an "active" third party can discover the initial message (by altering messages, impersonating other users, etc.).

In this chapter we consider interactive protocols based on the RSA-system ([RSA78]) as in Figure 1 below, in which only one party $\mathcal{Z}$, called the *signature authority*, can create signatures and issues these to the other parties, called the *individuals*. Such protocols are used, for instance, in credential and payment systems, in which a signature represents a credential or money. In fact, in such credential systems or payment systems, the signature authority issues different types of signatures, corresponding to different credentials or different values of money. The security of these systems depends on whether an individual (or a group of conspiring individuals) is not able to compute a useful signature of a type not issued by the signature authority, by using the signatures which were issued before by the authority $\mathcal{Z}$ (for instance by using the multiplicative property of RSA).

Initially, $\mathcal{Z}$ chooses two large primes P, Q and computes their product N. Further, $\mathcal{Z}$ chooses two integers a,b coprime to $\varphi(N) = (P-1)(Q-1)$. $\mathcal{Z}$ makes N, a, b public, and keeps P and Q secret. Let c, d also be some integers coprime to $\varphi(N)$. In this protocol, $\mathcal{Z}$ chooses a residue class u, and $\mathcal{A}$ wants to choose h in such a way that after the execution of this protocol, he is able to compute from $\{u,h,u^{1/a}h^{1/b}\}$ a pair $\{t,k\}$ satisfying $t \equiv u^{1/c}k^{1/d} \pmod{N}$. The reason for considering such problems is that in the payment systems and credential systems mentioned above, the user gets so-called blind signatures from $\mathcal{Z}$ which contain residue classes chosen by $\mathcal{Z}$ and residue classes chosen by the user himself.

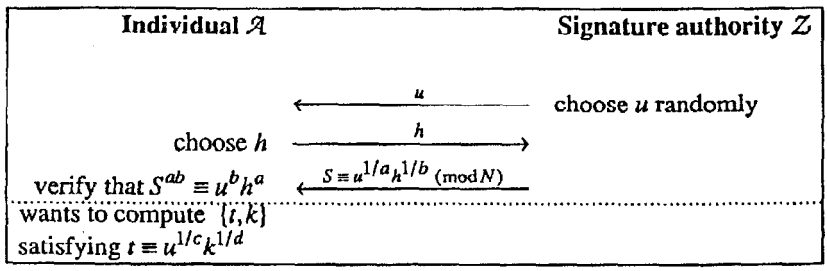

Fig. 1. An interactive signature-issuing protocol in which the signature authority $\mathcal{Z}$ issues a signature to individual $\mathcal{A}$.

In [EvH92] we studied the case in which $\mathcal{A}$ has no influence on the signature

received, that is, $\mathcal{A}$ chooses no residue class (i.e., b=1 in Figure 1). A necessary and sufficient condition was given for the computation of this new signature to be feasible for $\mathcal{A}$.

In [Dav82], [Denn84] and [DO85] the case is studied in which $\mathcal{Z}$ chooses no residue class, that is, in which individuals were able to obtain signatures on messages of their choice (i.e., a=1 in Figure 1). [Dav82] states that $\mathcal{A}$ can decrypt ciphertext encrypted under $\mathcal{Z}$'s public key and can forge $\mathcal{Z}$'s signature on meaningful messages; [Denn84] can foil this attack by using hashing. [DO85] showed that if $\mathcal{A}$ can get $L(N)^{1/2}$ RSA-signatures modulo N on carefully chosen residue classes, where $L(N) = \exp\{(1+o(1))(\log N \log\log N)^{1/2}\}$ as $N \to \infty$, then $\mathcal{A}$ can compute any RSA-signature modulo N of his choice in $L(N)^{1/2}$ bitoperations.

We consider more general interactive protocols in which $\mathcal{Z}$ issues a fixed number of RSA-signatures to $\mathcal{A}$. These RSA-signatures consist of products of rational powers of residue classes modulo the composite number N of the underlying RSA-scheme; some of these residue classes are chosen by $\mathcal{Z}$ and the others can be chosen freely by $\mathcal{A}$. We make the following two assumptions:

(i) $\mathcal{A}$ cannot compute RSA-roots of randomly chosen residue classes modulo N.

(ii) In his computations, the only operations that $\mathcal{A}$ uses are multiplications and divisions modulo N.

The problem whether assumption (ii) is necessary remains open. We formulate a necessary and sufficient condition under which $\mathcal{A}$ is able to influence the signatures he receives from $\mathcal{Z}$ in such a way that he can later use these signatures to compute a signature of a type not issued by $\mathcal{Z}$. It turns out that this condition is equivalent to the solvability of a particular quadratic equation in integral matrices.

