Computer Science Journal of Moldova, vol.**16**, no.2(47), 2008

Digital Signature Scheme Based on a New Hard Problem*

Nikolay A. Moldovyan

Abstract

Factorizing composite number $n = qr$, where q and r are two large primes, and finding discrete logarithm modulo large prime number p are two difficult computational problems which are usually put into the base of different digital signature schemes (DSSes). This paper introduces a new hard computational problem that consists in finding the kth roots modulo large prime $p = Nk^2 + 1$, where N is an even number and k is a prime with the length $|k| \geq 160$. Difficulty of the last problem is estimated as $O(\sqrt{k})$. It is proposed a new DSS with the public key $y = x^k \bmod p$, where x is the private key. The signature corresponding to some message M represents a pair of the $|p|$-bit numbers S and R calculated as follows: $R = t^k \bmod p$ and $S = tx^{f(R,M)} \bmod p$, where $f(R, M)$ is a compression function. The verification equation is $S^k \bmod p = y^{f(R,M)} R \bmod p$. The DSS is used to implement an efficient protocol for generating collective digital signatures.

1 Introduction

Information authentication in computer networks and information systems is usually performed with digital signature schemes (DSSes) that are attributed to public key cryptosystems. The DSSes are based on some well investigated hard computational problems. The upper boundary of the DSS security level is defined by the difficulty of the

*This work was supported by Russian Foundation for Basic Research grants # 08-07-00096-a and # 08-07-90100-Mol_a.

used hard problem. To get sufficiently high security the signature generation and signature verification procedures use calculations modulo comparatively large numbers. The modulus length defines significantly performance of the DSSes.

The most efficient known DSSes are based on the following three difficult problems [17]:

1. Factorization of a composite number $n = qr$, where q and r are two large primes.

2. Finding discrete logarithm modulo large prime number p.

3. Finding discrete logarithm in a group of points of some elliptic curve.

The indicated problems are hard, if the used primes and elliptic curves satisfy special requirements [9, 17]. The first problem is used in the following cryptosystems: RSA [19], Rabin's DSS [18], and in DSSes proposed in [14, 15]. The second problem is used in ElGamal's DSS [4], Schnorr's DSS [24], American standard DSA [16], Russian standard GOST R 34.10-94 [5], and in some DSSes presented in [11, 14]. The third problem is used, for example, in American standard ECDSA [1] and Russian standard GOST R 34-2001 [6].

In general the security level of the DSS can be estimated as number of group operations required to forge a signature. In this paper the performance is compared for different DSS in the case of minimum security level that can be estimated at present as 2^{80} modulo exponentiation operations. Solving the difficult problem that is put into the base of some considered DSS allows one to calculate signatures corresponding to arbitrary messages. Therefore the security is less or equal to the difficulty of the hard problem that is put into base of the DSS.

In the best case the DSS is as secure as difficulty of the used computational problem. If for some DSS we can prove the last fact, then such DSS is called provably secure. In literature the formal proof of the security level is presented for Rabin's DSS [18, 22] and for a class of provably secure DSSes which generalize the Rabin's cryptosystem [12].

However these provably secure DSSes have not gained wide practical application because of their comparatively low performance.

The best known algorithms for solving the first two problems have subexponential complexity [10]. Therefore in the case of the RSA and Rubin's DSS the minimum length of the value n is 1024 bits. In these DSSes the value n is used as modulus while performing computations corresponding to signature generation and verification procedures. Respectively, to provide the minimum security level the ElGamal's DSS, Schnorr's DSS, DSA, and GOST R 34.10-94 should use computations modulo 1024-bit prime number.

The best known algorithms for solving the third hard problem has exponential complexity (for special class of elliptic curves) and its hardness is estimated as $O(\sqrt{q})$ operations of multiplication of points, where q is a prime order of the group of points of the considered elliptic curve ($O(\cdot)$ is the order notation [3]). Operations performed on points include computations modulo prime p such that $|p| \approx |q|$, where $|x|$ denotes the bit length of the value x. Due to exponential dependence of the hardness of the discrete logarithm problem in a group of points of elliptic curves the minimum security level can be provided using the modulus p having sufficiently small length ($|p| \geq 160$ bits). Therefore the performance of the DSSes based on elliptic curves is higher against mentioned above DSSes, addition of two points reqire perfoming several multiplications modulo p and some auxiliary operations though.

