



---

**TECHNICAL UNIVERSITY**

**OF CLUJ-NAPOCA**

**AUTOMATION AND COMPUTER SCIENCE FACULTY**

**ing. Otto POSZET**

# **PHD THESIS**

**CONTRIBUTIONS TO IMPROVING THE PERFORMANCE OF  
DISTRIBUTED SYSTEMS IN SENSE OF ERROR CORRECTION  
AND DATA SECURITY**

**ADVISOR**

**Prof. dr. ing. Iosif IGNAT**

---

**2007**

---

# Table of Contents

<b><i>Table of Contents</i></b>	<b>4</b>
<b>1. Introduction</b>	<b>9</b>
1.1. Structure of the thesis	9
1.2. Importance of the domain	11
1.3. Classification of the codes	13
1.4. Information security and cryptography	14
1.5. Research directions	16
1.6. Conclusions	20
<b>2. Ensuring data correctness and data security using error correcting codes</b>	<b>21</b>
2.1. Error correcting and error detecting codes	21
2.1.1. Decodification of linear codes	23
2.2. Cyclic bloc codes	25
2.2.1. Polynomial representation of cyclic codes	25
2.2.2. Matrix representation of cyclic codes	26
2.2.3. Cyclic codes analysis using Discrete Fourier Transform	27
2.2.4. Systematic encoding	36
2.2.5. Syndromes computation	38
2.3. Reed-Solomon codes	39
2.3.1. Decoding Reed-Solomon codes	40
2.4. Error correcting codes based cryptosystems	48
2.4.1. Error correcting codes based public key encryption-decryption methods	49
2.5. Conclusions and own contributions	56
<b>3. Distributed computing systems for electronic payment</b>	<b>59</b>
3.1. General model of an Electronic Payment System	59
3.1.1. Classification of Electronic Payment Systems	61
3.2. Specifications for an off-line Electronic Payment System	63
3.3. Specifications for an on-line Electronic Payment System	66
3.5. Cryptographic techniques used in Electronic Payment Systems	70
3.5.1. Digital signatures	70
3.5.2. Mutual identifications – interactiv verification systems	73
3.5.3. The McEliece signature	81
3.5.4. One spendable electronic coins	82
3.6. Conclusions and own contributions	83
<b>4. Solutions proposed for increasing the performance of electronic payment systems</b>	<b>86</b>
4.1. Obtaining one-spendable electronic coins	86
4.1.1. Untraceability and security	86
4.1.2. Protocols for obtaining electronic money	88
4.2. The payment protocol	107
4.3. The deposit transaction	108
4.4. Conclusions and own contributions	112
<b>5. Arhitecture and implementation of a new, secure and fault-tolerant, off-line electronic payment system</b>	<b>114</b>

<b>The proposed off-line system</b>	<b>115</b>
5.1.1. Mutual identifications	116
5.1.2. The withdrawal protocol	116
5.1.3. The payment protocol	117
5.1.4. Deposit transaction and clearing	118
<b>5.2. Multiple precision arithmetic</b>	<b>119</b>
<b>5.3. Public key cryptosystem based on McEliece signature</b>	<b>119</b>
5.3.1. Information encoding and decoding using Reed – Solomon codes	119
5.3.2. The McEliece encryption and decryption algorithm	121
5.3.3. Implementation of the McEliece cryptosystem	122
<b>5.4. Client, payer</b>	<b>123</b>
5.4.1. Interaction Payer – Bank	123
5.4.2. Interaction Payer – Electronic Shop	124
<b>5.5. Shop, seller</b>	<b>125</b>
5.5.1. Virtual shop – web pages	126
5.5.2. Server magazin	129
<b>5.6. Bank</b>	<b>131</b>
<b>5.7. Conclusions and own contributions</b>	<b>134</b>
<b>6. Experimental results for the off-line system</b>	<b>8</b>
<b>6.1. McEliece based encryption – decryption</b>	<b>136</b>
6.1.1. Conclusions for the McEliece cryptosystem	139
<b>6.2. Identification protocols</b>	<b>140</b>
6.2.1. Rezults	140
<b>6.3. Withdrawal of electronic coins</b>	<b>141</b>
6.3.1. Coin generation with a smart card	141
6.3.2. Obtaining coins with restrictive blinding	142
6.3.3. Cut and choose for single-term coins    Cut and choose for multi-term coins	143
6.3.4. Conclusions	143
<b>6.4. Payment transaction</b>	<b>144</b>
6.4.1. Concluzions	145
<b>6.5. Concluzions and own contributions</b>	<b>145</b>
<b>7. A new, fault tolerant, on-line electronic payment system</b>	<b>148</b>
<b>7.1. Presentation of the proposed, fault tolerant, on-line electronic payment system</b>	<b>149</b>
<b>7.2. Security aspects</b>	<b>150</b>
<b>7.3. Fault tolerant aspects</b>	<b>151</b>
<b>7.4. Test results</b>	<b>152</b>
<b>7.5. Concluzions and own contributions</b>	<b>156</b>
<b>8. Final conclusions</b>	<b>158</b>
<b>References</b>	<b>163</b>
<b>Appendix</b>	<b>178</b>
<b>A.1. Notations, abbreviations</b>	<b>178</b>
<b>A.2. Algorithms for multiple precision arithmetic</b>	<b>181</b>
<b>A.3. Classes used in McEliece cryptosystem</b>	<b>186</b>
<b>A.4. The structure of the tables used by the bank</b>	<b>191</b>

## 1. Introduction

**Coding theory** is a very important, relative new chapter in applied mathematics, and has significant theoretical and technological results, for example **data transmission, compression, information security** and **cryptography**. This thesis proposes as main goal obtaining new methods for increasing security and correctness of distributed systems. To achieve the proposed goal the thesis analyses four main aspects: error correcting codes, data security, electronic payment systems and fault tolerance in on-line and off-line electronic payment systems.