This paper is organized as follows. The notation used is introduced in the next section, while in Section 3 the interactive protocol considered and the problem we are facing are defined. In Section 4 we analyze this problem by assuming that the individual performs only multiplications and divisions modulo N (this is called an algebraic strategy). A special case of the considered protocol with these algebraic strategies is studied in Section 5, while some generalizations to the protocol of Section 3 are given in Section 6.

2. Notation

The following notation is used throughout this paper.

$\gcd(a_1,\ldots,a_l)$	the greatest common divisor of $a_1,\ldots,a_l$; defined for rational numbers by $\gcd(a_1,\ldots,a_l):=\frac{\gcd(a_1 d,\ldots,a_l d)}{d}$, where $d\in\mathbb{N}$ is such that $a_1 d,\ldots,a_l d\in\mathbb{Z}$; this definition is independent of the choice of d.
S^k	the set of vectors $(a_1,\ldots,a_k)$ with $a_1,\ldots,a_k\in S$, for any set S; we use bold face characters to denote vectors.
$a\in_R S$	denotes the random selection of an element (that will be called a) from S according to the uniform probability distribution; for any set S.
$\mathbf{ab}$	$(a_1 b_1,\ldots,a_k b_k)$, if $\mathbf{a}=(a_1,\ldots,a_k)$ and $\mathbf{b}=(b_1,\ldots,b_k)$.
$\mathbf{a}\equiv\mathbf{b}\pmod m$	$m^{-1}(\mathbf{b}-\mathbf{a})\in\mathbb{Z}^k$; this is defined for $\mathbf{a},\mathbf{b}\in\mathbb{Q}^k$, $m,k\in\mathbb{N}$, $m>0$.
N	a composite, odd number.
$\mathbb{Z}_N^*$	the set $\{a \mid a\in\mathbb{N}, 1\le a\le N, \gcd(a,N)=1\}$ of $\varphi(N)$ elements.
$\tilde{\mathbb{Q}}_N$	the ring $\{\frac{a}{d} \mid a,d\in\mathbb{Z}, d>0, \gcd(d,\varphi(N))=1\}$.
$x^{1/d}\pmod N$	the d^{th} *RSA-root* of $x \pmod N$: the unique solution $S\in\mathbb{Z}_N^*$ to $S^d\equiv x\pmod N$ for $x\in\mathbb{Z}_N^*$ and $d\in\mathbb{Z}$ with $\gcd(d,\varphi(N))=1$.
$\mathbf{x}^{\mathbf{a}}\pmod N$	the number $S\in\mathbb{Z}_N^*$ with $S\equiv x_1^{a_1}x_2^{a_2}\ldots x_k^{a_k}\pmod N$, for $\mathbf{x}=(x_1,\ldots,x_k)\in(\mathbb{Z}_N^*)^k$ and $\mathbf{a}=(a_1,\ldots,a_k)\in(\tilde{\mathbb{Q}}_N)^k$.
$[\mathbf{a}_1\ldots\mathbf{a}_l]$	the matrix with columns $\mathbf{a}_1,\ldots,\mathbf{a}_l$.
$\mathbf{x}^A\pmod N$	$(\mathbf{x}^{\mathbf{a}_1},\ldots,\mathbf{x}^{\mathbf{a}_l})\in(\mathbb{Z}_N^*)^l$, for $A=[\mathbf{a}_1\ldots\mathbf{a}_l]\in(\tilde{\mathbb{Q}}_N)^{k,l}$ and $\mathbf{x}\in(\mathbb{Z}_N^*)^k$; so $(\mathbf{x}^A)^B=\mathbf{x}^{AB}$.
$l(n)$	length of the binary representation of $n\in\mathbb{N}$; the length of a negative integer m, a rational number p/q $(q\neq1)$, a vector $\mathbf{c}$, and a matrix $A=(a_{i,j})$ are defined by: $l(m)=l(-m)+1$, $l(p/q)=l(p)+l(q)+1$, $l(\mathbf{c})=\sum_i(l(c_i)+1)$, and $l(A)=\sum_{i,j}(l(a_{i,j})+1)$, respectively.
length(A,B)	$l(A)+l(B)$.

