First Study for Ramp Secret Sharing Schemes Through Greatest Common Divisor of Polynomials

Gerasimos C. Meletiou, Dimitrios S. Triantafyllou, and Michael N. Vrahatis

Abstract A ramp secret sharing scheme through greatest common divisor of polynomials is presented. Verification and shelf correcting protocols are also developed. The proposed approach can be implemented in a hybrid way using numerical and symbolical arithmetic. Numerical examples illustrating the proposed sharing schemes are also given.

1 Introduction

A (t, n)-threshold secret sharing scheme is a method in which the dealer distributes the secret to n participants [2]. In this scheme any t participants, $1 \leqslant t \leqslant n$, can cooperate and retrieve the secret but any $t - 1$ participants cannot reconstruct the secret. An (s, t, n)-threshold ramp scheme is a generalization of a threshold secret sharing scheme using two parameters. Namely, the value s which determines the lower threshold and t which is the upper threshold. In a ramp scheme, any t (or more than t) of the n players can compute the secret (exactly as in a (t, n)-threshold scheme). It is also required that no subset of s (or less than s) players can determine any information about the secret. We note that a $(t-1, t, n)$-ramp scheme is exactly the same as a (t, n)-threshold scheme. The parameters of a ramp scheme satisfy the conditions $0 \leqslant s < t \leqslant n$. For more information see [3, 11]. A ramp scheme is

G. C. Meletiou (✉)
University of Ioannina, School of Agriculture, Arta, Greece
e-mail: gmelet@uoi.gr

D. S. Triantafyllou
Department of Mathematics and Engineering Sciences, Hellenic Military Academy, Vari, Greece
e-mail: dtriant@sse.gr

M. N. Vrahatis
Computational Intelligence Laboratory (CILab), Department of Mathematics, University of Patras, Patras, Greece
e-mail: vrahatis@math.upatras.gr

N. J. Daras, T. M. Rassias (eds.), *Computational Mathematics and Variational Analysis*, Springer Optimization and Its Applications 159,
https://doi.org/10.1007/978-3-030-44625-3_14

essentially a non-perfect secret sharing scheme. Ramp schemes are useful because they can achieve a high information rate.

In Section 2 the proposed ramp secret sharing scheme and two self-correcting protocols are presented based on the greatest common divisor (GCD) of polynomials. In Section 3 a numerical linear algebra technique for computing the GCD of polynomials through Sylvester matrices is presented. In Section 4 an example evaluating the proposed method is presented. The two self-correcting protocols are also evaluated. In Section 5 a synopsis and concluding remarks are given.

2 Ramp Secret Sharing Scheme

Throughout this paper, it is assumed that all participants are cooperating giving the real data to each other. Also, verification protocols are given to tackle the case where an error in a cooperation is appeared.

Cryptographic Scheme Let D be the dealer, $P_1, P_2, \ldots, P_n$ be n participants, $s(x) := \prod_{i=1}^{n} m_i(x)$ be the secret, where $m_i(x)$, $i = 1, 2, \ldots, n$ are n polynomials of degree d_{m_i}, $i = 1, 2, \ldots, n$ such that for $i \neq j$, $i, j = 1, 2, \ldots, n$ the $m_i(x)$, $m_j(x)$ are coprime and $d(x)$ be a polynomial known by the dealer, where $d(x)$, $m_i(x)$ are coprime for all $i = 1, 2, \ldots, n$. Each participant P_i, $i = 1, 2, \ldots, n$ receives the following information (share) from the dealer D:

$$\begin{aligned} p_1(x) &= d(x) \cdot m_2(x) \cdots m_n(x), \\ p_2(x) &= d(x) \cdot m_1(x) \cdot m_3(x) \cdots m_n(x), \\ &\vdots \\ p_i(x) &= d(x) \cdot m_1(x) \cdots m_{i-1}(x) \cdot m_{i+1}(x) \cdots m_n(x), \\ &\vdots \\ p_n(x) &= d(x) \cdot m_1(x) \cdots m_{n-1}(x). \end{aligned} \tag{1}$$

Thus,

$$p_i(x) = d(x) \cdot \prod_{\substack{j=1 \\ j \neq i}}^{n} m_i(x), \qquad i = 1, 2, \ldots, n.$$

Remark 1 In a future work our scope is to study the polynomials over fields in such a way that the factorization will not be feasible.