In present paper we consider in detail the approach to designing DSSes which has been proposed earlier in our patent application [13]. The approach is based on using a new hard computational problem. The rest of the paper is organised as follows. In Section 2 we show that in particular cases finding the kth roots modulo large prime number is a hard problem. Such cases correspond to prime modulus with the structure $p = Nk^s + 1$, where N is an even number, k is a large prime, and $s \geq 2$. Difficulty of this new hard problem is estimated as $O(\sqrt{k})$. In Section 3 we introduce new DSS and estimate that the minimum security is obtained with the length of $|k| \geq 160$ bits and the modulus length of $|p| \geq 1024$ bits. We also describe a modified DSS providing reduction of the signature length and show that the proposed

algorithms are efficient to implement protocol for generating collective digital signatures. Section 4 concludes the paper.

Below we use the following terms.

The kth power residue modulo p is a value a for which the congruence $x^k \equiv a \bmod p$ has solutions.

The kth power non-residue modulo p is a value b for which the congruence $x^k \equiv b \bmod p$ has no solution.

These terms generalize the well known terms *quadratic residue* and *quadratic non-residue* [3, 9, 10].

Also we use the following notations:

kR_p is the set of the kth residues modulo p;

kNR_p is the set of the kth non-residues modulo p;

$|x|$ denotes the length of the binary representation of the value x;

$[\sqrt{k}]$ means the integer part of $\sqrt{k}$;

$||$ is the concatenation operation;

$\omega_p(a)$ denotes the order of the element a modulo prime p;

$\#\{*\}$ denotes the number of elements in the set $\{*\}$;

$\{a : **\}$ denotes a set of elements a possessing the property $**$;

$\varphi(n)$ is Euler phi function of n.

2 A new hard computational problem

2.1 Computing roots modulo prime

Difficulty of finding roots modulo a composite number is used in some of the known DSSes: RSA, Rabin's DSS, and others [11]. Indeed, the RSA [19, 10] is based on calculations modulo n that is a product of two randomly chosen prime numbers r and q: $n = rq$. In RSA the public key represents a pair of numbers (e, n). The signature corresponding to some message M is a value S, which satisfies the following verification equation: $S^e \bmod n = H$, where H is the hash function value corresponding to M. To generate a valid signature one should calculate the eth root modulo n. This problem is difficult untill the composite number n is factorized. The owner of the public key knows the related private key that is a number d, which is inverse of e modulo $\varphi(n)$, where

$\varphi(n) = (r-1)(q-1)$. Thus, we have $ed \bmod \varphi(n) = 1$. The signature generation is performed as follows: $S = H^d \bmod n$. Security of the RSA is based on difficulty of calculating d while φ(n) is an unknown value. The $\varphi(n)$ value can be easily calculated after factorization of the modulus n, therefore divisors of n have to be kept in secret. Thus, in the case of the RSA the problem of finding roots is dependent on the factorization problem.

The signature verification equation in Rabin's cryptosystems can be described as follows:

$$S^2 \bmod n = H||R,$$

where pair of numbers (S, R) is signature and H is the hash function value corresponding to the signed message M. The second element in the signature is the randomization value. To generate a signature one should select at random such R that value $H||R$ is quadratic residue modulo n and calculate quadratic root $\sqrt{H||R} \bmod n$. The last represents a difficult problem until the value n is factorized. If the divisors q and r are known, then the roots $\sqrt{H||R} \bmod q$ and $\sqrt{H||R} \bmod r$ can be easily calculated [22]. Then, using the Chinese Remainder Theorem, one can find the minimum value S such that $S \equiv \sqrt{H||R} \bmod q$ and $S \equiv \sqrt{H||R} \bmod r$, i. e. $S = \sqrt{H||R} \bmod n$. We have two different roots modulo q and two different roots modulo r, therefore we have four different roots modulo n. Each of the lasts satisfies the signature verification equation. Thus, finding square roots modulo n is a difficult problem that also depends on the factorization problem.

The main difference between the RSA and Rabin's DSS consists in the following. In RSA we have $\gcd(e, \varphi(n)) = 1$, ($\gcd(e, q-1) = 1$ and $\gcd(e, r-1) = 1$), but in Rabin's DSS $\gcd(2, q-1) \neq 1$ and $\gcd(2, r-1) \neq 1$. Actually, the fact that $2|q-1$ and $2|r-1$ requires to use some special algorithm to calculate the square roots. For some random prime p and large prime divisor $k|p-1$ with probability very close to 1 the complexity of finding k roots $\sqrt[k]{a} \bmod p$, where a is one of the kth power residues modulo p, is sufficiently low. Indeed, if prime k is sufficiently large, then with high probability k does not devide $\frac{p-1}{k}$ and it is easy to find some value Δ such that k divides $\frac{p-1}{k} + \Delta$, i. e.