3. The Protocol and Problem under Consideration

In this paper we consider the following interactive Protocol 1 (see Figure 2), which is more general than the protocol of Figure 1. The signature authority $\mathcal{Z}$ has created an RSA-modulus N, and issues RSA-signatures that will be products of rational powers of residue classes modulo N. Let $\mathbf{M}=\{A,B,C,D\}$ be a set of fixed rational matrices $A\in(\tilde{\mathbb{Q}}_N)^{k,l}$, $B\in(\tilde{\mathbb{Q}}_N)^{m,l}$, $C\in(\tilde{\mathbb{Q}}_N)^{k,n}$, $D\in(\tilde{\mathbb{Q}}_N)^{m,n}$. In Protocol 1, an individual $\mathcal{A}$ requests $\mathcal{Z}$ to create the RSA-signature (in fact it consists of l RSA-signatures)

$$\mathbf{s}_1\equiv\mathbf{u}^A\mathbf{h}_1^B\pmod N,$$

where $\mathbf{u}\in(\mathbb{Z}_N^*)^k$ is chosen by $\mathcal{Z}$, and $\mathbf{h}_1\in(\mathbb{Z}_N^*)^m$ is chosen by $\mathcal{A}$ ($\mathbf{h}_1$ may depend on

N,M = $\{A,B,C,D\}$ and $\mathbf{u}$). But actually, $\mathcal{A}$ wants to have the RSA-signature

$$s_2 \equiv \mathbf{u}^C \mathbf{h}_2^D \pmod{N},$$

for some $\mathbf{h}_2 \in (\mathbb{Z}_N^*)^m$ and $s_2 \in (\mathbb{Z}_N^*)^n$. Therefore he wants to choose $\mathbf{h}_1$ in such a way that after the execution of Protocol 1, he can compute from $\{N,A,B,C,D,\mathbf{u},\mathbf{h}_1, s_1 \equiv \mathbf{u}^A \mathbf{h}_1^B\}$ a pair $\{s_2,\mathbf{h}_2\}$ satisfying $s_2 \equiv \mathbf{u}^C \mathbf{h}_2^D$. This way of choosing $\mathbf{h}_1$ (which may depend on N,M = $\{A,B,C,D\}$ and $\mathbf{u}$) in order to be able to compute a pair $\{s_2,\mathbf{h}_2\}$, is called an M-*strategy*. We assume that $\mathcal{A}$ uses a probabilistic Turing machine that can do random coin tosses. We neglect the computation time used by $\mathcal{Z}$, so the running time of an M-strategy is the number of steps that $\mathcal{A}$'s machine needs to compute $\mathbf{h}_1, s_2$ and $\mathbf{h}_2$. We assume that the running time of an M-strategy depends only on N and M, i.e., is independent of the choice of $\mathbf{u}$ of $\mathcal{Z}$ and the random coinflips by $\mathcal{A}$'s machine. However, the output of an M-strategy is a stochastic variable on the probability space defined by the uniform choice of $\mathbf{u}$ from $(\mathbb{Z}_N^*)^k$ by $\mathcal{Z}$, and the random coin tosses of $\mathcal{A}$'s machine. In general, the probability that the M-strategy outputs $\{s_2,\mathbf{h}_2\}$ with $s_2 \equiv \mathbf{u}^C \mathbf{h}_2^D$ is smaller than 1.

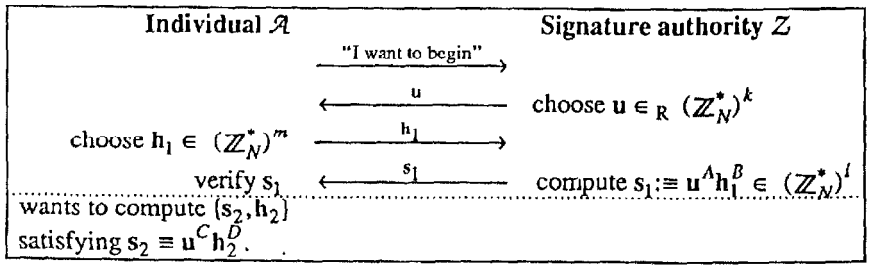

Fig. 2. The considered interactive Protocol 2

Problem 1. *For which systems of matrices* M = $\{A,B,C,D\}$ *are there feasible M-strategies, these are M-strategies with polynomial in length*(N,A,B,C,D) *running time that output with probability* $\geq \frac{1}{2}$, *say, a pair* $\{s_2,\mathbf{h}_2\}$ *satisfying* $s_2 \equiv \mathbf{u}^C \mathbf{h}_2^D$?

This problem was solved in [EvH90] for the special case that B and D are matrices consisting of only ones, i.e., for the non interactive case.

If there are no restrictions on $\mathbf{h}_2$ (e.g., $\mathbf{h}_2$ must be an element from a special subset of $(\mathbb{Z}_N^*)^m$), then we can restrict ourselves in Protocol 1 (and thus also in Problem 1) to the case that D consists of only zeros, according to the next lemma.

