Essentially optimal interactive certificates in linear algebra

Jean-Guillaume Dumas, Erich Kaltofen

Essentially optimal interactive certificates in linear algebra*

Jean-Guillaume Dumas[†] Erich Kaltofen[‡]

January 17, 2014

Abstract

Certificates to a linear algebra computation are additional data structures for each output, which can be used by a—possibly randomized—verification algorithm that proves the correctness of each output. The certificates are essentially optimal if the time (and space) complexity of verification is essentially linear in the input size N, meaning N times a factor $N^{o(1)}$, i.e., a factor $N^{\eta(N)}$ with $\lim_{N\to\infty} \eta(N) = 0$.

We give algorithms that compute essentially optimal certificates for the positive semidefiniteness, Frobenius form, characteristic and minimal polynomial of an $n \times n$ dense integer matrix A. Our certificates can be verified in Monte-Carlo bit complexity $(n^2 \log \|A\|)^{1+o(1)}$, where $\log \|A\|$ is the bit size of the integer entries, solving an open problem in [Kaltofen, Nehring, Saunders, Proc. ISSAC 2011] subject to computational hardness assumptions.

Second, we give algorithms that compute certificates for the rank of sparse or structured $n \times n$ matrices over an abstract field, whose Monte Carlo verification complexity is 2 matrix-times-vector products + $n^{1+o(1)}$ arithmetic operations in the field. For example, if the $n \times n$ input matrix is sparse with $n^{1+o(1)}$ non-zero entries, our rank certificate can be verified in $n^{1+o(1)}$ field operations.

All our certificates are based on interactive verification protocols with the interaction removed by a Fiat-Shamir identification heuristic. The validity of our verification procedure is subject to standard computational hardness assumptions from cryptography.

Computing MethodologiesSymbolic and Algebraic Manipulation

[†]Université de Grenoble. Laboratoire LJK, umr CNRS, INRIA, UJF, UPMF, GINP. 51, av. des Mathématiques, F38041 Grenoble, France. Jean-Guillaume.Dumas@imag.fr, ljk.imag.fr/membres/Jean-Guillaume.Dumas/

[‡]Department of Mathematics. North Carolina State University. Raleigh, NC 27695-8205, USA. kaltofen@math.ncsu.edu, www.kaltofen.us

*This research was supported in part by the Agence Nationale pour la Recherche under Grant ANR-11-BS02-013 HPAC (Dumas) and the National Science Foundation under Grant CCF-1115772 (Kaltofen).

1 Introduction

Suppose you want to externalize your computations to cloud services. Prior to payment of the services, it would be desirable to verify that the returned result has been correctly computed by the cloud servers. This model is economically viable only if the verification process requires less resources than the computation itself. It is therefore important to design *certificates* that can be used to verify a result at a lower cost than that of recomputing it.

For instance, a primality certificate is a formal proof that a number is prime. In [16], primality is assessed by presenting a primitive root and the factorization of $m-1$. The latter can be checked fast by remultiplying, and then primitivity is polynomially checkable.

In linear algebra our original motivation is related to sum-of-squares. By Artin's solution to Hilbert 17th Problem, any polynomial inequality $\forall \xi_1, \ldots, \xi_n \in \mathbb{R}, f(\xi_1, \ldots, \xi_n) \geq g(\xi_1, \ldots, \xi_n)$ can be proved by a fraction of sum-of-squares:

$$\exists u_i, v_j \in \mathbb{R}[x_1, \ldots, x_n], f - g = \left(\sum_{i=1}^{\ell} u_i^2\right) / \left(\sum_{j=1}^{m} v_j^2\right) \tag{1}$$

Such proofs can be used to establish global minimality for

$g = \inf_{\xi_v \in \mathbb{R}} f(\xi_1, \ldots, \xi_n)$ and constitute certificates in non-linear global optimization. A symmetric integer matrix $W \in \mathbb{SZ}^{n \times n}$ is positive semidefinite, denoted by $W \succeq 0$, if all its eigenvalues, which then must be real numbers, are non-negative. Then, a certificate for positive semidefiniteness of rational matrices constitutes, by its Cholesky factorizability, the final step in an exact rational sum-of-squares proof, namely

$$\begin{aligned} &\exists e \geq 0,\ W^{[1]} \succeq 0,\ W^{[2]} \succeq 0,\ W^{[2]} \neq \mathbf{0}: \\ &\quad (f-g)(x_1, \ldots, x_n) \cdot (m_e(x_1, \ldots, x_n)^T W^{[2]} m_e(x_1, \ldots, x_n)) = \\ &\qquad\qquad m_d(x_1, \ldots, x_n)^T W^{[1]} m_d(x_1, \ldots, x_n), \end{aligned} \tag{2}$$

where the entries in the vectors m_d, m_e are the terms occurring in u_i, v_j in (1). In fact, (2) is the semidefinite program that one solves.