$\frac{p-1}{k} + \Delta = hk$, where h is an integer (note that k does not divide Δ). Then we have:

$$a^{\frac{p-1}{k}} \equiv 1 \bmod p \Rightarrow a^{\frac{p-1}{k}+\Delta} \equiv a^{\Delta} \bmod p \Rightarrow a^{hk} \equiv a^{\Delta' d} \bmod p,$$

where $d = \gcd(\Delta, p-1)$. Let $\Delta'' = \Delta'^{-1} \bmod p-1$. Then we have:

$$\left(\left(a^{1/d}\right)^{h\Delta''}\right)^k \equiv a \bmod p \Rightarrow a^{1/k} \equiv \left(a^{1/d}\right)^{h\Delta''}.$$

With high probability the value d is sufficiently small and the dth root can be easily found, for example, using the method described in Section 2.2.

If $k^2|p-1$, then the method described above does not work, i. e. in the case of the prime $p = Nk^s + 1$, where N is an even number and $s \geq 2$, computing the kth roots is difficult. Below we estimate the computational complexity of this hard probem. The primes p having the indicated structure and different length $|p|$ can be easily generated for many different values s and $|N|$ (some examples are presented in Appendix 1).

2.2 Computing the kth roots modulo $p = Nk^s + 1$

The following three facts are well known [7, 20, 21]:

1. There exist $\frac{p-1}{k}$ different values $a_j \in \mathrm{kNR}_p$, where $j = 1, 2, ..., \frac{p-1}{k}$, each of which is the kth power residue.
2. For some value $a \in \mathrm{kR}_p$ the congruence $a^{\frac{p-1}{k}} \equiv 1 \bmod p$ holds.
3. For some value $b_i \in \mathrm{kNR}_p$ the congruence

$$b_i^{\frac{p-1}{k}} \equiv e_i \bmod p,$$

where $e_i = \sqrt[k]{1} \bmod p \neq 1$ and $i = 1, 2, ..., k-1$, holds.

Using these facts, it is easy to show that each of the roots e_i defines exactly $\frac{p-1}{k}$ different values b_{ij}, where $j = 1, 2, ..., \frac{p-1}{k}$, such that $b_{ij}^{\frac{p-1}{k}} \equiv e_i \bmod p$. Indeed, we have

$$b_{ij}^{\frac{p-1}{k}} \equiv b_{ij'}^{\frac{p-1}{k}} \bmod p \Rightarrow \left(\frac{b_{ij}}{b_{ij'}}\right)^{\frac{p-1}{k}} \equiv 1 \bmod p \Rightarrow \frac{b_{ij}}{b_{ij'}} \bmod p = a_{j''},$$

i. e. the ratio $\frac{b_{ij}}{b_{ij'}} \bmod p$ is the kth power residue. There exist exactly $\frac{p-1}{k}$ different values $a_{j''}$, hence there exist exactly $\frac{p-1}{k}$ different values $b_{ij'}$. Therefore selecting at random a value t we have probabilities

$$\Pr\left(t^{\frac{p-1}{k}} \bmod p = 1\right) = \Pr\left(t^{\frac{p-1}{k}} \bmod p = e_i\right)$$

for all $i = 1, 2, ..., k-1$. This fact is used while estimating the complexity of the algorithm described below.

Suppose a random value $a \in \mathrm{kR}_p$ is selected. It is evident that $a^{\frac{p-1}{k^2}} \bmod p \in \{e_1, e_2, ..., e_{k-1}, 1\}$. Number of the elements $a \in \mathrm{kR}_p$ for which we have $a^{\frac{p-1}{k^2}} \bmod p = 1$ is equal to $Z_1 = \#\{a : \omega_p(a) \leq \frac{p-1}{k^2} = N\}$. Number of the values $a \in \mathrm{kR}_p$ for which we have $a^{\frac{p-1}{k^2}} \bmod p \neq 1$ is equal to $Z_2 = \#\{a : \omega_p(a) \leq \frac{p-1}{k} = Nk\}$ (see Theorem 2.10.6 in [3]). It is known (see Theorem 2.8.4 in [3]) that

$$\sum_{d|N, d>0} \varphi(d) = N \Rightarrow Z_1 = N \quad \text{and} \quad Z_1 = Nk.$$

Probability $\Pr\left(a^{\frac{p-1}{k^2}} \equiv 1 \bmod p\right) = \frac{Z_1}{Z_2} = k^{-1}$ is negligible. With probability $1 - k^{-1}$ we have $a^{\frac{p-1}{k^2}} \bmod p = e_i \neq 1$.