Lemma 1. *Let* $C \in (\tilde{\mathbb{Q}}_N)^{k,n}$, $D \in (\tilde{\mathbb{Q}}_N)^{m,n}$ and $\mathbf{u} \in (\mathbb{Z}_N^*)^m$. *Then there exists a matrix* $\tilde{C} \in (\tilde{\mathbb{Q}}_N)^{k,n}$ *(which is computable in polynomial time from* C *and* D*), such*

that computing a pair $(\mathbf{s},\mathbf{h}) \in (\mathbb{Z}_N^*)^n \times (\mathbb{Z}_N^*)^m$ *satisfying* $\mathbf{s} \equiv \mathbf{u}^C\mathbf{h}^D$ *is polynomial-time equivalent to computing* $\mathbf{u}^{\tilde{C}}$.

Proof. We can reformulate the identity $\mathbf{s} \equiv \mathbf{u}^C\mathbf{h}^D$ as

$$\mathbf{u}^C \equiv (\mathbf{s},\mathbf{h})^{\begin{bmatrix} I \\ -D \end{bmatrix}},$$

where the first n coordinates of vector $(\mathbf{s},\mathbf{h})$ are those of $\mathbf{s}$, the last m coordinates are those of $\mathbf{h}$, the first n rows of $\begin{bmatrix} I \\ -D \end{bmatrix}$ are those of the identity matrix I, and the last m rows are those of $-D$. According to [KaBa79], we can find in polynomial time unimodular matrices P,Q and a matrix $\begin{bmatrix} G \\ 0 \end{bmatrix}$ in Smith normal form, such that $P\begin{bmatrix} G \\ 0 \end{bmatrix} = \begin{bmatrix} I \\ -D \end{bmatrix} Q$. Because $\begin{bmatrix} I \\ -D \end{bmatrix}$ has full column rank, G is invertible. Define $\tilde{C} = CQG^{-1}$. Then from $(\mathbf{s},\mathbf{h})$ satisfying $\mathbf{s} \equiv \mathbf{u}^C\mathbf{h}^D$ we can compute $\mathbf{u}^{\tilde{C}}$ in polynomial time because

$$\begin{aligned} \mathbf{u}^{\tilde{C}} = \mathbf{u}^{CQG^{-1}} &= (\mathbf{s},\mathbf{h})^{\begin{bmatrix} I \\ -D \end{bmatrix} QG^{-1}} \equiv (\mathbf{s},\mathbf{h})^{P\begin{bmatrix} G \\ 0 \end{bmatrix} G^{-1}} = (\mathbf{s},\mathbf{h})^{P\begin{bmatrix} I \\ 0 \end{bmatrix}} \\ &= \text{first } n \text{ coordinates of } (\mathbf{s},\mathbf{h})^P. \end{aligned}$$

If, on the other hand, $\mathbf{u}^{\tilde{C}}$ is given, then we obtain the pair $(\mathbf{s},\mathbf{h})$ satisfying $\mathbf{s} \equiv \mathbf{u}^C\mathbf{h}^D$ by first computing the vector $\tilde{\mathbf{s}} = (\mathbf{u}^{\tilde{C}},1,\ldots,1)$ of length $n+m$ and then by defining $\mathbf{s},\mathbf{h}$ by $(\mathbf{s},\mathbf{h}) := \tilde{\mathbf{s}}^{P^{-1}}$. ❑

4. Algebraic Strategies

As shown in the previous section, we may restrict ourselves to the case in which M = $\{A,B,C,[0]\}$. It seems a hard problem to determine the matrices A,B,C for which there exist *arbitrary* feasible M-strategies. Therefore, we consider only M-strategies of a special kind, so-called *algebraic* M-*strategies*. In an algebraic strategy, $\mathcal{A}$ applied to $\mathbf{u}$ only multiplications and divisions mod N in order to compute $\mathbf{h}_1$, and the choice of these multiplications and divisions is independent of $\mathbf{u}$. It is conceivable that algebraic strategies are the best, i.e., if there is no feasible algebraic M-strategy, then there is also no feasible M-strategy of another kind. But we have no insight in this matter

Let $A \in (\tilde{\mathbb{Q}}_N)^{k,l}, B \in (\tilde{\mathbb{Q}}_N)^{m,l}, C \in (\tilde{\mathbb{Q}}_N)^{k,n}$ be fixed rational matrices. $\mathcal{A}$ is assumed to follow an algebraic M-strategy, hence, in Protocol 1, $\mathbf{h}_1$ must consist of products of integral powers of the entries of $\mathbf{u}$, i.e. $\mathbf{h}_1 \equiv \mathbf{u}^X$, for some $X \in \mathbb{Z}^{k,m}$. So instead of analyzing the general Protocol 1, we will analyze Protocol 2 in this section (see Figure 3, in which we write $\mathbf{h}$ in stead of $\mathbf{h}_1$).