Thus arose the question how to give possibly probabilistically checkable certificates for linear algebra problems. In [13] the certificates are restricted to those that are checkable in essentially optimal time, that is, in bit complexity $(n^2 \log \|W\|)^{1+o(1)}$, where $\log \|W\|$ is the bit size of the entries in W. Quadratic time is feasible because a matrix multiplication AB can be certified by the resulting product matrix C via Rusin Freivalds's [9] (see also [14]) probabilistic check: check $A(Bv) = Cv$ for a random vector v.

Note that programs that check their results from [5] have the higher matrix-multiplication time complexity. In [13] a certificate for matrix rank was presented, based on Storjohann's Las Vegas rank algorithm [19], but matrix positive semidefiniteness remained open. Also the presented certificate for the rank did not take into account a possible structure in the matrix.

In the following we solve these two problems. In both cases, positive semidefiniteness and structured or blackbox matrices, our solution is to use either *interactive* certificates under the random oracle model, or heuristics under standard computational hardness assumptions from cryptography. Removing the cryptographic assumptions remains however a fundamental open problem. Providing certificates to other problems, such as the determinant or the minimal and characteristic polynomial of blackbox matrices, is also open.

In Section 2, we detail the different notions of certification that can be used and in particular the relaxation we make over the certificates of [13]: in the certificates presented here, we allow the verifier to provide the random bits used by the prover, in an interactive manner. We also present in this section the Fiat-Shamir derandomization heuristic that can turn any interactive certificate into a non-interactive heuristic certificate.

More precisely, the idea is to devise an interactive protocol for the random oracle model, and then to replace oracle accesses by the computation of an appropriately chosen function h [7, 2].

Then we first present in Section 3 an interactive certificate for the Frobenius normal form that can be verified in $O(n^{2+\epsilon}(\log \|A\|)^{1+\epsilon})$ binary operations for any $\epsilon > 0$, as in [13], but our new certificate also occupies an optimal space of $O(n^{2+\epsilon}(\log \|A\|)^{1+\epsilon})$ bits. This is an order of magnitude improvement over [13, Theorem 4]. This certificate can then be used as in the latter paper to certify the minimal and characteristic polynomial as well as positive semidefiniteness, while keeping the lower memory requirements. In the same section we also present another, stand-alone, characteristic polynomial certificate, which can also be used for positive semidefiniteness, with slightly smaller random evaluation points.

Finally in Section 4 we present a new certificate for the rank of sparse or structured matrices.

The certificate combines an interactive certificate of non-singularity, giving a lower bound to the rank, with an interactive certificate for an upper bound to the rank. Overall the interactive certificate for the rank requires only $2\Omega + n^{1+\epsilon}$ arithmetic operations over any coefficient domain, where Ω is the number of operations required to perform one matrix-times-vector product. For instance, if the matrix is sparse with only $n^{1+\epsilon}$ non-zero elements, then the certificate verification is essentially linear.

2 Notions of certificate

The ideas in this paper arise from linear algebra, probabilistic algorithms, program testing and cryptography.

We will in particular combine:

- the notions of certificates for linear algebra of Kaltofen et al. [13], themselves extending program checkers of Blum and Kannan [5] and randomized algorithms of Freivalds [9],
- with probabilistic interactive proofs of Babai [1] and Goldwasser et al. [11],

- as well as Fiat-Shamir heuristic [7, 2] turning interactive certificates into non-interactive heuristics subject to computational hardness.

We first recall some of these notions an then define in Section 2.3 what we mean by perfectly complete, sound and efficient interactive certificates.

2.1 Arthur-Merlin interactive proof systems

A proof usually has two parts, a theorem T and a proof Π, and the validity of the proof can be checked by a verifier V. Now, an *interactive proof*, or a $\sum$*-protocol*, is a dialogue between a prover P (or *Peggy* in the following) and a verifier V (or *Victor* in the following), where V can ask a series of questions, or challenges, q_1, q_2, ... and P can respond alternatively with a series of strings π_1, π_2, ..., the responses, in order to prove the theorem T. The theorem is sometimes decomposed into two parts, the hypothesis, or input, H, and the commitment, C. Then the verifier can accept or reject the proof: $V(H, C, q_1, \pi_1, q_2, \pi_2, \ldots) \in \{\text{accept}, \text{reject}\}$.

To be useful, such proof systems should satisfy *completeness* (the prover can convince the verifier that a true statement is indeed true) and soundness (the prover cannot convince the verifier that a false statement is true). More precisely, the protocol is *complete* if the probability that a true statement is rejected by the verifier can be made arbitrarily small. Similarly, the protocol is *sound* if the probability that a false statement is accepted by the verifier can be made arbitrarily small. The completeness (resp. soundness) is *perfect* if accepted (resp. rejected) statement are always true (resp. false).

It turns out that interactive proofs with perfect completeness are as powerful as interactive proofs [10]. Thus in the following, as we want to prove correctness of a result more than proving knowledge of it, we will only use interactive proofs with perfect completeness.

On the one hand, if a protocol is both perfectly complete and perfectly sound then it is deterministic. On the other hand, if at least one of completeness and soundness is not perfect, then the proof is probabilistic and correspond to Monte Carlo algorithms (always fast, probably correct).