Taking into account that for each i there exists i' such that $e_{i'} = e_i^{-1} \bmod p$ we can write $a^{\frac{p-1}{k^2}} e_{i'} \equiv 1 \bmod p$, therefore

$$a^{\frac{p-1}{k^2}} b^{\frac{p-1}{k}} \equiv a^N b^{\frac{p-1}{k}} \equiv 1 \bmod p, \tag{1}$$

where $b \in \mathrm{kNR}_p$.

If congruence (1) is fulfilled, then we can easily calculate a root $\sqrt[k]{a} \bmod p$. Indeed, congruence (1) can be represented as

$$a^k b^{\frac{p-1}{k^2} k} \equiv a^{k-N} \bmod p, \tag{2}$$

where with sufficiently high probability we have $\gcd(k - N, p - 1) = 1$. Suppose that the last relation holds (in other case the problem

is only a bit more complex). Then it is possible to compute value $N' = (k - N)^{-1} \bmod p - 1$. Therefore we get

$$a^{N'k} b^{N' \frac{p-1}{k^2} k} \equiv a \bmod p,$$

hence

$$\left(a^{N'} b^{N' \frac{p-1}{k^2}} \right)^k \equiv a \bmod p. \tag{3}$$

Congruence (3) shows that value $X = a^{N'} b^{N' \frac{p-1}{k^2}} \bmod p$ represents one of roots $\sqrt[k]{a} \bmod p$. Other $k-1$ roots $\sqrt[k]{a} \bmod p$ can be computed using the formula $e_i X \bmod p$, $i = 1, 2, ..., k-1$ (roots $\sqrt[k]{1} \bmod p$ can be found computing the sequence $\{\epsilon, \epsilon^2 \bmod p, ..., \epsilon^{k-1} \bmod p, \epsilon^k \bmod p = 1\}$, where ϵ is the kth order element modulo p).

A value $b \in \mathrm{kNR}_p$ satisfying congruence (1) can be computed as follows. The value b can be represented as $b = b_i b_j \bmod p$, where $b_i, b_j \in \mathrm{kNR}_p$:

$$a^{\frac{p-1}{k^2}} b_i^{\frac{p-1}{k}} b_j^{\frac{p-1}{k}} \equiv 1 \bmod p \Rightarrow$$

$$a^{\frac{p-1}{k^2}} b_i^{\frac{p-1}{k}} \equiv b_j^{-\frac{p-1}{k}} \bmod p. \tag{4}$$

The following algorithm finds the required values b_i and b_j with high probability.

1. Select at random a value b_i and calculate the value $A_i = a^{\frac{p-1}{k^2}} b_i^{\frac{p-1}{k}} \bmod p$. Construct a table with entries (A_i, b_i) for $i = 1, 2, ..., [\sqrt{k}] + \Delta$, where $\Delta \ll [\sqrt{k}]$. Complexity of this step is $O(\sqrt{k})$ exponentiation operations.

2. Select at random a value b_j and calculate the value $B_j = b_j^{-\frac{p-1}{k}} \bmod p$. Construct a table with entries (B_j, b_j) for $j = 1, 2, ..., [\sqrt{k}] + \Delta$, where $\Delta \ll [\sqrt{k}]$. Complexity of the second step is $O(\sqrt{k})$ exponentiation operations.

3. Sort the first table by component A_i. Complexity of this step is $O(\sqrt{k} \cdot |k|)$ comparison operations.

4. For $j = 1$ to $[\sqrt{k}]+\Delta$ check if the value B_j is equal to the value of the first component of some entry in the first table. Complexity of this step is $O(\sqrt{k} \cdot |k|)$ comparison operations.