We now assume also that it is computationally infeasible for $\mathcal{A}$ to compute RSA-roots modulo N, since otherwise he could forge all signatures. Under this assumption, the Corollary of [EvH92] implies the following for Protocol 2.

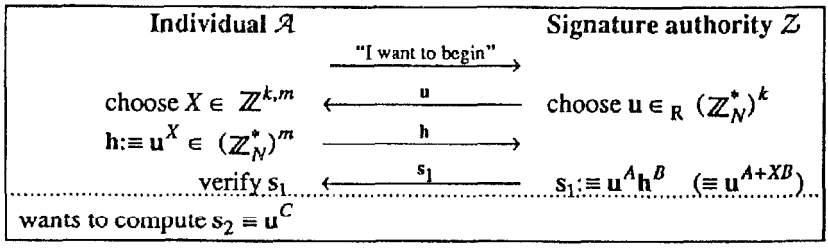

Fig. 3. Protocol 2, which is equivalent with Protocol 1, if $\mathcal{A}$ follows an algebraic M-strategy.

Proposition. *Let A,B,C be matrices with rational entries. Then the following two statements are equivalent*

(i) *There is a probabilistic polynomial in length(A,B,C) algorithm to compute integral matrices X,Y,Z such that* $C = (A+XB)Y + Z$.

(ii) *There is a feasible algebraic* M*-strategy for the individual* $\mathcal{A}$ *to compute* $\mathbf{h}$ *from* $\{N,A,B,C,\mathbf{u}\}$ *and* $\mathbf{u}^C$ *from* $\{N,A,B,C,\mathbf{u},\mathbf{u}^A\mathbf{h}^B\}$.

Because of this result, we are interested in the following problem.

Problem 2. *Let* $A \in (\tilde{\mathbb{Q}}_N)^{k,l}$, $B \in (\tilde{\mathbb{Q}}_N)^{m,l}$, $C \in (\tilde{\mathbb{Q}}_N)^{k,n}$ *be rational matrices. Find a polynomial (in length(A,B,C))-time algorithm that decides whether the equation*

$$C = (A+XB)Y + Z$$

is solvable in integral matrices $X \in \mathbb{Z}^{k,m}$, $Y \in \mathbb{Z}^{l,n}$, $Z \in \mathbb{Z}^{k,n}$, *and if so, find a solution X,Y,Z.*

We have not been able to solve Problem 2 in full generality. We have proven that there exists such a polynomial-time algorithm for Problem 2 in the case that n=1, but this is not included in this paper. In the next section, we solve Problem 2 in the special case that k=l=m=n=1.

5. A Special Case of Protocol 2

Let $a,b,c \in \tilde{\mathbb{Q}}_N$ be fixed and assume that the denominators of a,b, and c are coprime to $\varphi(N)$. We analyze the following Protocol 3 between $\mathcal{Z}$ and $\mathcal{A}$ (see Figure 4). In this

protocol, $\mathcal{A}$ receives from $\mathcal{Z}$ the RSA-signature $u^{a+xb} \pmod N$, which $\mathcal{A}$ can verify. Note that $\mathcal{A}$ cannot compute this signature on a randomly chosen residue class u himself, because in general $a+xb \in \mathbb{Q} \setminus \mathbb{Z}$.

The next lemma states when it is feasible for $\mathcal{A}$ to compute u^c after the execution of this protocol.

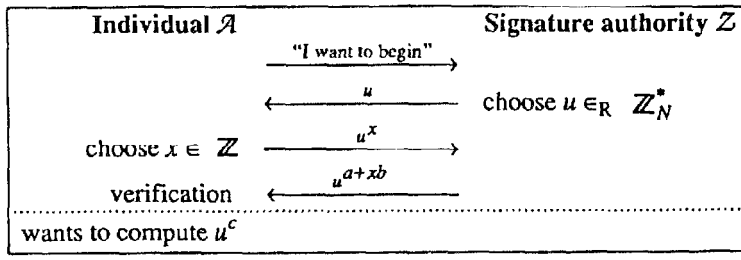

Fig. 4. Protocol 3.

Lemma 2. *$\mathcal{A}$ can choose x in Protocol 3 in such a way (and in polynomial time) that it is feasible for him to compute u^c after the execution of the protocol if and only if* $\gcd(1,a,b) \mid c$.

Note that $\gcd(1,a,b)$ is in general not 1, because a and b are rational numbers. This lemma can be proved by using Corollary 1 and the following two lemmas.