In the last years the reasesarches in this domain were focused on the following main directions: projecting and analyzing new cryptographical methods and ciphers, new hash functions, authentication algorithms, public key cryptosystems based on elliptic curves, mobile communication security, smart cards, quantic cryptosystems, cryptosystems based on error correcting codes, electronic payment systems.

The main contributions in the first chapter were: presentation of the importance of the domain, classification of the existing codes, general model of the cryptography and its main goals, possible attacks in cryptosystems, and indication of the main research directions in coding theory.

## 2. Ensuring data correctness and data security using error correcting codes

In this chapter I analyse the mathematic background used in coding theory, and the possibility of error correction in data transmission or data storage. After the definition of linear codes, Hamming distance, Hamming weight, information encoding, I defined three main decoding strategies: *Complete Hard-Decision Decoding*, *t-Bounded Distance Decoding* and *Bounded Hard-Decision Decoding*.

The most important and the most efficient codes are the cyclic codes. This class of codes allows an elegant mathematical representation, are very performant and from this reason very often used in practice. After the mathematical background (polynomial and matrix representation), I used the Discrete Fourier Transform (DFT) and the linear complexity of sequences in analysis of this kind of error correcting codes.

Based on Blahut's theorem, we can compute the Hamming weight of a codeword ( $b^n$ ) as the linear complexity of the sequence  $(B^n)^\infty$  having the first period ( $B^n$ ):  $w_H(b^n) = \Lambda((B^n)^\infty)$ . So, the minimal distance  $d_{\min}$  of the code is equal to minimal linear complexity of all the sequences  $(B^n)^\infty$ , who have the first period ( $B^n$ ) a non-zero vector vector of length  $n$ , containing a defined zero-pattern.

**Definition 17.** The linear complexity  $\Lambda(a^n)$  or  $\Lambda(a^\infty)$  of a finite or semi-infinite sequence ( $a^n$ ) or ( $a^\infty$ ) is the smallest positive integer  $L$  for which there are coefficients  $c_1, c_2, \dots, c_L$  in  $F$  so that the following linear recursivity exists:

$$a_i + c_1 a_{i-1} + c_2 a_{i-2} + \dots + c_L a_{i-L} = 0 \quad \text{for every } L \leq i < n \text{ or } L \leq i < \infty. \quad (33)$$

**Lema 2. Rank lema for semi-infinite periodically sequences.** The linear complexity of a periodically, semi-infinite sequence  $(a^n)^\infty$  is equal with the rank of the circulant matrix  $M(a^n)$ :

$$L = \Lambda((a^n)^\infty) = \text{rank} [M(a^n)] \quad (36)$$

This lema can be seen as a special case of the lema for finite sequence, but the symbols  $\Delta$  are substituted with the first symbol from the next period. We obtained the following equations:

$$\text{rank}[M(A^n)] = w_H[(a^n)], \quad \text{rank}[M(a^n)] = w_H[(A^n)]; \quad (37)$$

$$\Lambda((A^n)^\infty) = \text{rank} [M(A^n)], \quad \Lambda((a^n)^\infty) = \text{rank} [M(a^n)]; \quad (38)$$

$$\Lambda((A^n)^\infty) = w_H[(a^n)], \quad \text{and} \quad \Lambda((a^n)^\infty) = w_H[(A^n)]. \quad (39)$$

So different cyclic codes and their minimal Hamming distance can be evaluated with the aid of two methods: rank computing or using recursivity, computing the minimal linear complexity of sequences containing a given zero-pattern (the Fourier transform of the codewords).

Schaub proposed an algorithm to compute the rank of the matrix  $M(A^n)$ , and he searches linear independent rows. In this goal he forms a set  $S$  of linear independent rows, having the initial value  $S = \{r_0\}$ , first row from  $M(A^n)$ , and he inserts in this set each new, linear independent row found in  $M(A^n)$ .

I proposed a new, efficient rank computing algorithm, based on “bordering”. Each time, if the algorithm finds a sub-matrix of dimension  $k$ , having the determinant different from zero, it constructs all the  $k+1$  dimensionally matrixes, using the rows and the columns of the initial,  $n$  dimension matrix (“bordering”). If there are at least one matrix of dimension  $k+1$ , having the determinant different from zero, then the algorithm repeats this “bordering” procedure for  $k+2$ , until we reach  $n$ , or all the matrixes have determinant zero. If all the  $k+1$  dimension matrixes have determinant 0, that means, that the rank of the initial matrix is  $k$ . This algorithm is usefull in developing and analysis of cyclic codes used in construction of error correcting codes based cryprosystem.

In the following I indicated the possibility of systematically encoding using the polynomial representation of the cyclic codes: encoding with the aid of generator polynomial or the parity check polynomial. This kind of codifications can be implemented easily with shift registers.

#### Systematically encoding using generator polynomial:

If  $g(x) = g_0 + g_1x + \dots + g_{n-k-1}x^{n-k-1} + x^{n-k}$  is the generator polynomial for the cyclic code  $[n, k]_q$ , then for the information word  $u = (u_0, \dots, u_{k-1}) \leftrightarrow u(x)$  the corresponding parity check word will be  $p = (p_0, \dots, p_{n-k-1}) \leftrightarrow p(x) = R_{g(x)}[-x^{n-k}u(x)]$ , (40) and the corresponding code word will be:

$$a = (p_0, \dots, p_{n-k-1}, u_0, \dots, u_{k-1}) \leftrightarrow a(x) = p(x) + x^{n-k}u(x). \quad (41)$$

#### Systematically encoding using parity check polynomial:

Let  $h(x) = h_0 + h_1x + \dots + h_{k-1}x^{k-1} + x^k$  be the parity check polynomial for the cyclic code  $[n, k]_q$ , and the information word  $a = (a_0, \dots, a_{n-k-1}, a_{n-k}, \dots, a_{n-1})$ , where position  $(0, \dots, n-k-1)$  are used for parity checking, and positions  $(n-k, \dots, n-1)$  are informational positions. For the parity check

positions we have:  $a_m = -\sum_{i=0}^{k-1} h_i a_{m-i+k} = f(a_{m+1}, \dots, a_{m+k})$ . (43)

The decodification has 3 phases, and uses the syndrome of the received word:

1. Syndrome computing :  $s(x) = R_{g(x)}[y(x)]$ .
2. Finding the error vector  $e(x)$  (error pattern) corresponding the syndrome  $s(x)$ .
3. Error correction:  $a(x) = y(x) - s(x)$ .