This algorithm requires storage for about $4\sqrt{k}$ (i. e. $O(\sqrt{k})$) $|p|$-bit numbers. For randomly selected b_i and b_j we have $\Pr(A_i = B_j) = k^{-1}$, therefore in two tables each of which contains $\sqrt{k} + \Delta$ random values with probability more than 0.5 there are equal values $A_{i_0} = B_{j_0}$ (see birthday paradox [17, 10]). Thus, with probability about 0.5 the algorithm finds values b_{i_0} and b_{j_0} satisfying congruence (4). Having such values we can easily compute the value $b = b_{i_0} b_{j_0} \bmod p$ satisfying congruence (1) and then compute $X = \sqrt[k]{a} \bmod p$. On the whole the complexity of the algorithm can be estimated as $\approx 2\sqrt{k}$ modulo exponentiation operations. Trying the algorithm several times we will get value X with probability close to 1. Hardness of this procedure is $W = O(\sqrt{k})$. If $|k| = 160$, then $W \approx 2^{80}$ exponentiation operations. Thus, the hardness depends exponentially on the value $\sqrt{k}$ and in the case $|k| = 160$ the considered problem is computationally infeasible (untill a new significantly more efficient algorithm is developed).

2.3 Dependence on the discrete logarithm problem

In Section 2.2 we have proposed a direct algorithm for finding the kth roots which works for arbitrary large values $|p|$. In the case of sufficiently small size of the value p the kth roots can be computed by means of finding discrete logarithm as follows.

1. Generate a primitive element g modulo p.

2. Calculate logarithm $\log_g a \bmod p$.

3. Divide $\log_g a \bmod p$ by k (at this step it gets the value $\log_g \sqrt[k]{a} \bmod p$; note that logarithm corresponding to some value $a \in \mathrm{kR}_p$ is multiple to the value k).

4. Exponentiate g to the power $z = \log_g \sqrt[k]{a} \bmod p$ and get the value $\sqrt[k]{a} = g^{\log_g z} \bmod p$.

Let us justify the division operation that is to be performed at step 3. If $a \in \mathrm{kR}_p$, then there exists some value x for which $x^k \equiv a \bmod p$ holds. The last expression can be represented as follows

$$\left(g^{\log_g x}\right)^k \equiv g^{k\cdot\log_g x} \equiv a \equiv g^{\log_g a} \bmod p,$$

i. e. $k|\log_g a$. For values $|p| \approx 1024$ bits difficulty of finding logarithms is approximately equal to 2^{80} operations [8]. Therefore in the case $p = Nk^2+1$, $|k| = 160$ bits, and $|N| < 704$ bits the algorithm described in this section is more efficient. If $|N| > 704$ bits, then the algorithm described in Section 2.2 is more efficient.

3 New digital signature scheme

3.1 Initial design

New hard computational problem described in Section 2 is used in the DSS described below. It uses the prime modulus having the structure $p = Nk^2 + 1$, where k is a large prime ($|k| \geq 160$) and N is such even number that $|p| \geq 1024$ bits. A random value x is selected as a private key. The public key y is computed using the formula $y = x^k \bmod p$. The signature represents a pair of $|p|$-bit numbers S and R.

Suppose a message M is given. The signature generation procedure is performed as follows:

1. Select at random a value $t < p-1$ and calculate the first element of the signature: $R = t^k \bmod p$.

2. Using some specified hash function $F_H(M)$ calculate the hash value H corresponding to the message M and compute $f(R, M) = RH \bmod \delta$, where δ is a large prime that is a part of the signature generation algorithm. For example, it is acceptable to use a randomly selected prime δ such that $|\delta| = 160$. The function $F_H(M)$ is also a part of the DSS. For example, one can use the hash function SHA-1 [17] recommended by US National Institute of Standards and technology (NIST).

3. Calculate the second element of the signature: $S = x^{f(R,M)}t \bmod p$.

The signature verification procedure is defined by the following congruence:

$$S^k \equiv y^{f(R,M)} R \bmod p.$$

The signature length is $|S| + |R| \approx 2|p|$.

Accordingly to the hard problem put into base of this DSS the direct way to attack this cryptosystem is to compute one of the kth roots from the public key (it is not necessary to guess exactly the value of the secret key x, since each of the values $xe_i \bmod p$ can be used to generate a valid signature).