Remark 2 The participant P_i knows the whole product $p_i(x)$, as a polynomial of degree d_{m_i} $i = 1, 2, \ldots, n$ but he does not know the following factorization:

$$d(x) \cdot m_1(x) \cdots m_{i-1}(x) \cdot m_{i+1}(x) \cdots m_n(x).$$

Remark 3 In general, a direct corollary of the fundamental theorem of algebra [13] states that a polynomial can be factorized over the complex domain into a product $a_n(x - r_1)(x - r_2)\cdots(x - r_n)$, where a_n is the leading coefficient and $r_1, r_2, \ldots, r_n$ are all of its n complex roots. On the other hand, it is well known that, for polynomials of degrees more than four, no general closed-form formulas for their roots exist. For these cases we can apply various root-finding algorithm for the approximation of the roots of a polynomial. This approximation is not an easy task and it is depended on the inner tolerance that it will be used during the floating point operations. Specifically, for a polynomial of degree n the required bit operations are $O\big(n^{12} + n^9\big(\log(|p|)\big)^3\big)$, where $p = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_0$ is the polynomial and $\left|\sum_{i=0}^{n} a_i\, x^i\right| = \left(\sum_{i=0}^{n} a_i^2\right)^{1/2}$ for a polynomial with real coefficients [6]. Victor Pan in 2002 [9] presented almost optimal algorithms for numerical factorization of univariate polynomials, while Sagraloff and Mehlhorn in 2016 [10] introduced a hybrid of the Descartes method and Newton iteration which is in comparable complexity with Pan's algorithms.

In our case, after approximating all the roots of $p_i(x)$ the participant P_i may retrieve the secret by computing all the combinations of the roots. The scheme can be improved and become more robust by giving the dealer to participants different inner tolerances in order the participants not to be able to approximate all the roots. In that case the scheme looses its self-correcting protocols.

Theorem 1 *If all participants* $P_1,\ P_2, \ldots,\ P_n$ *are cooperating, they can derive the secret* $s(x)$.

Proof Suppose that the two participants P_i and P_j, $1 \leqslant i,\ j \leqslant n,\ i \neq j$ are cooperating. Participant P_i knows the following information from the dealer:

$$p_i(x) = d(x) \cdot m_1(x) \cdots m_{i-1}(x) \cdot m_{i+1}(x) \cdots m_n(x),$$

while participant P_j knows the following information from the dealer:

$$p_j(x) = d(x) \cdot m_1(x) \cdots m_{j-1}(x) \cdot m_{j+1}(x) \cdots m_n(x).$$

The GCD of the previous two polynomials is given as follows:

$$\gcd\big\{p_i(x),\ p_j(x)\big\} := g_{i,j}(x) = d(x) \cdot \prod_{\substack{k=1 \\ k \neq i,j}}^{n} m_i(x).$$

Then,

i. participant P_i finds out the identity $m_j(x)$ of participant P_j by dividing $p_i(x)$ with the GCD: $g_{i,j}(x)$: $p_i(x)/g_{i,j}(x) = m_j(x)$ and
ii. participant P_j finds out the identity $m_i(x)$ of participant P_i by dividing $p_j(x)$ with the GCD: $g_{i,j}(x)$: $p_j(x)/g_{i,j}(x) = m_i(x)$.

In addition, participant P_i informs participant P_j about his/her identity m_j and vice versa. Thus, both participants P_i and P_j know $m_i(x)$, $m_j(x)$ and the whole product $d(x) \cdot \prod_{j=1}^{n} m_i(x)$.