**Definition 19.** For a given  $p$  and  $m$  and for a given minimal distance  $d = d_{\min}$ , a Reed-Solomon code can be defined as:  $(n, k, d_{\min})_q = (p^m - 1, p^m - d, d)_q$ . (45)

If  $(a_0, \dots, a_{n-1}) = a(x)$  are the codewords in time domain, then the representations of these codewords in frequency domain are  $(A_0, \dots, A_{n-1}) = A(x)$ , and these codewords contain on  $n - k = d - 1$  successive positions 0. These positions are the parity check positions in frequency domain. So, the code has following form:

$$C = \{ a(x) \mid A(x) = R_{x^{-1}}[x^{1+d-1}B(x)] \text{ pentru } \text{grad } B(x) \leq n - d \}. \quad (46)$$

For decodification, if the syndrome is different from 0, we must locate the eronated positions and these errors must be eliminated. For this, we must resolve the key equation system, with  $\tau$  equations and  $\tau$  unknown terms.

There is a special class of cryptosystem, who use error correcting codes in cryptographically goals. In this kind of systems security is achieved by insertion of random correctable error patterns in encoded information (codewords). So, a cryptosystem with locally randomness is obtained. Example for cryptosystems based on error correcting codes: McEliece, Niederreiter, Stern, Li-Wang, Rao-Nam.

The McEliece cryptosystem is used as starting point in this thesis, in construction of electronic payment system. In the McEliece public key cryptosystem the private key contains random generated matrices, and the public key is the set  $Z$  of correctable error patterns ( $Z = \{ \underline{z} \in \text{GF}(q)^n \mid w_H(\underline{z}) = \lfloor \frac{d-1}{2} \rfloor \}$ ), and the encryption matrix.

**Definition 22.** If  $C$  is an  $[n, k, d]$  linear code,  $G$  the generator matrix,  $S$  a random generated non-singular matrix and  $P$  a permutation matrix, then the **McEliece** encryption matrix can be defined as:  $E = S G P$ . (93)

**Encryption.** The  $\underline{x}$  information words will be encoded as follows:  $\underline{y} = \underline{x}E + \underline{z}$ , where  $\underline{z}$  is a randomly chosen error pattern from  $Z$ .

**Decryption.**  $\underline{y}P^T = \underline{x}SG + \underline{z}P^T$ , where  $\underline{z}P^T \in Z$ . Then  $\underline{x}S$  can be computed with the secret decryption of  $C$ , known the generator matrix. The plaintext will be  $(\underline{x}S)S^{-1}$ . (95)

**Key.** Public key:  $Z$  and  $E$ . Private key:  $S, P$ , and  $G$ .

In this chapter I analysed some performant error correcting codes from the point of view of efficiency, mathematical representation and code distance. We saw an analysis based on Discrete Fourier Transform and linear complexity of sequences. I proposed a performant rank computing algorithm, which can be used in obtaining the minimal code distance for a cyclic code. We saw systematically encoding methods (using the generator or the parity check polynomial) and decoding using syndrome calculations. An important class of cyclic codes are the Reed-Solomon codes. After mathematical background we saw the encoding and decoding in time or frequency domain. At the end of this chapter we have studied cryptosystems based on error correcting codes, and the most important was the McEliece cryptosystem. We saw the advantages and disadvantages of these systems. The McEliece scheme will be used later in construction of an electronic payment system because it behaves well even in the presence of intensive noises.

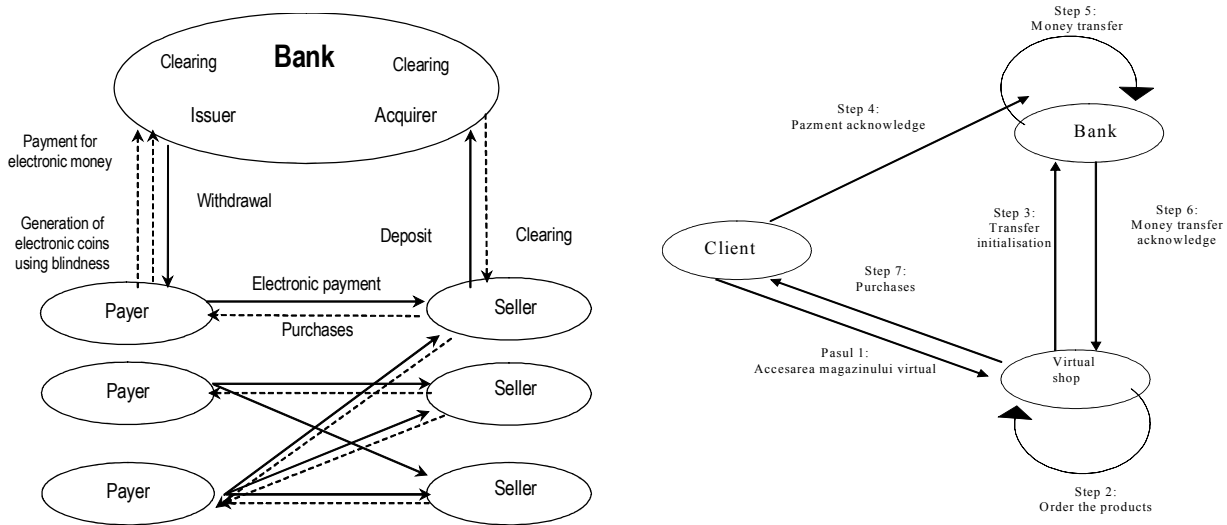
Main own contributions in this chapter:

1. Code analysis in time and frequency domain using DFT.
2. A new, rank computing algorithm used in the computing of minimal code distance.
3. Step by step formulation of analysis and synthesis of Reed-Solomon codes used later in construction of the proposed electronic payment system.
4. Replacement of Goppa codes in the McEliece system with Reed-Solomon codes to obtain a newer, most performant and systematically designed cryptosystem.