The next direct attack is connected with computing one of the kth degree roots from the signature element R. Suppose the attacker has computed a root $\sqrt[k]{R} \bmod p = t'$, where the element R satisfies the condition $\gcd(f(R,M), p-1) = 1$ (the respective signature can be selected from several known valid signatures). Then the attacker can compute integer $f' = f^{-1}(R,M) \bmod p-1$. The value t' is connected with t: $t' = te_i \bmod p$, therefore he can compute the value $x' = (S/t')^{f'} \bmod p = (S/t)^{f'} e_i^{-f'} \bmod p = xe_i' \bmod p$ that is one of the kth roots from the secret key. These attacks are infeasible at present, if $|k| \geq 160$ bits.

Let us consider an attack that for some given values $R \in \mathrm{kR}_p$ and H allows one to generate a valid signature (R,S). Such attack can be easily used to find the kth roots from y using the following algorithm.

1. Select at random a value t $(0 < t < p)$.

2. Calculate $R = t^k \bmod p$, $E = f(R,M) = RH \bmod \delta$, and $\gcd(p-1, E)$. If $\gcd(p-1, E) \neq 1$, then go to step 1.

3. Using the attack generate the signature (R,S). Under our assumption the signature (R,S) satisfies the congruence $S^k \equiv y^E R \bmod p$.

4. Calculate the values $E' = E^{-1} \bmod p-1$ and $\sqrt[k]{y} = (S/t)^{E'} \bmod p$.

Thus, any attack like the considered one is as difficult as finding the kth roots modulo p (any value $a \in \mathrm{kR}_p$ can be used in the considered algorithm as the public key value y).

There are also possible some hypothetic attacks that for some given values $y \in \mathrm{kR}_p$, S, and H allow one to compute such value R that the

signature (R, S) satisfies the verification congruence. This means that attacker is able to solve the congruences like $Ry^{f(R,M)} \equiv C \bmod p$, where C is a constant. However the difficulty of solving such congruences modulo large prime p relatively R is put into the base of the GOST R 34.10-94, ElGamal's, and Schnorr's DSSes.

3.2 Modified DSS

It is possible to reduce the signature length using the value $E = f(R, M)$ as signature element instead of the R element. A variant of the modified DSS is presented by the following verification procedure:

1. Using the signature (E, S) compute $R = S^k y^{-E} \bmod p$.
2. Calculate $E' = f(R, M) = RH \bmod \delta$.
3. Compare E' with E. If $E' = E$, then the signature is valid.

In the modified DSS the signature length is equal to $|E| + |S| = |\delta| + |S| \approx |p|$. A numerical illustration of the modified DSS is presented in Appendix 2.

The straightforward attacks mentioned in Section 2.1 are also applicable to the modified DSS. Let us consider another attack that is efficient in the case of small size of the compression function $f(R, M)$, for example in the case of small value $|\delta|$. It can be performed as the following algorithm.

1. Select at random the values $E < \delta$ and $S < p$.
2. Compute value $R' = S^k y^{-E} \bmod p$.
3. Calculate value $E' = f(R', M)$.
4. Compare E' and E. If $E' \neq E$, then jump to step 1.

On the average the work effort of this algorithm is $\approx 2\delta$ exponentiation operations, since $\Pr(E' = E) = \delta^{-1}$. This attack is infeasible at present, if $|f(R, M)| \geq 160$ bits.

3.3 Collective signature protocol

Suppose the jth user owns the public key y_j depending on his private key $x_j < p$ as follows: $y_j = x_j^k \bmod p$, where $j = 1, 2, ..., \mu$. Suppose an electronic document M is given and m $(m \leq \mu)$ users owning the public keys $y_{\alpha_1}, y_{\alpha_2}, ..., y_{\alpha_m}$ should sign it simultaneously. The following

protocol produces a collective digital signature (CDS) and solves the indicated problem more efficiently than the known protocols for signing simultaneously a contract [23].

1. Each α_ith user selects at random a value $t_{\alpha_i} < p$ and computes the public value $R_{\alpha_i} = t_{\alpha_i}^k \bmod p$, where $i = 1, 2, ..., m$.

2. Some of the users (or one of them) calculate the common randomization value

$$R = R_{\alpha_1} R_{\alpha_2} ... R_{\alpha_m} \bmod p$$

and then calculate the first part of the CDS $E = f(R, M)$, where f is a specified compression function. For example, we will use the following function $E = RH \bmod \delta$, where δ is a large prime having length $|\delta| = 160$ and H is a hash value computed from the message M.

3. Using the values R and t_{α_i} each α_ith user computes its share in the CDS:

$$S_{\alpha_i} = x_{\alpha_i}^E t_{\alpha_i} \bmod p$$

that is supposed to be available to all users of the group.