If all the n participants cooperate, then by revealing their identities they are able to derive the secret $s(x) = \prod_{i=1}^{n} m_i(x)$. □

Assume now that two participants, e.g., P_1 and P_2 decide to cooperate in order to compute a part of the secret. They can derive $m_1(x)$ and $m_2(x)$ and the Least Common Multiple (LCM) of them:

$$r(x) := \operatorname{lcm}\{p_1(x),\ p_2(x)\} = d(x) \cdot \prod_{i=1}^{n} m_i(x) = d(x) \cdot s(x),$$

as a polynomial. The secret $s(x)$ divides $r(x)$ and is divided by $m_1(x) \cdot m_2(x)$, thus

$$m_1(x) \cdot m_2(x)/s(x)/\operatorname{lcm}\{p_1(x),\ p_2(x)\}. \tag{2}$$

In other terms the shares $p_1(x)$ and $p_2(x)$ define a restriction for the secret $s(x)$. That means that from the shares of the participants P_1 and P_2, a partial information for the secret $s(x)$ can be derived (ramp scheme). In order to compute all the roots of their polynomials they have to use polynomial root-finding algorithms of complexity mentioned above.

More general, assume that k participants cooperate in order to compute a part of the secret, $1 < k < n$ and without loss of generality let $P_1, P_2, \ldots, P_k$ be the participants. They can compute $m_1(x)$, $m_2(x), \ldots, m_k(x)$, $\operatorname{lcm}\{p_1(x),\ p_2(x), \ldots, p_k(x)\} = d(x) \cdot \prod_{i=1}^{n} m_i(x) = d(x) \cdot s(x)$ as polynomial and its roots and they can recognize the roots of $m_i(x),\ i = 1, 2, \ldots, k$ as roots of the secret. Therefore, we have the following condition:

$$\prod_{i=1}^{k} m_i(x)/s(x)/\operatorname{lcm}\{p_1(x),\ p_2(x), \ldots, p_k(x)\}. \tag{3}$$

The "division interval" becomes more narrow since $m_1(x) \cdot m_2(x)/\prod_{i=1}^{k} m_i(x)$. In the case where $k = n - 1$ the condition (3) becomes

$$\prod_{i=1}^{n-1} m_i(x)/s(x)/d(x) \cdot s(x). \tag{4}$$

Assume that $\hat{s}(x)$ satisfies (4) and let

$$\hat{d}(x) := \frac{\operatorname{lcm}\{p_1(x),\ p_2(x), \ldots, p_{n-1}(x)\}}{\hat{s}(x)} = \frac{d(x) \cdot s(x)}{\hat{s}(x)},$$

$$\hat{m}_n(x) := \frac{\hat{s}(x)}{\prod_{i=1}^{n-1} m_i(x)},$$

$$\hat{p}_n(x) := \hat{d}(x) \cdot \prod_{i=1}^{n-1} m_i(x).$$

It is easy to verify that $\hat{s}(x)$ is a pseudo secret for the shares $p_1(x), \ldots, p_{n-1}(x)$, $\hat{p}_n(x)$.

Self-Correcting Protocols In the following proposition, two self-correcting protocols are presenting, in the case where error information in cooperations is given.

Proposition 1 (Self-Correcting Protocols)

Protocol 1 Suppose that the participant P_i, $1 \leqslant i \leqslant n$ is pairwise cooperating with all the other participants. Then, from these cooperations, the following pieces of information can be derived:

$$\begin{array}{lll}
P_i,\ P_1: & \gcd\{p_i(x),\ p_1(x)\} \longrightarrow & m_1(x), m_i(x) \\
P_i,\ P_2: & \gcd\{p_i(x),\ p_2(x)\} \longrightarrow & m_2(x), m_i(x) \\
\vdots & \vdots & \vdots \\
P_i,\ P_{i-1}: & \gcd\{p_i(x),\ p_{i-1}(x)\} \longrightarrow & m_{i-1}(x), m_i(x) \\
P_i,\ P_{i+1}: & \gcd\{p_i(x),\ p_{i+1}(x)\} \longrightarrow & m_{i+1}(x), m_i(x) \\
\vdots & \vdots & \vdots \\
P_i,\ P_n: & \gcd\{p_i(x),\ p_n(x)\} \longrightarrow & m_n(x), m_i(x)
\end{array}$$

where $m_i(x), m_j(x),\ j = 1, \ldots, n,\ j \neq i$ are obtained by dividing $p_j(x)$, $p_i(x)$ with $\gcd\{p_i(x), p_j(x)\}$ respectively. Also, $m_i(x)$ in all the previous computations should be the same. Thus, if there is anywhere a different result, then in the corresponding pair will be an error.