Lemma 3. *Let $a,b,c \in \mathbb{Q}$, $c \neq 0$. Then there exists an integer λ such that* $\gcd(a+\lambda b,c) = \gcd(a,b,c)$, *and this λ can be computed in polynomial (in length(a,b,c)) time.*

Proof. Define $\bar{a} = a / \gcd(a,b,c), \bar{b} = b / \gcd(a,b,c), \bar{c} = c / \gcd(a,b,c)$. Thus $\bar{a},\bar{b},\bar{c}$ are integers with $\gcd(\bar{a},\bar{b},\bar{c}) = 1$. It suffices to show that we can compute in polynomial time a $\lambda \in \mathbb{N}$ that satisfies $\gcd(\bar{a}+\lambda\bar{b},\bar{c}) = 1$.

For each prime number p and each $a \in \mathbb{Z}, a \neq 0$, let $\mathrm{ord}_p(a)$ be the integer such that $a \cdot p^{-\mathrm{ord}_p(a)}$ is an integer not divisible by p. Take

$$\lambda = \prod_{p \mid \bar{c}, p \nmid \bar{a}} p^{\mathrm{ord}_p(\bar{c})}.$$

Let p be a prime dividing $\bar{c}$. If $p \mid \bar{a}$, then $p \nmid \bar{b}$ (by $\gcd(\bar{a},\bar{b},\bar{c}) = 1$) and $p \nmid \lambda$ (by definition of λ), hence $p \nmid (\bar{a}+\lambda\bar{b})$. If $p \nmid \bar{a}$, then $p \mid \lambda$ (by definition of λ and $\mathrm{ord}_p(\bar{c}) \geq 1$) and hence also $p \nmid (\bar{a}+\lambda\bar{b})$. We conclude that no prime divides both $\bar{c}$ and $(\bar{a}+\lambda\bar{b})$; therefore $\gcd(\bar{a}+\lambda\bar{b},\bar{c}) = 1$.

Define the sequence $c_0 := |\bar{c}|$ and $c_{i+1} := c_i / \gcd(\bar{a},c_i)$ for $i = 0,1,2,\ldots$. Let i_0 be the

smallest integer such that $\gcd(\bar{a}, c_{i_0}) = 1$. It is easy to see that $c_{i_0} = \lambda$ and that $i_0 \leq l(\bar{c})$; thus λ can be computed in polynomial time. ❑

Lemma 4. *Let $a,b,c \in \mathbb{Q}$. Then there are $x,y,z \in \mathbb{Z}$ such that $c = (a+xb)y + z$ if and only if $\gcd(1,a,b) | c$. Further, if such $x,y,z \in \mathbb{Z}$ exist, then they can be computed in polynomial (in length(a,b,c)) time.*

Proof. Note that $a,b,1$ are integral multiples of $\gcd(1,a,b)$. Hence, if there exist $x,y,z \in \mathbb{Z}$ such that $c = (a+xb)y + z$, then c is also an integral multiple of $\gcd(1,a,b)$. Hence $\gcd(1,a,b) | c$.

On the other hand, assume that $\gcd(1,a,b) | c$. By Lemma 3 we can compute in polynomial time an $x \in \mathbb{Z}$ such that $\gcd(a+xb,1) = \gcd(1,a,b)$. Further we can compute in polynomial time $y,z \in \mathbb{Z}$ with $c = (a+xb)y + z$ (e.g., let $d \in \mathbb{N}$ such that $da,db,dc \in \mathbb{N}$, and use $\gcd(a+xb,1) | c$ and Euclid's algorithm to compute $y, z \in \mathbb{Z}$ with $dc = (da+xdb)y + dz$). This proves Lemma 4. ❑

Proof of Lemma 2.

(i) Suppose that $\gcd(1,a,b) | c$. According to Lemma 4, $\mathcal{A}$ can compute in polynomial time numbers $x,y,z \in \mathbb{Z}$ such that $c = (a+xb)y + z$. $\mathcal{A}$ will use the obtained number x during the execution of Protocol 2. Afterwards, $\mathcal{A}$ can compute u^c from $\{N,a,b,c,u,u^{a+xb}\}$ as follows:

$$(u^{a+xb})^y \cdot u^z \equiv u^c \pmod{N}.$$

(ii) Suppose that $\mathcal{A}$ can choose x in Protocol 2 in such a way that it is feasible for him to compute u^c after the execution of the protocol. Corollary 1 states that computing u^c from $\{N,a,b,c,u,u^{a+xb}\}$ for uniformly chosen $u \in \mathbb{Z}_N^*$ is feasible for $\mathcal{A}$ if and only if $c \in \mathbb{Z}\{1, a+xb\}$. That is, if and only if there are $y,z \in \mathbb{Z}$ such that $c = (a+xb)y + z$. ❑

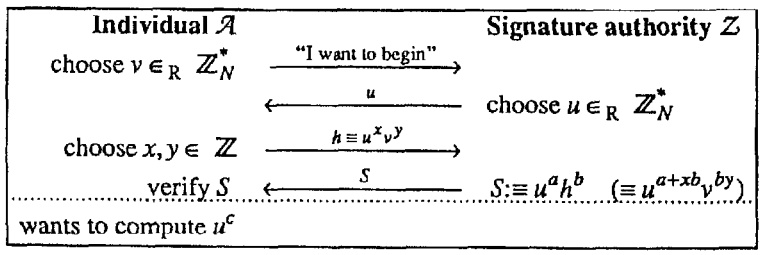

Fig. 5. Protocol 4.