### 3. Distributed computing systems for electronic payment

A very important application in distributed systems is e-commerce and e-business. The most difficult problems in this kind of systems are security, confidentiality and efficiency.

**Definition 29.** An **Electronic Payment System (EPS)** is built by a set of parties and a set of interactions between them, having the main goal value interchange using electronic money as payment instrument. The participating entities (roles) are: payers (clients), sellers (electronic shops), and money emission and acquisition entities (banks). The main interactions are: withdrawal of electronic money, payment, deposit, clearing.



**General model for an off-line and for an on-line electronic payment system**

To obtain a more secure and more reliable system, I proposed the combining of traditional protocols with error correcting codes. In this idea, I used the modified McEliece signature to encrypt each message between the parties. The immunity against noises is ensured by the Reed-Solomon codes used in McEliece scheme. If the intentionally inserted error patterns have a Hamming weight  $\omega$  less than the correction capacity  $\lfloor \frac{d-1}{2} \rfloor$  of the code, then the system can correct yet accidentally inserted error having weight less than or equal to  $\lfloor \frac{d-1}{2} \rfloor - \omega$ .

In this chapter after the definition and general model of EPS, describing main parties and interactions between them, I analysed the main advantages and disadvantages of this kind of systems. The main disadvantage of the traditional EPS is that they do not have fault tolerance at all. To prevent this, I proposed an enhancement of each protocol, combining traditional EPS with error correcting codes. That's way I defined a new encryption scheme, named "*modified McEliece signature*", based on Reed-Solomon codes. The first example of use is a ***new identification protocol***, proposed by me, and it is based on modified McEliece signature.

Own contributions in this chapter:

1. Step by step specification for an off-line and on-line Electronic Payment System (EPS).
2. Definition of a new kind of encryption-decryption scheme, named "*modified McEliece signature*", and based on Reed-Solomon codes.
3. Modification and enhancement of the known identification protocols with the aid of the new McEliece signature, obtaining fault tolerant, error correcting protocols.
4. I proposed a new mutual identification scheme based on the three step identification protocol and on the McEliece signature.

## 4. Solutions proposed for increasing the performance of electronic payment systems

In off-line EPS there are two contradictory interests: **confidentiality** (untraceability) vs. **security**. Both of them must be satisfied, the EPS to be accepted from users and organizations. This is the reason, that electronic coins are generated by clients in a format unknown by the bank, but they are signed by the bank in order to get a given value. The value of the coin depends on the secret

key used by the bank for signature. To ensure degradation integrity, electronic coins must contain encoded the unique identifier of the client, who generates and use the coin. The most efficient form of the electronic coin is the three-part form:

$$\text{Coin} = \{S: \text{secret\_key}, P: \text{public\_key}, \sigma_B(P): \text{certified\_public\_key}\} \quad (157)$$

So, the withdrawal protocol must contain at least two elements: blinding (untraceability) and verifying the inclusion of client's ID in the coin. In thesis the following protocols are studied and enhanced:

- Withdrawal in the presence of a Trusted Third Party
- Withdrawal with the aid of a smart card
- Cut and choose for single term coin
- Cut and choose for multi term coin
- Withdrawal using restrictive blindness

Each protocol is enhanced by error correcting codes, using the modified McEliece signature, so, all the protocols are a fault tolerant behaviour. To evaluate the security of these protocols, I made a probability computation. The probability, that an intruder can break the cut and choose protocol for multi-term coins, is:

$$P(L, k, L/2) = \frac{k}{L} \cdot \frac{k-1}{L-1} \dots \frac{k - \left(\frac{L}{2} - 1\right)}{L - \left(\frac{L}{2} - 1\right)} = \frac{C_k^{L/2}}{C_L^{L/2}} = \frac{k \cdot (k-1) \dots (k - L/2 + 1)}{L \cdot (L-1) \dots (L/2 + 1)} \quad (197)$$

In case of a successful attack, the bank can afterwards compute the secret ID of the client, known that S contains on the last positions this ID, using the following equation,:

$$S = (c_2 - c_1)^{-1} (r_1 - r_2) \bmod q. \quad (317)$$

$$\text{and } \omega = (r_1 + c_1 S) \bmod q. \quad (318)$$

In the last part of this chapter I described the proposed, fault tolerant version of the payment and deposit protocol for electronic coins represented in three part form.

Own contributions in this chapter are:

1. Enhancements for increasing **security** and for obtaining **fault tolerance** for *withdrawal* protocols, *payment* and *deposit* protocol, using modified McEliece signature. The modified and proposed protocols are:
  - Fault tolerant withdrawal protocol in presence of a TTP
  - Fault tolerant withdrawal protocol using a smart card
  - Fault tolerant cut and choose protocol for single term coins
  - Fault tolerant cut and choose protocol for multi term coins
  - Fault tolerant withdrawal using restrictive blindness
  - Fault tolerant divided key withdrawal protocol
  - Fault tolerant payment
  - Deposit protocol
2. Computation of the breaking probability for the cut and choose protocol, using single term, or multi term coins.
3. Analysis of the necessary verifications in the deposit protocol for single term or multi term coins, respectively for divided keys, and computation of client's ID for dishonest users (in case of two or more spending the coin).