Protocol 2 Suppose that every participant is pairwise cooperating with two other participants. Without loss of generality assume that every participant is cooperating in a cycling order first with his/her previous and afterwards with his/her next participant. Let P_i, $1 \leqslant k \leqslant n$ be a participant that gave an error information. By the above assumption, P_i cooperates firstly with the P_{i-1} and next with the P_{i+1}. Obviously, in the case where $i = 1$ then $i - 1$ is considered n and if $i = n$, then $i + 1$ is considered 1. Then, from these cooperations, the following pieces of information can be derived:

$$\begin{array}{lll}
P_i,\ P_{i-1}: & \gcd\{p_i(x),\ p_{i-1}(x)\} \longrightarrow & m_{i-1}(x), m_i(x) \\
P_i,\ P_{i+1}: & \gcd\{p_i(x),\ p_{i+1}(x)\} \longrightarrow & m_{i+1}(x), m_i(x)
\end{array}$$

Thus, from these cooperations, P_{i-1} and P_{i+1} should reveal the same identity $m_i(x)$. In the case where it does not hold, then P_i has given an error information.

3 Computation of The Greatest Common Divisor of Polynomials

For completeness purposes, we shall allow us to briefly discuss a few basic concepts regarding a numerical method for computing the GCD of a set of polynomials. For more details we refer the interested reader to [1, 4, 5, 12, 14].

Definition 1 Let

$$p_i(x) = p_{i,d_{m_i}} x^{d_{m_i}} + p_{i,d_{m_i}-1} x^{d_{m_i}-1} + p_{i,d_{m_i}-2} x^{d_{m_i}-2} + \cdots + p_{i,0}, \qquad i=1, 2, \ldots, n,$$

be n polynomials as defined in (2). Without loss of generality let $p_1(x)$ be the polynomial of maximal degree d_{m_1}. Let $p_2(x)$ be the polynomial with the second maximum degree. Consider the following matrices:

$$S_1 = \begin{bmatrix} p_{1,d_{m_1}} & p_{1,d_{m_1}-1} & p_{1,d_{m_1}-2} & \cdots & p_{1,d_0} & 0 & \ldots & 0 & 0 \\ 0 & p_{1,d_{m_1}} & p_{1,d_{m_1}-1} & \cdots & p_{1,d_1} & p_{1,d_0} & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & p_{1,d_{m_1}} & & \cdots & & p_{1,d_1} & p_{1,d_0} \end{bmatrix},$$

and

$$S_i = \begin{bmatrix} p_{i,d_{m_i}} & p_{i,d_{m_i}-1} & p_{i,d_{m_i}-2} & \cdots & p_{i,d_0} & 0 & \ldots & 0 & 0 \\ 0 & p_{i,d_{m_i}} & p_{i,d_{m_i}-1} & \cdots & p_{i,d_1} & p_{i,d_0} & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & p_{i,d_{m_i}} & & \cdots & & p_{i,d_1} & p_{i,d_0} \end{bmatrix}, \quad i = 2, \ldots, n,$$

where S_1 is an $d_{m_2} \times (d_{m_1} + d_{m_2})$ block matrix representing $p_1(x)$ and S_i is an $d_{m_1} \times (d_{m_1} + d_{m_2})$ matrix which represents $p_i(x)$, $i = 2, \ldots, n$. The classical Sylvester matrix is defined as the following $(d_{m_1} \cdot d_{m_2} + dm_2) \times (d_{m_1} + d_{m_2})$ matrix [1]:

$$S = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_n \end{bmatrix}.$$

By collecting the first row of every block S_i, $i = 2, 3, \ldots, n$ as follows:

$$B = \begin{bmatrix} p_{2,d_{m_2}} & p_{2,d_{m_2-1}} & p_{2,d_{m_2-2}} & \cdots & p_{2,0} \\ p_{3,d_{m_3}} & p_{3,d_{m_3-1}} & p_{3,d_{m_3-2}} & \cdots & p_{3,0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{n,d_{m_n}} & p_{n,d_{m_n-1}} & p_{n,d_{m_n-2}} & \cdots & p_{n,0} \end{bmatrix},$$

and reconstructing the Sylvester matrix S as follows:

$$S^* = \begin{bmatrix} B & \theta & \theta & \Theta & & & \\ \theta & B & \theta & \Theta & & & \\ \theta & \theta & B & \Theta & & & \\ & & & \ddots & \ddots & & \\ & & & \Theta & & \theta & B \\ & & & S_0 & & & \end{bmatrix},$$

where Θ is a zero square matrix, θ is a zero column vector, and I is the identity matrix, we get a matrix with n same blocks called modified Sylvester matrix S^* [12].