We generalize Protocol 3 to Protocol 4 (see Figure 5), in which $\mathcal{A}$ initially chooses some residue class, but we will prove that doing so does not influence the feasibility of computing the signature u^c after the execution of the protocol.

We are interested in how $\mathcal{A}$ should choose v,x,y so that it is feasible for him to compute u^c from $\{u,v,u^{a+xb}v^{by},a,b,c,x,y\}$. According to the Corollary of [EvH92] this computation is feasible if and only if there is an integral solution z_1,z_2,z_3 to

$$\begin{cases} z_1 + z_3(a+xb) &= c, \\ z_2 + z_3 by &= 0. \end{cases}$$

According to Lemma 2, a necessary condition for the first equation is that $\gcd(1,a,b)|c$. But the number z_3 obtained there does not need to be a solution of the second equation. If $y=0$, then the z_3 obtained is also a solution of the second equation. Hence a *necessary* condition for the simultaneous solvability of the two equations is that $\gcd(1,a,b)|c$; and if $y=0$, then this condition is also *sufficient*. Therefore, the best strategy for $\mathcal{A}$ is to choose $y=0$ and to take x according to Lemma 3, and thus this algebraic strategy "works" if and only if $\gcd(1,a,b)|c$.

6. Generalizations

In Protocol 1 (see Figure 2) the system of matrices used is $M = \{A,B,C,D\}$, so the individual will receive one type of signature (s_1), and wants to compute a second type (s_2).

We now assume that there are $(t+1)$ types of signatures, so $\mathcal{Z}$ creates a public system of matrices $M = \{A_1,B_1,\ldots,A_{t+1},B_{t+1}\}$, where the matrices (A_i,B_i) are used for the i^{th} type. We assume that $\mathcal{Z}$ issues only signatures of type $1,\ldots,t$ to $\mathcal{A}$, and that $\mathcal{A}$ tries to compute a signature of type $t+1$ from these. Therefore, we want to consider the serial Protocol 5 (see Figure 6), in which we assume that $\mathcal{Z}$ uses the same $\mathbf{u}$ in every signature, that the individual chooses $\mathbf{h}_1,\ldots,\mathbf{h}_t$ (where $\mathbf{h}_i$ may depend on M,$\mathbf{u}$ and $\mathbf{s}_1,\ldots,\mathbf{s}_{i-1}$), and receives the signatures

$$\mathbf{s}_i \equiv \mathbf{u}^{A_i}\mathbf{h}_i^{B_i} \quad (i=1,\ldots,t).$$

$\mathcal{A}$ will not receive signatures of type $(t+1)$, so he tries to choose $\{\mathbf{h}_1,\ldots,\mathbf{h}_t\}$ in such a way that after receiving $\{\mathbf{s}_1,\ldots,\mathbf{s}_t\}$, he is able to compute a pair $(\mathbf{s}_{t+1},\mathbf{h}_{t+1})$ such that

$$\mathbf{s}_{t+1} \equiv \mathbf{u}^{A_{t+1}}\mathbf{h}_{t+1}^{B_{t+1}}.$$

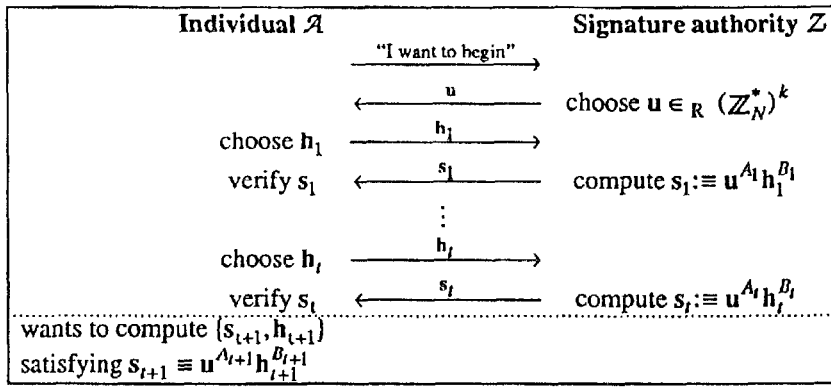

Fig. 6. The serial signature issuing Protocol 5, in which $\mathcal{A}$ receives the signatures $s_1,\ldots,s_t$.