Usually, in cryptology, the prover has infinite power and the verifier is polynomial time. In our setting, we will instead require that the verifier has lower computational complexity than any known algorithm computing the property.

2.2 Certificates in linear algebra

For Blum and Kannan [5] a program checker for a program P is itself a program C. For any instance I on which program P is run, C is run subsequently. C either certifies that the program P is correct on I or declares P to be buggy. There, the programs can be rerun on modified inputs, as in their matrix rank check, and thus might require more time to check their work than to do the work itself.

On the contrary, in [12, 13], a certificate for a problem that is given by input/output specifications is an input-dependent data structure and an algorithm that computes from that input and its certificate the specified

output, and that has lower computational complexity than any known algorithm that does the same when only receiving the input. Correctness of the data structure is not assumed but validated by the algorithm. With respect to interactive proofs, the input/output is related to the property to be proven together with the commitment. However, as no interaction is possible between the prover and the verifier, this amounts to using a single round protocol where the prover sends only the commitment and then the verifier accepts it or not.

In this paper we use a modified version where we allow interactive exchanges between the prover and the verifier but preserve the requirements on lower total complexity for the verifier. Moreover, we then can convert back these two-rounds protocol into one round protocols via Fiat-Shamir heuristic: hash the input and commitment with an unpredictable and universal hash function (such as a cryptographic hash function), to simulate the random challenges proposed by the verifier.

It turns out that it seems easier to design certificates that are interactive than to design directly single round certificates. This could be related to the power of the interactive proof system complexity class (IP) and the probabilistically checkable proofs (PCP).

2.3 Interactive certificates

As there exists ways to reduce interactive proofs with k rounds to interactive proofs with perfect completeness and 2 rounds, we will limit ourselves to these cases for our definition of interactive certificates.

More precisely, in the following we use interactive certificates of a given property, mainly as *two-rounds probabilistic $\sum$-protocols with perfect completeness*:

1. The prover of a property sends a commitment to the verifier.
2. The verifier sends back a (randomly sampled) challenge, potentially depending on both the property and the commitment.
3. The provers completes the protocol with a response convincing the verifier of the property.

In order to become an *interactive certificate*, this two round $\sum$-protocol should then satisfy soundness, perfect completeness and efficiency as follows:

i. The protocol is *perfectly complete*: a true statement will always be accepted by the verifier.

ii. The protocol is *sound*: the probability that a false statement will be accepted by the verifier can be made arbitrarily small.

iii. The protocol is *efficient*: the verifier has lower computational complexity than any known algorithm that computes the true statement when only receiving the input.

The interactive certificate can also be said to be *essentially optimal* when the verifier needs only time and space complexity of the same order of magnitude as the size of the input and output to verify the latter.

With this relaxed model, we are able in the following to improve on some space complexities for integer linear algebra problems and also on time complexities for some problems over generic domains, like the rank of blackbox matrices.

2.4 Fiat-Shamir heuristic derandomization into a single round

In a practical perspective (say when using a compiled library, rather than an interpreter; or when posting the certificate in question) it is not always possible for the verifier (a user wanting a result) to interact with the prover (the program).

Then, there is always the possibility to transform an interactive certificate into a non-interactive heuristic. Here we use the strong Fiat-Shamir heuristic [7, 2, 3], where the random challenge message of the verifier is replaced by a cryptographic hash of the property *and* the commitment message. In practice, the cryptographic hash can be used as a seed for a pseudo-randomly generated sequence that the prover can generate a priori. For an a posteriori verification, the verifier decides whether to accept or not the certificate, as in two rounds interactive protocols, but has also to check that the challenge used by the prover has really been generated using the input and commitment as seeds.

In this setting, breaking the protocol is somewhat equivalent to breaking the cryptographic hash function: finding a combination of input and false commitment that will be accepted by the verifier relates to knowing in advance some parts of the output of the hash function. See for instance Section 4.4 where breaking the protocol is equivalent to predicting the value of some bits in a hash, and that can for instance be used to factor integers if Blumb-Blum-Shub hash function is used.

Note that it is important to use the *strong* heuristic that uses a combination of both the input and the commitment for the hashing. See for instance Section 3.2 where we need the result itself to be part of the seed in order to obtain a correct certificate.

3 Reducing space with respect to one-round certificates over the integers

3.1 Interactive certificate for residue systems

In [13, Theorem 5], the given certificates for the rank and determinant of an integer matrix are essentially optimal whereas the certificates for the Frobenius normal form, the characteristic and minimal polynomial and positive semidefiniteness are not: they require residue systems that occupies cubic bit space whereas the input and results are only quadratic.

Those residue systems allow the verifier to check an integer matrix factorization ($A = LU$ for gaussian elimination or $A = SFS^{-1}$ for the Frobenius form) where the resulting factors are in general of cubic size (quadratic number of entries but each one with linear magnitude) via

Freivalds' certificate. The trick is to store these factorizations modulo many distinct primes. Then if the integer matrix factorization is not correct it means that $A - LU$ (resp. $A - SFS^{-1}$) is non zero. Therefore, from a bound on the maximal possible size of this difference (roughly cubic), it cannot be zero modulo a large number of primes. Consequently, if the set of distinct primes is larger than the bound, selecting a random prime p in the set and checking whether $A - LU$ (resp. $A - SFS^{-1}$) is zero modulo p would reveal the false statement with high probability.