Theorem 2 ([1, 12]) *Let S^* be the modified Sylvester matrix of n polynomials $p_1(x), p_2(x), \ldots, p_n(x)$ and $P \cdot S^* = L \cdot U$ the LU factorization with partial pivoting of S^*. Then the nonzero elements of the last nonzero row of U define the coefficients of GCD of the polynomials $p_1(x), p_2(x), \ldots, p_n(x)$.*

Taking advantage of the special form of S^* and zeroing and updating only specific entries of the first block at each step we get the following modified LU factorization algorithm (for more details we refer the interested reader to [12]):

Algorithm The modified LU factorization

STEP 1 : Construct the modified generalized Sylvester matrix S^*

STEP 2 : While number of same blocks > 3 do
if number of same blocks of $(S^*)^{(i)}$ = odd
move the last block after S_1
endif
Compute the upper triangular matrix U
$U = LU(B)$, where B contains
the two first same blocks of $(S^*)^{(i)}$

STEP 3 : Compute the upper triangular matrix U' :
$U' = LU(S^*)^{(k)}$

Computational Complexity The required computational complexity of the previous algorithm is given as follows [12]:

$$O\left((d_{m_1}+d_{m_2})^3\left(2\log_2 2(d_{m_1})-\frac{1}{3}\right)+(d_{m_1}+d_{m_2})^2\left(2n\log_2 2(d_{m_1})+d_{m_2}\right)\right),$$

which is significant reduced in compare with the complexity of the classical LU factorization [4, 5] for a matrix of size of the generalized Sylvester.

Numerical Stability The modified LU factorization computes the exact factorization of a slightly perturbed initial matrix. For the modified LU factorization, it holds that

$$S^* + E = L \cdot U,$$

with

$$\| E \|_\infty \leqslant (d_{m_1}+d_{m_2})^2 \rho u \prod_{i=1}^{\log_2 2(d_{m_1})} \| L_i \|_\infty \| S^* \|_\infty,$$

where ρ is the growth factor and u the unit round off [12]. More details for the stability of the LU factorization with partial pivoting can be found in [14].

4 Hybrid Implementation

In this section we present an example evaluating the proposed method in a hybrid way. The computation of the GCD of polynomials is achieved using floating point arithmetic and the modified LU factorization algorithm and the divisions of polynomials symbolically.

Table 1 Polynomials of participants given by the dealer

Participant	Polynomial
P_1	$x^6 - 42x^5 + 685x^4 - 5460x^3 + 22084x^2 - 43008x + 31680$
P_2	$x^6 - 41x^5 + 645x^4 - 4855x^3 + 17834x^2 - 29424x + 15840$
P_3	$x^6 - 40x^5 + 607x^4 - 4324x^3 + 14572x^2 - 21376x + 10560$
P_4	$x^6 - 39x^5 + 571x^4 - 3861x^3 + 12100x^2 - 16692x + 7920$

Let us suppose that we have one dealer and four participants. Let $s(x) = x^4 - 10x^3 + 35x^2 - 50x + 24$ be the secret and $d(x) = x^3 - 33x^2 + 362x - 1320$ the polynomial chosen by the dealer for increasing the difficulty of breaking the secret key. The dealer gives to participants the polynomials shown in Table 1.