## 5. Architecture and implementation of a new, secure and fault-tolerant, *off-line* electronic payment system

In this chapter I describe the implementation of a fault tolerant off-line electronic payment system, based on one-spendable electronic coins. The goal of this system was the testing of the protocols proposed in previous chapters from the point of view of performance and fault tolerance. Combining classical EPS with error correcting codes will create a new, fault tolerant EPS with increased security.

The parties present in system are: client (payer), seller (electronic shop), electronic bank. Each protocol was implemented in two versions: the original form (without fault tolerance) and the proposed, fault tolerant version. Electronic coins are represented in three part form: (*privat\_key*, *public\_key*, *certified\_public\_key*). All the coins are one-spendable, transactions are untraceable, but the system ensure the degradation integrity, so, coins spent more than once, allows computing the secret ID of the unhoneat user.

The first step in implementation was the creation of a mathematical library for very large numbers, so that all the functions and arithmetical operations took place in the arithmetic of finit fields. The next step was the implementation of the proposed cryptosystem: modified McEliece cryptosystem, based on Reed-Solomon codes. The last step was the implementation of main parties: clients (having electronic wallets with observers), virtual shops (web pages, shopping server) and electronic banks (bank server, clearing, verifications, data bases). Between each entities there are different interactions (the protocols presented in previous chapters), and these protocols were implemented in basic form and in enhanced, proposed, fault tolerant form.

Own contributions:

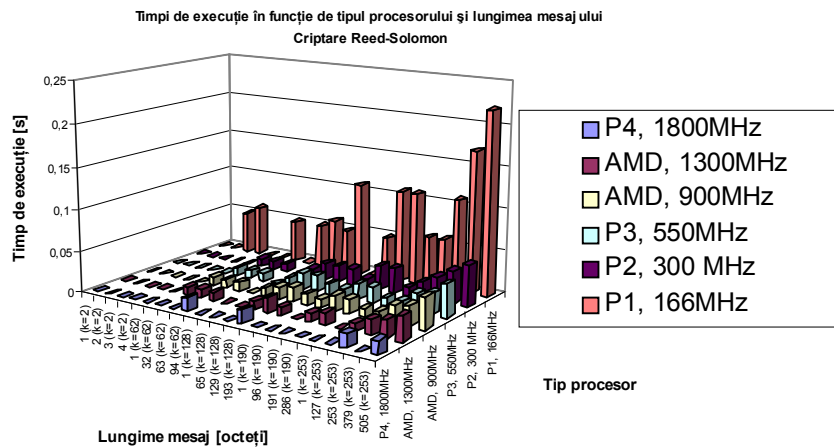
1. Proposal and implementation of a new, ***fault tolerant, off-line EPS***, based on error correcting code, and having payment instruments one-spendable electronic coins.
2. Choose and implementation of protocols used between parties in basic form, and in proposed, fault tolerant form: ***mutual identifications, withdrawal in different versions, payment protocol, deposit and clearing***.
3. Creation of a ***mathematical function library*** for very larg numbers, in the arithmetic of finite fields.
4. Implementation of the ***modified McEliece cryptosystem***, based on Reed-Solomon codes.

## 6. Experimental results for the off-line system

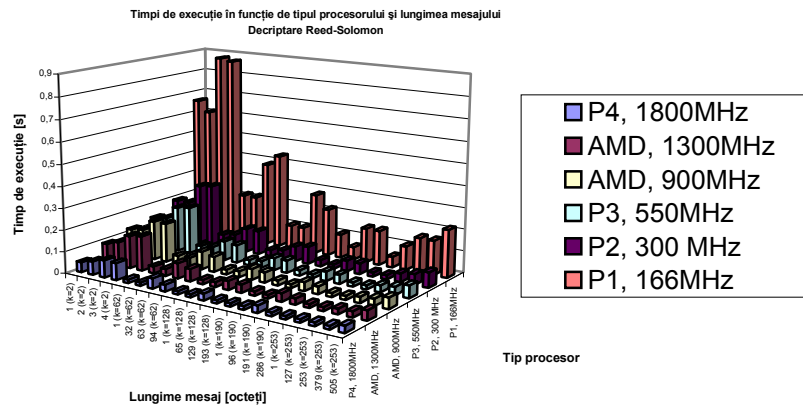
In this chapter I present the main experimental results, obtained with the aid of the implemented off-line EPS. The protocols were compared in basic and in fault tolerant form, and all the algorithms were executed on 6 different platforms (high-end and low-end processors). The tested protocols were:

- encryption-decryption McEliece,
- identifications: Schnorr and the own, proposed protocol
- withdrawal
  - using a smart card
  - cut and choose for single-term coins
  - cut and choose for multi-term coins
  - restrictive blindness
- payment protocol

For example, in the following two figures we can see the experimental results for modified McEliece cryptosystem, in function of processor type and message length.



### Encryption McEliece



### Decryption McEliece

The main conclusions are :

1. McEliece encoding needs a shorter time as decoding, because decoding is a more complex process.
2. The proposed identification algorithm is faster as Schnorr, because it uses matrix algebra against expansive exponential computing.
3. The proposed, modified forms of the protocols are a little bit slower as the basic forms, because there is an additional security level inserted, but this kind of modification give us an increased security and error correction capability. That's way, these protocols can be used very well in wireless applications, even in the presence of intensive noises.
4. The fastest withdrawal protocol is the withdrawal using a smart card, and the fastest withdrawal protocol without a tamper resistant hardware is the restrictive blindness.
5. Fault tolerance in case of coin loosing or conection interrupt cannot be obtained only with the aid of error correcting codes. In this case we must apply advanced fault tolerant techniques (logging, restore points, etc.), as we will see in the next chapter.