Our idea here is to use several rounds interactive certificates: instead of storing the factorizations modulo many distinct primes, just compute them on demand of the verifier. The verifier has just to select random primes and the prover will respond with the factorization modulo these primes.

Theorem 1. *Let $A \in \mathbb{Z}^{n\times n}$. There exist an* interactive certificate *for the Frobenius normal form, the characteristic or minimal polynomial of A. The interactive certificate can be verified in $n^{2+o(1)}(\log \|A\|)^{1+o(1)}$ time and occupies $n^{2+o(1)}(\log \|A\|)^{1+o(1)}$ bit space.*

Proof. Use the same algorithm as in [13, Theorem 4] but replacing the random choice by the verifier of a given tuple (p, S_p, F_p, T_p) (where $T_p \equiv S_p^{-1} \mod p$) by the choice of a random prime p by the verifier and a response of a corresponding (S_p, F_p, T_p) modulo p by the prover. □

Corollary 1. *There exist a* non-interactive heuristic certificate *for the Frobenius normal form, the characteristic or minimal polynomial that occupy the same space and can be verified in the same time.*

Proof. We use Fiat-Shamir. The prover:

1. computes the integer Frobenius normal form F (or the characteristic or minimal polynomial) over the integers;
2. then he chooses a cryptographic hash function and a pseudo-random prime generator;
3. he computes the hash of the input matrix together with the result;
4. this hash is used as a seed for the pseudo-random prime generator to generate one (or a constant number of) prime number(s);
5. the prover finally produces the Frobenius normal form and the change of basis modulo that prime(s).

The certificate is then composed of the input, the output, the hash function, the pseudo-random prime generator, the generated prime numbers and the associated triples (S_p, F_p, T_p).

The verifier then:

1. checks that the hash function and the pseudo-random prime generator are well-known, cryptographically secure, functions;
2. checks that he can recover the primes via hashing the combination of the input and the output;
3. and verifies the zero equivalence modulo p of $(F - F_p) \mod p$, $(S_pT_p - I) \mod p$ and $(S_pF_pT_p - A) \mod p$.

□

3.2 Direct interactive certificate for the characteristic polynomial and positive definiteness of integer matrices

In [13], the certificate for characteristic polynomial occupies roughly $n^{3+o(1)}$ bit space as it requires the Frobenius matrix normal form with a similarity residue system with primes bounded by $O(n(\log(n) + \log \|A\|))$.

As shown in Theorem 1, with an interactive certificate and a random oracle for the choice of prime numbers of the latter size, this yields an interactive certificate with only $n^{2+o(1)}$ bit space requirements.

We propose in the following Figure 1 a simpler certificate, still relying on the determinant certificate, but with evaluation points bounded only by $O(n)$. This gives a similar but smaller $o(1)$ factor in the complexity.

	Peggy		*Victor*
Input		$A \in \mathbb{Z}^{n\times n}$	
Commitment	$g \in \mathbb{Z}[X] = charpoly_A$	$1:\ g(X) \rightarrow$	$degree(g) \stackrel{?}{=} n$
Challenge		$2:\ \lambda \leftarrow$	$\lambda \in \mathbb{Z}$
Response	$\delta \in \mathbb{Z} = det(\lambda I - A)$	$3:\ \delta \rightarrow$	$\delta \stackrel{?}{=} g(\lambda)$
	$C : Cert(\delta = det(\lambda I - A))$	$4:\ C \rightarrow$	$\delta \stackrel{?}{=} det(\lambda I - A)$

Figure 1: Interactive certificate for the characteristic polynomial

Theorem 2. *For $A \in \mathbb{Z}^{n\times n}$, the interactive certificate of Figure 1 for the characteristic polynomial is sound, perfectly complete and the number of operations performed by the verifier, as well as the bit space required to store this certificate, is bounded by $n^{2+o(1)}(\log \|A\|)^{1+o(1)}$.*

Proof. For the determinant certificate we use [13, Theorem 5] of which complexity matches that of the present theorem.

If Peggy and Victor are honest then the definition of the characteristic polynomial yields $charpoly_A = det(XI - A)$ and thus the protocol is perfectly complete.

If Peggy is dishonest then $g - charpoly_A$ being of degree at most n, it has at most n roots. Thus if Victor samples random elements among the first say cn integers, *after* the commitment g, the probability that Victor accepts the certificate is less than $1/c$. If the protocol is repeated k times with independent draws of λ, then the probability that Victor accepts it k times is lower than $\left(\frac{1}{c}\right)^k$ and therefore the protocol is sound.