Let us assume that the participants P_1 and P_2 inform each other about their polynomials and compute their GCD. The information that they found is

$$\begin{aligned} g_{12}(x) &:= \gcd\{p_1(x),\ p_2(x)\} \\ &= x^5 - 40x^4 + 605x^3 - 4250x^2 + 13584x - 15840. \end{aligned}$$

The participant P_1 divides his/her polynomial $p_1(x)$ with $g_{12}(x)$ and finds the factor $m_2(x)$ of participant P_2:

$$\begin{aligned} \frac{p_1(x)}{g_{12}(x)} &= \frac{x^6 - 42x^5 + 685x^4 - 5460x^3 + 22084x^2 - 43008x + 31680}{x^5 - 40x^4 + 605x^3 - 4250x^2 + 13584x - 15840} \\ &= x - 2 \\ &= m_2(x). \end{aligned}$$

Similarly, participant P_2 divides his/her polynomial $p_2(x)$ with the GCD $g_{12}(x)$ and finds the factor $m_1(x)$ of participant P_1:

$$\begin{aligned} \frac{p_2(x)}{g_{12}(x)} &= \frac{x^6 - 41x^5 + 645x^4 - 4855x^3 + 17834x^2 - 29424x + 15840}{x^5 - 40x^4 + 605x^3 - 4250x^2 + 13584x - 15840} \\ &= x - 1 \\ &= m_1(x). \end{aligned}$$

Thus, participants P_1 and P_2 have found a part of the secret $s(x)$ but they have to cooperate with the other two participants as well in order to compute the secret $s(x)$, since they have to find out $m_3(x)$ and $m_4(x)$ in order to deconvolute $d(x)$ from their LCM of their polynomials. If all participants reveal their polynomials $p_i(x),\ i = 1, 2, 3, 4$ and compute the GCD $g(x)$, then:

$$g(x) := \gcd\{p_1(x),\ p_2(x),\ p_3(x),\ p_4(x)\}$$
$$= x^3 - 33x^2 + 362x - 1320.$$

Remark 4 The previous GCD was computed using the modified LU factorization algorithm with inner tolerance 10^{-16}. Since in numerical arithmetic different tolerances may lead to different results, the dealer may include the tolerance as information to the participants along with their polynomials.

Each participant divides his/her polynomial $p_i(x), i = 1, 2, 3, 4$ with the computed GCD $g(x)$ and obtain the results presented in Table 2.

Participant, e.g., P_2 can now cooperate with P_1 as shown before in order to find out $m_1(x)$ and inform P_1 about it. Thus P_1 can now multiply $g_1(x)$ with $m_1(x)$ in order to retrieve the secret $s(x)$:

$$g_1(x) \cdot m_1(x) = \left(x^3 - 9x^2 + 26x - 24\right) \cdot \left(x - 1\right)$$
$$= x^4 - 10x^3 + 35x^2 - 50x + 24$$
$$= s(x).$$

The same result is also obtained if all participants cooperate in cyclic pairs as shown in Table 3. The secret is obtained by the product $\prod_{i=1}^{4} m_i(x)$:

$$s(x) = \prod_{i=1}^{4} m_i(x) = x^4 - 10x^3 + 35x^2 - 50x + 24.$$

Table 2 Polynomials of participants after division and the corresponding results

Participant	Polynomial division
P_1	$\frac{p_1(x)}{g(x)} = \frac{x^6 - 42x^5 + 685x^4 - 5460x^3 + 22084x^2 - 43008x + 31680}{x^3 - 33x^2 + 362x - 1320}$
P_2	$\frac{p_2(x)}{g(x)} = \frac{x^6 - 41x^5 + 645x^4 - 4855x^3 + 17834x^2 - 29424x + 15840}{x^3 - 33x^2 + 362x - 1320}$
P_3	$\frac{p_3(x)}{g(x)} = \frac{x^6 - 40x^5 + 607x^4 - 4324x^3 + 14572x^2 - 21376x + 10560}{x^3 - 33x^2 + 362x - 1320}$
P_4	$\frac{p_4(x)}{g(x)} = \frac{x^6 - 39x^5 + 571x^4 - 3861x^3 + 12100x^2 - 16692x + 7920}{x^3 - 33x^2 + 362x - 1320}$
Participant	Result $g_i(x)$
P_1	$g_1(x) = x^3 - 9x^2 + 26x - 24 = (x-2)(x-3)(x-4)$
P_2	$g_2(x) = x^3 - 8x^2 + 19x - 12 = (x-1)(x-3)(x-4)$
P_3	$g_3(x) = x^3 - 7x^2 + 14x - 8 = (x-1)(x-2)(x-4)$
P_4	$g_4(x) = x^3 - 6x^2 + 11x - 6 = (x-1)(x-2)(x-3)$