### Own contributions:

1. Test and comparison of the protocols studied in chapter 3. and 4. using the implementation described in chapter 5.
2. The tests were made on 6 different platforms, and the protocols were implemented in basic and in fault tolerant form.
3. Based on experimental result, we saw the advantages and disadvantages of each algorithm.
4. Possibility of optimization by computations executed before transactions.

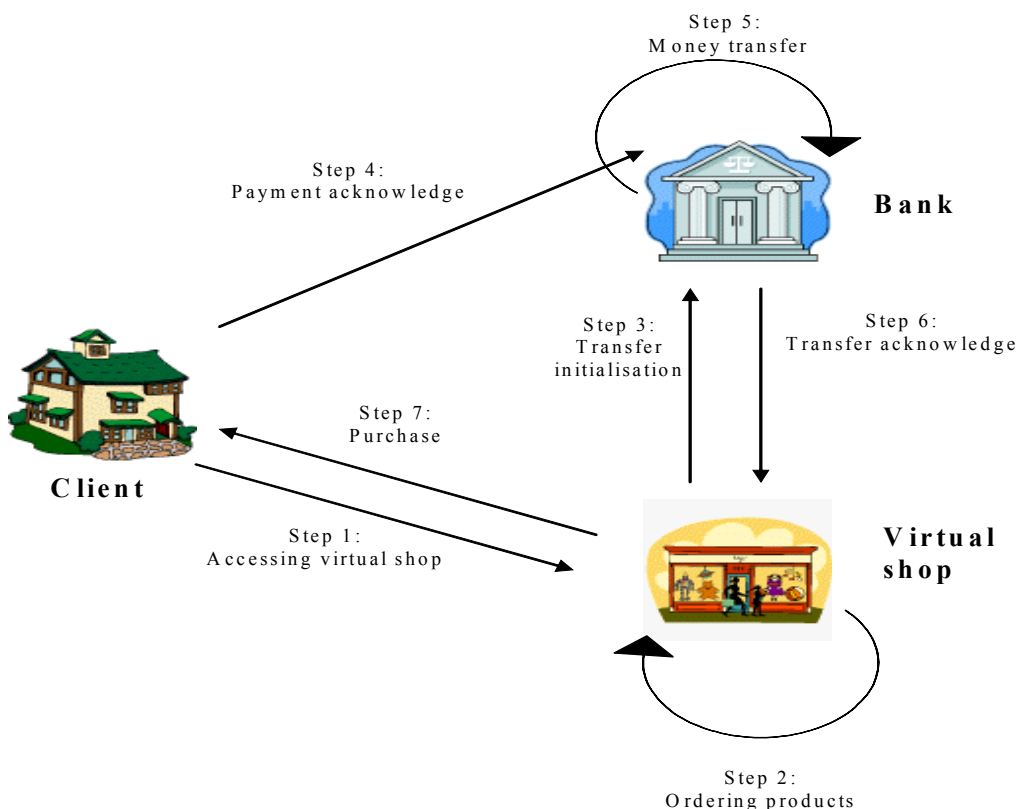
## 7. A new, fault tolerant, *on-line* electronic payment system

In the off-line EPS all the transactions are untraceable, and this is the reason, that only degradation integrity can be ensured. In the on-line EPS the payment protocol is supervised by a Trusted third Party, so the ambiguous transactions are not enabled – preventive integrity. That’s way, the on-line systems are spread more often as the off-line systems. In the existent EPS the main problem is security, and fault tolerance is ignored. This is the reason, why in this chapter I’d like to propose significant enhancements for EPS in sense of fault tolerance.

In the figure below I represent the general model of an on-line EPS, main parties, and transactions between them. The main parties are: client, virtual shop, bank. The transactions are:

- |                            |                         |
|----------------------------|-------------------------|
| 1. Accessing virtual shop  | 5. Money transfer       |
| 2. Ordering products       | 6. Transfer acknowledge |
| 3. Transfer initialization | 7. Purchase             |
| 4. Payment acknowledge     |                         |

In the thesis I described all the security and fault tolerant measures proposed, for each transaction all the possible defection scenarios, and treatments in each case.



### The arhitecture of the proposed on-line electronic payment system

After this, using Monte-Carlo method, I realized a simulation for the transactions success rate in function of the loss rate between client-bank, respectively between virtual shop and bank. For this, I defined terms: “success” and “not-success”, as follows:

$$Succes = (payment \wedge purchase) \vee (non-payment \wedge non-purchase) \tag{347}$$

$$Not-succes = (payment \wedge not-purchase) \vee (non-payment \wedge purchase) \tag{348}$$

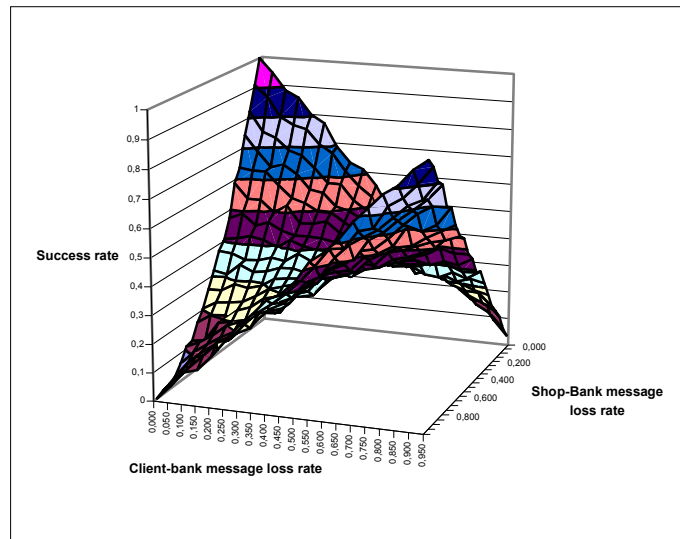
Analysing all the possible cases, I completed two tables (for success and for not-success) for the whole system, and I wrote a simulation program for computing the success rate in non tolerant case, respectively in fault tolerant case. The simulation results can be shown in the figures below. We can observe the **increasing of the success rate** in case of the fault tolerant system against the non fault tolerant case.