For the complexity, one chooses a constant $c > 2$ so that λ has $O(\log(n))$ bits. Thus δ, as the determinant of $\lambda I - A$, is bounded by Hadamard's bound to $O(n \log(\|A\| + n))$ bits. With Horner evaluation and Chinese remaindering, the check $g(\lambda) \stackrel{?}{=} \delta$ can thus be performed in

less than $O(n^2 \log(\|A\| + n))$ operations. This is within the announced bound. □

Corollary 2. *Let A be an $n \times n$ symmetric matrix having minors bound H_A of bit length $\log_2(H_A) = n^{1+o(1)}$. The signature of A can be verified by an interactive certificate in $n^{2+o(1)}$ binary operations with a $n^{2+o(1)}$ bit space characteristic polynomial certificate. Thus the same certificate serves for positive or negative definiteness or semidefiniteness.*

Proof. We just use the certificate of [13, Corollary 1] but replace their characteristic polynomial certificate by the interactive one of Figure 1 and Theorem 2. □

4 Interactive certificate for the rank of sparse matrices

Now we turn to matrices over any domain and count arithmetic operations instead of bit complexity.

For the sake of simplicity we will use the notation $\mathbb{F}$ as for finite fields but the results are valid over any abstract field, provided that the random sampling is done on a finite subset S of the domain.

We improve on $O(n^2)$ certificates for the rank (given with say an LU factorization), when the matrix is sparse, structured or given as blackbox. That is to say when the product of the matrix by a vector is of complexity Ω less than $2n^2$. If the matrix is given as a blackbox, then the only possible operation with the matrix is the latter matrix-times-vector product.

For a matrix of rank r, if Ω is the cost of one of those matrix-times-vector product, the blackbox certificates of [17] would also require $O(nr)$ extra arithmetic operations and at least $O(r)$ extra matrix-times-vector products for a total of $O(r\Omega + nr)$ arithmetic operations.

In the following we show that it is possible to reduce the time and space complexity bounds of verifying certificates for the rank of blackbox matrices to only $2\Omega + n^{1+o(1)}$ arithmetic operations. This is essentially optimal, e.g. for sparse matrices, as reading and storing a matrix of dimensions $n \times n$ should also require $O(\Omega + n)$ operations.

We proceed in two steps. First we certify that there exists an $r \times r$ non-singular minor in the matrix. Second, we precondition the matrix so that it is of generic rank profile and exhibit a vector in the null-space of the leading $(r+1) \times (r+1)$ minor of the preconditioned matrix.

4.1 Certifying non-singularity

Theorem 3. *Let S be a finite subset of $\mathbb{F}$ with at least two distinct elements. For $A \in \mathbb{F}^{n \times n}$, which matrix-times-vector products costs Ω operations in $\mathbb{F}$, the interactive certificate of Figure 2 for non-singularity is sound, perfectly complete and the number of arithmetic operations performed by the verifier is bounded by $\Omega + n$.*

Proof. If Peggy and Victor are honest, then Peggy can solve the system with an invertible matrix and provide $w = A^{-1}b$ to Victor. Therefore the protocol is perfectly complete.

If Peggy is dishonest, then it means that A is singular. Therefore, it means that the rank of A is at most $n-1$.

We use, e.g., Gaussian elimination to get $A = PLUQ$, where P and Q are permutation matrices, L is unit invertible lower triangular and U is upper triangular. As the rank of A is at most $n-1$, U is of the form $\left[\begin{smallmatrix} U_1 & U_2 \\ 0 & 0 \end{smallmatrix}\right]$ where $U_1 \in \mathbb{F}^{(n-1)\times(n-1)}$ is upper triangular. One then sees that making the system inconsistent is equivalent to setting to zero at least the last entry of $L^{-1}P^{-1}b$ in $\mathbb{F}^n$. Thus, with probability at least $1 - 1/|S|$, the challenge vector proposed by Victor makes the system inconsistent. In the latter case, Peggy will never be able to find a solution to the system.

Thus Victor can accept the certificate of Peggy only when he has randomly found a consistent vector. The probability that this happens k times with k independent selections of b is bounded by $\frac{1}{|S|^k}$. Therefore, when the matrix is singular, Victor can accept repeated applications of the protocol only with negligible probability and the protocol is sound.

For the complexity, Victor needs to perform one matrix-times-vector product with A, of arithmetic complexity Ω. Victor also needs to produce a random vector of size n of elements in S.

□

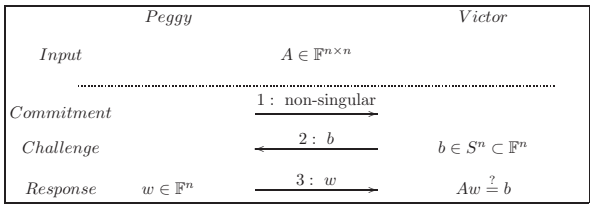

Figure 2: Blackbox interactive certificate of non-singularity

4.2 Certifying an upper bound for the rank

For an upper bound, we precondition $A \in \mathbb{F}^{m\times n}$ of rank r so that the leading $r \times r$ minor of the preconditioned matrix is non-zero and then present a non-zero vector in the nullspace of the $r+1$ leading minor. We use the butterfly probabilistic preconditioners of [6, Theorem 6.3] that can precondition an $n \times n$ matrix of rank r so that the first r rows of the preconditioned matrix become linearly independent with high probability. We denote by $\mathbb{B}_S^{n\times n}$ the set of such butterfly networks composed by less than $n(\log_2(n))$ switches of the form $\begin{bmatrix} 1 & \alpha \\ 1 & 1+\alpha \end{bmatrix}$, for $\alpha \in S \subset \mathbb{F}$. Choosing a random butterfly reduces to choosing an element α, a row index and a column index for each of its switches.