Table 3 Cooperation participant P_2 in pairs

Participants	GCD $g_i(x)$
P_1, P_2	$g_{12}(x) = \gcd\{p_1(x), p_2(x)\} = x^5 - 40x^4 + 605x^3 - 4250x^2 + 13584x - 15840$
P_2, P_3	$g_{23}(x) = \gcd\{p_2(x), p_3(x)\} = x^5 - 38x^4 + 531x^3 - 3262x^2 + 8048x - 5280$
P_3, P_4	$g_{34}(x) = \gcd\{p_3(x), p_4(x)\} = x^5 - 36x^4 + 463x^3 - 2472x^2 + 4684x - 2640$
P_4, P_1	$g_{41}(x) = \gcd\{p_4(x), p_1(x)\} = x^5 - 38x^4 + 533x^3 - 3328x^2 + 8772x - 7920$
Participants	Polynomial deconvolution $m_i(x)$
P_1, P_2	$m_2(x) = \dfrac{p_1(x)}{g_{12}(x)} = x - 2$
P_2, P_3	$m_3(x) = \dfrac{p_2(x)}{g_{23}(x)} = x - 3$
P_3, P_4	$m_4(x) = \dfrac{p_3(x)}{g_{34}(x)} = x - 4$
P_4, P_1	$m_1(x) = \dfrac{p_4(x)}{g_{41}(x)} = x - 1$

Table 4 Cooperation of participant P_2 and P_4 with P_1 in pairs

Participants	GCD $g_i(x)$
P_1, P_2	$g_1(x) = \gcd\{p_1(x), p_2(x)\} = x^5 - 40x^4 + 605x^3 - 4250x^2 + 13584x - 15840$
P_2, P_3	$g_3(x) = \gcd\{p_2(x), p_3(x)\} = x^4 - 34x^3 + 395x^2 - 1682x + 1320$
P_2, P_4	$g_4(x) = \gcd\{p_3(x), p_4(x)\} = x^5 - 37x^4 + 497x^3 - 2867x^2 + 6366x - 3960$
Participants	Polynomial deconvolution $m_i(x)$
P_1, P_2	$m_2(x) = \dfrac{p_1(x)}{g_1(x)} = x - 2$
	$m_1(x) = \dfrac{p_2(x)}{g_1(x)} = x - 1$
P_2, P_3	$\hat{m}_3(x) = \dfrac{p_2(x)}{g_3(x)} = x^2 - 7x + 12 = (x-3)\cdot(x-4)$
	$\tilde{m}_3(x) = \dfrac{p_3(x)}{g_3(x)} = x^2 - 17x + 30 = (x-2)\cdot(x-15)$
P_2, P_4	$m_4(x) = \dfrac{p_2(x)}{g_4(x)} = x - 4$
	$m_2(x) = \dfrac{p_4(x)}{g_4(x)} = x - 2$

Let us suppose that one of the participants, e.g., P_3 gives wrong information and let $\hat{p}_3(x) = x^6 - 51x^5 + 1003x^4 - 9417x^3 + 41764x^2 - 72900x + 39600$ be the false instead of the real one $p_3(x) = x^6 - 40x^5 + 607x^4 - 4324x^3 + 14572x^2 - 21376x + 10560$. $\hat{p}_3(x)$ will be either coprime with the polynomials of the other participants or may have some common roots but not the right ones.

Protocol 1

Let participant P_2 cooperate per pair with all other ones. The results are summarized in Table 4. Every P_i, $i = 1, 3, 4$ through the cooperation with P_2 should give as result $m_2(x)$ to P_2. As it is shown in Table 4 only participant P_3 did not give the factor $m_2(x)$ as it was supposed to do. Thus P_3 had given wrong information.