So, in this chapter, in order to obtain a secure and fault tolerant, on-line EPS, I analysed all the transactions between parties, discovering the vulnerable points of the system in case of possible malfunctions. Based on this analysis I proposed security and fault tolerant solutions (logging, atomic transactions, restore points) to obtain a more reliable system. The proposed solutions were tested by Monte-Carlo simulation, and finally we observed the increase of the transactions success rate in fault tolerant case.

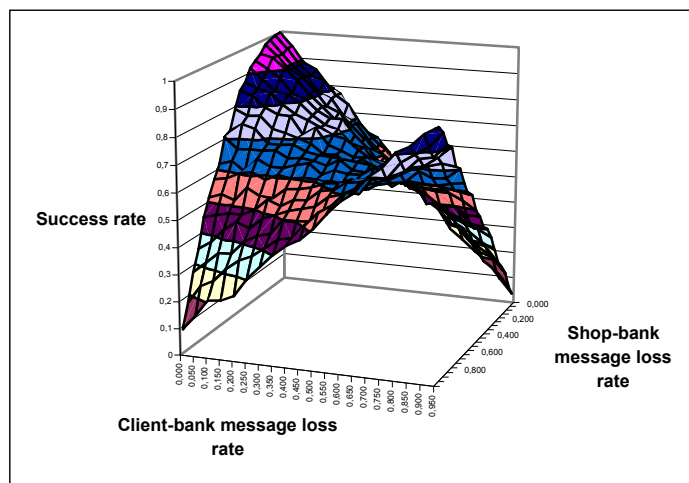
Own contributions :

1. Proposal and projecting of the fault tolerant EPS.
2. Analysis of all the malfunction scenarios, and proposal of the necessary security and fault tolerant measures.
3. Monte-Carlo simulation to compute the success rate in case of non fault tolerant system and for the fault tolerant case.

A possible enhancement for this on-line system could be the insertion of a “payment processor”, to treat inter-banking requests, in case of a world wide payment system.



**Transaction success rate in function of message loss rate.  
Non fault-tolerant case.**



**Transaction success rate in function of message loss rate.  
Fault-tolerant case.**

## 8. Final conclusions

The main goal in this thesis was finding new solution for increase the security and correctness (fault tolerance) of distributed computer systems. To achieve the proposed goal, I unified different research direction in the thesis: coding theory, cryptography and electronic payment systems. The theoretical results were applied in the experimental models projected and implemented in the last chapters. Using error correcting codes in cryptography is an efficient method to improve security and data integrity. A very important research direction is e-commerce. Because few existent EPS are fault tolerant, the theoretically results can be applied in practice to improve the payment systems in this sense. Fault tolerance can be obtained at two levels: low level (error correcting codes) and high level (advanced fault tolerant techniques). In the thesis I used the first method to improve off-line EPS, and the second method, to construct a fault tolerant on-line EPS.

The most important *own contributions* in the thesis are :

1. I proposed a *new algorithm* for the analysis of cyclic block codes, based on Discrete Fourier Transform, and rank computing.
2. I projected and I built a new cryptosystem, named “modified McEliece cryptosystem”, based on Reed-Solomon codes, with error correcting capabilities. This cryptosystem was used in construction of the off-line EPS.
3. A new, performant mutual identification scheme, based on modified McEliece signature.
4. Proposal of a new off-line EPS based on one-spendable electronic coins with error correcting capabilities. In this goal I defined all the parties and transactions existents in the system, and for each protocol I proposed enhancements to achieve fault tolerance: identification, withdrawal, payment and deposit. The security degree of the withdrawal protocol was computed using a probability approach.
5. Proposal of a new, fault tolerance, on-line EPS. After the system specification, I analysed all the transactions and all the possible malfunction scenarios, and for each one I proposed fault tolerant solutions.
6. I projected and I implemented a simulation algorithm based on Monte-Carlo method, to evaluate the success rate in the fault tolerant and non fault tolerant case, for the on-line EPS. The result was an increased success rate in the fault tolerant case.

Possible future researches could be focused on two main directions: *increasing security* and *increasing fault tolerance*. Using quantic cryptosystems, elliptic curve cryptography, and error correcting codes together with advanced fault tolerant techniques can improve significantly the security and fault tolerance. The results in this thesis can confirme that, by unifying these directions, it is possible to obtain *fault tolerant, secure distributed systems*.

## Selected references

- [1] [Barga, Lomat, 2003]  
Barga, R., Lomat, D., „Phoenix Project: Fault-Tolerant Applications”, Microsoft Research, 2003.
- [2] [Birman, 2005]