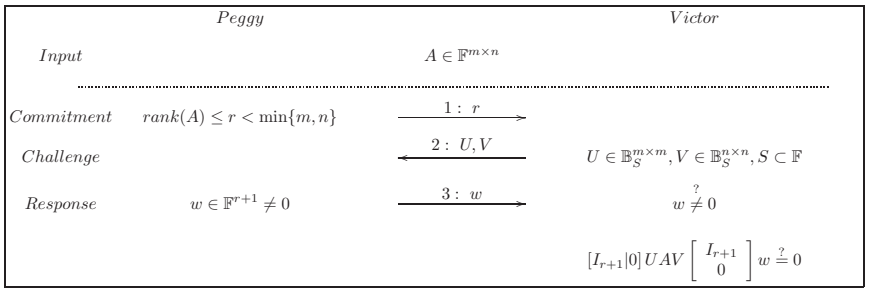

Figure 3: Blackbox interactive certificate for an upper bound to the rank

Theorem 4. *Let $A \in \mathbb{F}^{m\times n}$, which matrix-times-vector products costs Ω operations in $\mathbb{F}$ and let S be a finite subset of $\mathbb{F}$ with $|S| > 2\min\{m,n\}(\lceil\log_2(m)\rceil + \lceil\log_2(n)\rceil)$. The interactive certificate of Figure 3 proving an upper bound for the rank of A is sound, perfectly complete and the number of arithmetic operations performed by the verifier is bounded by*
$\Omega + (m+n)^{1+o(1)}$.

Proof. If Peggy is honest this means that the rank of A is upper bounded by $r < \min\{m,n\}$. Thus the rank of $M = [I_{r+1}|0]\, UAV \begin{bmatrix} I_{r+1} \\ 0 \end{bmatrix} \in \mathbb{F}^{(r+1)\times(r+1)}$ is also upper bounded by r. Therefore, there exist at least one non-zero vector w in the nullspace of M. Hence Peggy can produce it and the protocol is perfectly complete.

If Peggy is dishonest, this means that the rank of A is at least $r+1$. Now, from [6, Theorem 6.3], the butterfly preconditioner $U \in \mathbb{B}_S^{m\times m}$ will make the first $r+1$ rows of UA linearly dependent with probability less than $\frac{(r+1)\lceil\log_2(m)\rceil}{|S|}$. Similarly the butterfly preconditioner $V \in \mathbb{B}_S^{n\times n}$ will make the first $r+1$ columns of AV linearly dependent with probability less than $\frac{(r+1)\lceil\log_2(n)\rceil}{|S|}$. Overall the $(r+1)\times(r+1)$ leading principal minor of UAV will be non-zero with probability at least $1 - \frac{(r+1)(\lceil\log_2(m)\rceil+\lceil\log_2(n)\rceil)}{|S|} \geq 1 - \frac{\min\{m,n\}(\lceil\log_2(m)\rceil+\lceil\log_2(n)\rceil)}{|S|} \geq \frac{1}{2}$. In this case the minor is invertible and Peggy will never be able to produce a non-zero vector in its kernel. The only possibility for Victor to accept the certificate is thus that the leading minor is zero and the probability that this happens k times with k independent selections of U and V is thus bounded by $\frac{1}{2^k}$. Thus Victor can accept repeated applications of the protocol only with negligible probability and the protocol is sound.

For the complexity, we know from [6, Theorem 6.2] that butterflies of respective sizes $m\lceil\log_2(m)\rceil/2$ and $n\lceil\log_2(n)\rceil/2$ are sufficient.

Victor thus needs to produce $(m\lceil\log_2(m)\rceil/2 + n\lceil\log_2(n)\rceil/2)$ random elements in S and $O(\log_2(m) + \log_2(n))$ random bits for the row and columns indices. Then the successive applications of U, A and V to a vector cost no more than $3m\lceil\log_2(m)\rceil/2 + \Omega + 3n\lceil\log_2(n)\rceil/2$ arithmetic operations. □

Remark 1. *Over small fields,*

it might not be possible to find a sufficiently large subset S. Then one can use extension fields or change the preconditioners. For instance, $\tilde{U} \in \mathbb{W}^{(r+1)\times m}$, and respectively $\tilde{V} \in \mathbb{W}^{n\times(r+1)}$, can be taken as sparse matrix preconditioners, as in [20] (see also [6, Corollary 7.3]), and replace respectively $[I_{r+1}|0]\,U$ and $V \begin{bmatrix} I_{r+1} \\ 0 \end{bmatrix}$. They are randomly sampled with the Wiedemann distribution, denoted by $\mathbb{W}$, and have thus not more than $2n(2+\log_2(n))^2$ non zero entries with probability at least $1/8$, [20, Theorem 1].