Table 5 Cyclic cooperation in pairs

Participants	GCD $g_i(x)$
P_2, P_3	$g_{23}(x) = \gcd\{p_1(x), \hat{p}_3(x)\} = x^4 - 34x^3 + 395x^2 - 1682x + 1320$
P_3, P_4	$g_{34}(x) = \gcd\{\hat{p}_3(x), p_4(x)\} = x^5 - 36x^4 + 463x^3 - 2472x^2 + 4684x - 2640$
Participants	Polynomial deconvolution $m_i(x)$
P_2, P_3	$\hat{m}_2(x) = \frac{p_2(x)}{g_{23}(x)} = x^2 - 7x + 12 = (x-3)\cdot(x-4)$
P_3, P_4	$\hat{m}_4(x) = \frac{p_4(x)}{g_{34}(x)} = x - 3$

Protocol 2
Participant P_3 cooperates per pair with the previous participant P_2 and the next one P_4 as shown in Table 5. From the cooperation with P_3 the participants P_2 and P_4 should take as result the same polynomial $m_3(x) = \frac{p_2(x)}{g_{23}(x)} = x - 3$ and $m_3(x) = \frac{p_4(x)}{g_{34}(x)} = x - 3$ but they have taken different ones. Also $\hat{m}_4(x)$ should be a polynomial of larger degree since it should include $m_3(x)$ and $m_4(x)$ as a product. Also $\hat{m}_2(x)$ should include $m_2(x)$ and $m_3(x)$ as a product. The third participant gave wrong results with both the previous and next participant, thus he/she gave wrong information.

5 Synopsis and Concluding Remarks

In this paper a ramp secret sharing scheme is presented. The scheme is based on the computation of the greatest common divisor of polynomials. A subset of the participants can derive information about the secret and can approximate the roots of their shares, but the approximation of all real roots of a polynomial is, in general, a hard task. The dealer can make more difficult the approximation of the roots of the polynomials by selecting roots with many decimal digits. Two correcting protocols are also presented in order to recognize the participant that gave false information. The computation of the greatest common divisor of polynomials is implemented through stable numerical linear algebra methods in an efficient way. Polynomial divisions can be evaluated either numerically through Horner's algorithm which is proved that is optimal [7, 8] in respect of floating point operations or symbolically.

The proposed scheme will be improved in a future work by constructing, by the dealer a vector with different inner tolerances for the participants. In that way, any participant will compute all the roots of its polynomial but not all of them will be the same with that of the secret ones. Furthermore, we will use polynomials over fields of non-zero characteristic in order to significantly improve the robustness of the proposed method.

References

1. S. Barnett, Greatest common divisor of several polynomials. Linear Multilinear A **8**, 271–279 (1980)
2. G.R. Blakley, Safeguarding cryptographic keys, in *Proceedings AFIPS 1979 National Computer Conference* (1979), pp. 313–317
3. G.R. Blakley, C. Meadows, Security of ramp schemes, in *Advances in Cryptology*, ed. by G.R. Blakley, D. Chaum. CRYPTO 1984. Lecture Notes in Computer Science, (Springer, Berlin, Heidelberg, 1984), **196**, 242–268
4. R.L. Burden, J.D. Faires, *Numerical Analysis*, 6th edn. (Brooks/Cole Publishing Company, Pacific Grove, 1997)
5. B.N. Datta, *Numerical Linear Algebra and Applications*, 2nd edn. (SIAM, Philadelphia, 2010)
6. A.K. Lenstra, H.W. Lenstra, Jr., L. Lovász, Factoring polynomials with rational coefficients. Math. Ann. **261**, 515–534 (1982)
7. A.M. Ostrowski, *On Two Problems in Abstract Algebra Connected with Horner's Rule*. Studies in Mathematics and Mechanics (Academic, New York, 1954), pp. 40–48
8. V.Y. Pan, On means of calculating values of polynomials. Russ. Math. Surv. **21**, 105–136 (1966)
9. V. Pan, Univariate polynomials: nearly optimal algorithms for numerical factorization and root-finding. J. Symb. Comput. **33**, 701–733 (2002)
10. M. Sagraloff, K. Mehlhorn, Computing real roots of real polynomials. J. Symb. Comput. **73**, 46–86 (2016)
11. D.R. Stinson, Ideal ramp schemes and related combinatorial objects. Discrete Math. **341**, 299–307 (2018)
12. D. Triantafyllou, M. Mitrouli, On rank and null space computation of the generalized Sylvester matrix. Numer. Algorithms **54**, 297–324 (2010)
13. B.L. van der Waerden, *Algebra*, vol. I (Springer, New York, 1991)
14. J.H. Wilkinson, *The Algebraic Eigenvalue Problem* (Clarendon Press, Oxford, 1965)