- Birmann P. Kenneth, "Reliable Distributed Systems. Technologies, Web Services and Applications", ISBN 10-0-387-21509-3, Springer Science and Business Media, Inc., 2005
- [3] **[Blahut, 1984]**  
Blahut E. Rechard, "Theory and practice of error control codes ", Addison Wesley, 1984
- [4] **[Brands, 1998]**  
Brands Stefan, "Electronic Cash", Brands Technologies, The Netherlands, Handbook on Algorithms and Theory of Computation, 1998, CRC Press, ISBN 0849326494
- [5] **[Bruss, Lütkenhaus, 2001]**  
Bruß Dagmar, Lütkenhaus Norbert, „Quantum Key Distribution: from Principles to Practicalities”, ISI, Torino, Italy, Helsinki Institute of Physics, Finland, October, 2001
- [6] **[Cheng, Siegel, Wu, 2000]**  
Cheng K. Michael, Siegel H. Paul, Wu Xin-Wen, "List decoding Reed-Solomon codes", Signal Transmission and Recording Group, University of California, San Diego, Proceedings on NSIC Annual Meeting, June 28, 2000
- [7] **[Coulouris, Dollimore, Kindberg, 2001]**  
Coulouris G., Dollimore, J., Kindberg, T., „Distributed Systems: Concepts and Design”, Addison-Wesley, 2001.
- [8] **[Friedrichs, 1996]**  
Friedrichs Bernd, “Kanalcodierung, Grundlagen und Anwendungen in modernen Kommunikationssystemen”, Springer Verlag, Berlin, Heidelberg, New York, 1996.
- [9] **[Ince, 2004]**  
Ince Darrel, “Developing Distributed and E-Commerce Applications”, Second Edition, Pearson – Addison Wesley, ISBN 0-321-15422-3, International Edition, 2004
- [10] **[Jalote, 1994]**  
Jalote, P., „Fault Tolerance in Distributed Systems”, Prentice Hall, 1994.
- [11] **[Lint, 1994]**  
Lint, J.H. van, "Introduction to Coding Theory", Second Edition, Graduate Texts in Mathematics vol. 86, Springer-Verlag, Berlin Heidelberg, 1994
- [12] **[Menzes, Oorschot, Vanstone, 1997]**  
Menzes A., Oorschot P.van, Vanstone S., “Handbook of Applied Cryptography”, CRC Press Inc., University of Waterloo, 1997.
- [13] **[Patriciu, Pietroşanu-Ene, Bica, Văduva, Voicu, 2001]**  
Patriciu Victor-Valeriu, Pietroşanu-Ene Monica, Bica Ion, Văduva Călin, Voicu Nicolae, "Securitatea comerţului electronic", Editura BIC ALL, Bucureşti, 2001
- [14] **[Patriciu, Pietroşanu-Ene, Bica, Cristea, 1998]**  
Patriciu Victor-Valeriu, Pietroşanu-Ene Monica, Bica Ion, Cristea Costel, "Securitatea informatică în Unix şi Internet", Editura Tehnică, Bucureşti, 1998
- [15] **[Pedragal-Martin, Ramamritham, 2001]**  
Pedragal-Martin, C., Ramamritham, K., Guaranteeing recoverability in electronic commerce, Proc. 3rd Int. Workshop WECWIS 2001
- [16] **[Poszet, 1999a]**  
Poszet O., „DFT and Linear Complexity of Sequences in Construction of Linear Error Detecting and Error Correcting Codes”, Analele Univ. Oradea, Fascicola Electrotehnica, Sectiunea D., pp. 67-72, 1999
- [17] **[Poszet, Ignat, 2002]**  
Poszet O., Ignat I., „About Information Security in Electronic Payment Systems”, Proceedings on the First Roedunet Conference, Cluj-Napoca, April, 2002
- [18] **[Poszet, Ignat, Praţa, Drăgan, 2002]**  
Poszet O., Ignat I., Praţa A., Drăgan H., „A probabilistic approach on the security of an EPS”, Proceedings on The Fourth International Conference on Renewable Sources and Environmental Electro-Technologies, R.S.E.E., Oradea, April, 2002
- [19] **[Poszet, Ignat, Drăgan, 2003a]**  
Poszet O., Ignat I., Drăgan H., „An Off-line Electronic Payment System Based on One-spendable Electronic Coins”, Proceedings on the Conference of Engineering of Modern Electric Systems, E.M.E.S., Section C., Oradea, May 29-31, 2003
- [20] **[Poszet, Ignat, Drăgan, 2003b]**  
Poszet O., Ignat I., Drăgan H., „Analysis And Design Of Error Correcting Cyclic Codes Using a New, Rank Bounding Algorithm”, Proceedings on the Conference of Engineering of Modern Electric Systems, E.M.E.S., Section C., Oradea, May 29-31, 2003
- [21] **[Poszet, Ignat, Vari-Kakas, Novac, 2004a]**  
Poszet O., Ignat I., Vari-Kakas S., Novac O. „Methods for Cyclic Codes Decodification”, Proceedings on The Fifth International Conference on Renewable Sources and Environmental Electro-Technologies, R.S.E.E., Oradea, May 2004, pp.61-65
- [22] **[Poszet, Vari-Kakas, 2005]**

- Poszet O., Vari-Kakas S., "An Efficient Identification Protocol for Electronic Payment Systems", Proceedings Budapest Tech, Hungary, nov. 2005
- [23] **[Poszet, Vari-Kakas, Novac, Ignat, 2006]**  
Poszet O., Vari-Kakas S., Novac O., Ignat I., "Fault tolerant protocols used in electronic payment systems", Proceedings on The 6th International Conference on Renewable Sources and Environmental Electro-Technologies, R.S.E.E., Oradea, June 2006
- [24] **[Radu, 1997]**  
Radu Cristian, "Analysis and Design of Electronic Payment Systems", Ph. D. Thesis, Katholieke Universiteit Leuven, Belgium, 1997
- [25] **[Rama, Vijayaraghavan, 2004]**  
Rama, S., Vijayaraghavan, V., Fault-tolerance in e-commerce web servers, ECE Department, Univ. of Wisconsin Madison, 2004.
- [26] **[Rivest, Shamir, Adleman, 1978]**  
Rivest R.L., Shamir A., Adleman L., "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", Communications of the ACM, vol.21, no.2, pp.120-126, 1978
- [27] **[Schaub, 1988]**  
Schaub Thomas, „A Linear Complexity Approach to Cyclic Codes”, Diss. ETH No 8730, Swiss Federal Institute of Technology, Zuerich, 1988
- [28] **[Schnorr, 1991]**  
Schnorr C.P., "Efficient signature generation by smart cards", Journal of Cryptology, Vol.4., No.3., 1991, pp.161-174]
- [29] **[Tilburg, 1994]**  
Tilburg Johan van, "Security-Analysis of a Class of Cryptosystems Based on Linear Error-Correcting Codes", Royal PTT Nederland NV, PTT Research, Leidschendam, 1994