4.3 Blackbox interactive certificate for the rank

Now we can propose a complete certificate for the rank.

If the matrix is full rank, then it is sufficient to produce a certificate for a maximal lower bound. Otherwise, it will use a non-singularity certificate on a sub-matrix of dimension $r \times r$ together with an upper bound certificate: for a matrix $A \in \mathbb{F}^{m\times n}$,

1. Compute $I^{(r)} \in \mathbb{F}^{m\times m}$ and $J^{(r)} \in \mathbb{F}^{n\times n}$ row and column subsets of A such that $I^{(r)}AJ^{(r)}$ is non singular and use the non-singularity certificate of Figure 2 on the latter. This provides a certified lower bound for the rank of A.
2. Use the certificate for an upper bound r of the rank of A of Figure 3.
3. With certified lower bound r and upper bound r, the rank is certified.

Using Theorems 3 and 4, we have proven:

Corollary 3. *For $A \in \mathbb{F}^{m\times n}$, which matrix-times-vector products costs Ω operations in $\mathbb{F}$ and let S be a finite subset of $\mathbb{F}$ with $|S| > 2\min\{m,n\}(\lceil\log_2(m)\rceil + \lceil\log_2(n)\rceil)$. The above $\sum$-protocol provides an interactive certificate for the rank of A. This interactive certificate is sound, perfectly complete and the number of arithmetic operations performed by the verifier is bounded by*

$$2\Omega + (m+n)^{1+o(1)}.$$

4.4 Reducing breaking the random oracle to factorization

Now we look at the derandomization of the previous certificates using the strong Fiat-Shamir heuristic, see Section 2.4, where the random challenge messages of the verifier are replaced by a cryptographic hash of the property and the commitment messages.

First, it is proven in [15] that this methodology always produces digital signature schemes that are provably secure against chosen message attacks in the "Random Oracle Model" – when the hash function is modeled by a random oracle. In other words, it is equivalent for a dishonest Peggy to e.g. produce consistent systems for singular matrices or to break the random oracle.

Second, we can, e.g., use the Blumb-Blum-Shub perfect random generator [4]: it transforms a seed x_0 (for us the matrix) into a bit string

$b_1, \ldots, b_k$ with $b_i = x_i \mod 2$; $x_{i+1} = x_i^e \mod N$ for some RSA public key (e, N). It is for instance shown in [8] that knowing a number of b_i's polynomial in the size of N, say a number $B_N = O(\log^\gamma(N))$ bits, enables one to factor it. Now, we show next that forging a consistency certificate $Ax \stackrel{?}{=} b$ is equivalent to predicting the value of at least one bit of the random right-hand side vector b.

Lemma 1. *Forging the non-singularity certificate fixes at least one bit of the random right-hand side vector.*

Proof. A singular matrix $A \in \mathbb{F}^{n\times n}$ has rank at most $n-1$. Write it as $A = P\left[\begin{smallmatrix} 0 & 0 \\ 0 & L \end{smallmatrix}\right] UQ$ where P and Q are permutation matrices, $L \in \mathbb{F}^{(n-1)\times(n-1)}$ is lower triangular and U is unit invertible upper triangular. Then if $Aw = b$, it means that $\left[\begin{smallmatrix} 0 & 0 \\ 0 & L \end{smallmatrix}\right] z = P^{-1}b$ for $z = UQw$. Therefore, the first entry of $P^{-1}b$ must be zero. □

Therefore, we fix the RSA modulus N and require as a certificate that the consistency check is repeated B_N times. When the protocol is repeated with Fiat-Shamir derandomization, we use as successive random vectors b, the hash of

the input and the previous iteration. If Peggy can find a matrix A of dimension n (polynomial in the size of N) for which she can forge the B_N repeated applications of the certificate, then

she can predict B_N bits of the Blumb-Blum-Shub hashes in $O(B_N(\Omega + n^{1+o(1)}))$ operations. Thus Peggy can factor N in polynomial time.

Acknowledgments

We thank Brice Boyer, Clément Pernet and Jean-Louis Roch for useful discussions on this subject.

References

[1] László Babai. Trading group theory for randomness. In Sedgewick [18], pages 421–429. http://dx.doi.org/10.1145/22145.22192.

[2] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In Victoria Ashby, editor, *Proceedings of the 1st ACM Conference on Computer and Communications Security*, pages 62–73, Fairfax, Virginia, November 1993. ACM Press. http://www-cse.ucsd.edu/users/mihir/papers/ro.pdf.

[3] David Bernhard, Olivier Pereira, and Bogdan Warinschi. How not to prove yourself: Pitfalls of the fiat-shamir heuristic and applications to helios. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT'12*, volume 7658 of *Lecture Notes in Computer Science*, pages 626–643. Springer, 2012. http://www.uclouvain.be/crypto/services/download/publications.pdf.87e67d05ee05000b.6d61696e2e

[4] Lenore Blum, Manuel Blum, and Michael Shub. Comparison of two pseudo-random number generators. In David Chaum, Ronald L. Rivest, and Alan T. Sherman, editors, *Advances in Cryptology—CRYPTO'82*, pages 61–78. Plenum Press, New York and London, 1983, 23–25 August 1982. http://dx.doi.org/10.1007/978-1-4757-0602-4_6.