If we assume that $\mathcal{A}$ will use an algebraic M-strategy, then we can prove it suffices to consider algebraic strategies on protocols with $t=1$, i.e., we can reduce Protocol 5 in polynomial time to Protocol 1 as follows:

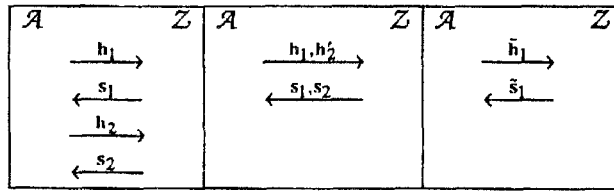

Fig. 7. How to modify Protocol 5 into Protocol 1.

Moves 3 up to 6 of Protocol 5 are shown in the left-hand side of Figure 7. Let d be the smallest positive integer such that dA_1 and dB_1 are integral matrices (so d can be the lcm of all the denominators of A_1 and B_1). Hence $\mathbf{s}_1^d$ can be computed by $\mathcal{A}$ *without* knowing $\mathbf{s}_1$, because $\mathbf{s}_1^d \equiv \mathbf{u}^{dA_1}\mathbf{h}_1^{dB_1}$, and the used exponents are integral. In order to create $\mathbf{h}_2$, $\mathcal{A}$ might use $\mathbf{s}_1$. But $\mathcal{A}$ only applies multiplications and divisions on $\mathbf{s}_1$, so $\mathcal{A}$ is able to compute $\mathbf{h}_2' := \mathbf{h}_2^d$ *without* knowing $\mathbf{s}_1$ ($\mathcal{A}$ will only use $\mathbf{s}_1^d$, which he could compute without knowing $\mathbf{s}_1$). By defining the new matrix $B_2' := \frac{1}{d}B_2$, we have that $\mathbf{h}_2'^{B_2'} \equiv \mathbf{h}_2^{B_2}$, so $\mathcal{A}$ does not need to know $\mathbf{s}_1$ in order to compute $\mathbf{h}_2^{B_2}$ (by using matrix B_2'). The possibility that $\mathcal{A}$ can compute $\mathbf{s}_{t+1}$ at the end of the protocol remains the same if we carry out the first four moves in parallel instead of serially (see Figure 7, middle). By defining $\tilde{\mathbf{s}}_1 := (\mathbf{s}_1, \mathbf{s}_2)$, $\tilde{\mathbf{h}}_1 := (\mathbf{h}_1, \mathbf{h}_2')$, $\tilde{A}_1 := [A_1\ A_2]$, $\tilde{B}_1 := [B_1\ B_2']^T$, we have that $\tilde{\mathbf{s}}_1 \equiv \mathbf{u}^{\tilde{A}_1}\mathbf{h}_1^{\tilde{B}_1}$; thus we can combine the first four moves into two (see Figure 7, right-hand side). In this way we obtain a protocol with 2 moves less. By repeating this argument, we only have to analyze a protocol with 2+2 moves, i.e., Protocol 1.

References

[Dav82] George Davida, Chosen signature cryptanalysis of the RSA (MIT) public key cryptosystem, Tech. rept. TR-CS-82-2, Dept of Electrical Engineering and Computer Science, Univ. of Wisconsin, October 1982.

[Denn84] Dorothy Denning, "Digital signatures with RSA and other public-key cryptosystems", *Comm. of the ACM*, **27** (1984) pp. 388-392.

[DO85] Yvo Desmedt and Andrew Odlyzko, "A chosen text attack on the RSA cryptosystem and some discrete logarithm schemes", *Advances in Cryptology-CRYPTO 85*, H.C. Williams ed., LNCS 218, Springer-Verlag, pp. 516-522.

[EGS85] Shimon Even, Oded Goldreich and Adi Shamir, "On the security of ping-pong protocols when implemented using the RSA", *Advances in Cryptology-CRYPTO 85*, H.C. Williams ed., LNCS 218, Springer-Verlag, pp. 58-72.

[EvH92] Jan-Hendrik Evertse, Eugène van Heyst, "Which new RSA signatures can be computed from certain given RSA signatures?", *Journal of Cryptology*, **5** (1992), pp. 41-52.

[KaBa79] R. Kannan and A. Bachem, "Polynomial algorithms for computing the Smith and Hermite normal forms of an integer matrix", *SIAM Journal on Computing*, **8** (1979) pp. 499-507.

[RSA78] R.L. Rivest, A. Shamir, and L. Adleman, A Method for Obtaining Digital Signatures and Public Key Cryptosystems, *Comm. of the ACM* **21** (1978) pp. 120-126.