[5] Manuel Blum and Sampath Kannan. Designing programs that check their work. *Journal of the ACM*, 42(1):269–291, January 1995. http://www.icsi.berkeley.edu/pubs/techreports/tr-88-009.pdf.

[6] Li Chen, Wayne Eberly, Erich L. Kaltofen, B. David Saunders, William J. Turner, and Gilles Villard. Efficient matrix preconditioners for black box linear algebra. *Linear Algebra and its Applications*, 343-344:119–146, 2002. http://www.math.ncsu.edu/~kaltofen/bibliography/02/CEKSTV02.pdf.

[7] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In A. M. Odlyzko, editor, *Advances in Cryptology—CRYPTO'86*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194. Springer-Verlag, 1987, 11–15 August 1986. http://www.cs.rit.edu/~jjk8346/FiatShamir.pdf.

[8] Roger Fischlin and Claus P. Schnorr. Stronger security proofs for RSA and Rabin bits. In *Advances in Cryptology - EUROCRYPT'97*, volume 1233 of *Lecture Notes in Computer Science*, pages 267–279, 1997. http://dx.doi.org/10.1007/3-540-69053-0_19.

[9] Rūsiņš Freivalds. Fast probabilistic algorithms. In J. Bečvář, editor, *Mathematical Foundations of Computer Science 1979*, volume 74 of *Lecture Notes in Computer Science*, pages 57–69, Olomouc, Czechoslovakia, September 1979. Springer-Verlag. http://dx.doi.org/10.1007/3-540-09526-8_5.

[10] Martin Furer, Oded Goldreich, Yishay Mansour, Michael Sipser, and Stathis Zachos. On completeness and soundness in interactive proof systems. In Silvio Micali, editor, *Randomness and Computation*, volume 5, pages 429–442. Advances in Computing Research, JAI Press, Greenwich, Connecticut, 1989. http://www.wisdom.weizmann.ac.il/~oded/PS/fgmsz.ps.

[11] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof-systems. In Sedgewick [18], pages 291–304. http://dx.doi.org/10.1145/22145.22178.

[12] Erich L. Kaltofen, Bin Li, Zhengfeng Yang, and Lihong Zhi. Exact certification in global polynomial optimization via sums-of-squares of rational functions with rational coefficients. *Journal of Symbolic Computation*, 47(1):1–15, January 2012. http://www.math.ncsu.edu/~kaltofen/bibliography/09/KLYZ09.pdf.

[13] Erich L. Kaltofen, Michael Nehring, and B. David Saunders. Quadratic-time certificates in linear algebra. In Anton Leykin, editor, *ISSAC'2011, Proceedings of the 2011 ACM International Symposium on Symbolic and Algebraic Computation, San Jose,*

California, USA, pages 171–176. ACM Press, New York, June 2011. http://www.math.ncsu.edu/~kaltofen/bibliography/11/KNS11.pdf.

[14] Tracy Kimbrel and Rakesh Kumar Sinha. A probabilistic algorithm for verifying matrix products using $O(n^2)$ time and $\log_2 n + O(1)$ random bits. *Information Processing Letters*, 45(2):107–110, February 1993. ftp://trout.cs.washington.edu/tr/1991/08/UW-CSE-91-08-06.pdf.

[15] David Pointcheval and Jacques Stern. Security proofs for signature schemes. In Ueli Maurer, editor, *Advances in Cryptology—EUROCRYPT'96*, volume 1070 of *Lecture Notes in Computer Science*, pages 387–398. Springer-Verlag, 12–16 May 1996. http://www.di.ens.fr/~pointche/Documents/Papers/1996_eurocrypt.pdf.

[16] Vaughan R. Pratt. Every prime has a succinct certificate. *SIAM Journal on Computing*, 4(3):214–220, September 1975. http://boole.stanford.edu/pub/SucCert.pdf.

[17] B. David Saunders, Arne Storjohann, and Gilles Villard. Matrix rank certification. *Electronic Journal of Linear Algebra*, 11:16–23, 2004. http://perso.ens-lyon.fr/gilles.villard/BIBLIOGRAPHIE/PDF/rank-certif.pdf.

[18] Robert Sedgewick, editor. *STOC '85, ACM Symposium on Theory of Computing, Providence, Rhode Island, USA*. ACM Press, New York, May 1985.

[19] Arne Storjohann. Integer matrix rank certification. In John P. May, editor, *ISSAC'2009, Proceedings of the 2009 ACM International Symposium on Symbolic and Algebraic Computation, Seoul, Korea*, pages 333–340, July 2009. https://cs.uwaterloo.ca/~astorjoh/issac09.pdf.

[20] Douglas H. Wiedemann. Solving sparse linear equations over finite fields. *IEEE Transactions on Information Theory*, 32(1):54–62, January 1986. http://dx.doi.org/10.1109/TIT.1986.1057137.