4. Calculate the second element of the CDS:

$$S = S_{\alpha_1} S_{\alpha_2} ... S_{\alpha_m} \bmod p.$$

Thus, the CDS is computed with $2m$ modulo exponentiations. The CDS length is fixed and equals to $|S| + |\delta|$.

The CDS verification procedure is performed as follows.

1. Compute the collective public key y: $y = y_{\alpha_1} y_{\alpha_2} ... y_{\alpha_m} \bmod p$.

2. Using the CDS (E, S) compute value R':

$$R' = S^k y^{-E} \bmod p.$$

3. Compute $E' = R'H \bmod \delta$.

4. Compare values E' and E. If $E' = E$, then the signature is valid. Otherwise the signature is false.

The proposed algorithm can be used in the collective signature protocols that are free of the participation of any trusted party that is needed in the multisignature protocol proposed in [2] for implementing the fixed size signature corresponding to some group of users. In our algorithm we need no declosure of any user's private key, while generating the collective digital signature with fixed size.

4 Conclusion

A new hard computational problem have been proposed as cryptographic primitive. The problem consists in finding the kth roots modulo large prime $p = Nk^s+1$, where k is a large prime such that $k \geq 160$ bits and N is an even number such that $|N| + s|k| \geq 1024$ bits. We have proposed an algorithm for solving this problem. The complexity W of the algorithm depends exponentially on $|k|$ and is described with the formula $W = O(\sqrt{k})$.

Using a novel hard computational problem a new DSS has been proposed. The proposed DSS has been used to design a collective digital signature protocol that produces the fixed size signature shared by arbitrary number of users $m \geq 1$.

We hope that this paper will initiate other attempts aimed to developing new efficient algorithms for solving the proposed computational problem. Such researches will contribute to the security estimation of the presented new DSS and to evaluation of the proposed computational problem as a new cryptographic primitive.

References

[1] ANSI X9.62 and FIPS 186-2. *Elliptic curve signature algorithm*, 1998

[2] A.Boldyreva. *Efficient Threshold Signature, Multisignature and Blind Signature Schemes Based on the Gap-Diffi-Hellman-Group Signature Scheme.* Springer-Verlag LNCS, 2003, vol. 2139, pp. 31–46.

[3] J. Buchmann. *Introduction to Cryptography.* Springer-Verlag. Berlin, 2003. - 335 p.

[4] ElGamal T. *A public key cryptosystem and a signature scheme based on discrete logarithms.* IEEE Transactions on Information Theory. 1985, Vol. IT-31, No. 4. pp.469–472.

[5] GOST R 34.10-94. *Russian Federation Standard. Information Technology. Cryptographic data Security. Produce and check procedures of Electronic Digital Signature based on Asymmetric Cryptographic Algorithm.* Government Committee of the Russia for Standards, 1994 (in Russian).

[6] GOST R 34.10-2001. *Russian Federation Standard. Information Technology. Cryptographic data Security. Produce and check procedures of Electronic Digital Signature.* Government Committee of the Russia for Standards, 2001 (in Russian).

[7] L.H.Hua. *Introduction to Number Theory.* Springer, Berlin, Heidelberg, New York, 1982.

[8] International Standard ISO/IEC 14888-3:2006(E). *Information technology – Security techniques – Digital Signatures with appendix – Part 3: Discrete logarithm based mechanisms.*

[9] N. Koblitz. *A Course in Number Theory and Cryptography.* Springer-Verlag. Berlin, 2003. - 236 p.

[10] A.J.Menezes, P.C. Van Oorschot, and S.A.Vanstone. *Handbook of Applied Cryptography.* CRC Press, Boca Raton, FL, 1997.

[11] A.A.Moldovyan, D.N.Moldovyan, L.V.Gortinskaya. *Cryptoschemes Based on New Signature Formation Mechanism.* Computer Science Journal of Moldova. 2006.vol. 14. No 3 (42). pp. 397–411.

[12] N.A.Moldovyan, A.A.Moldovyan. *Class of Provably Secure Information Authentication Systems.* 4th Int. Workshop MMM-ANCS'07 Proc. September 13-15, 2007, St. Petersburg, Russia. Springer Verlag CCIS. 2007. vol. 1, pp. 147–152.

[13] N.A.Moldovyan, A.A.Moldovyan. *A Method for Generating and Verifying Electronic Digital Signature Certifying an Electronic Document.* Russian patent application # 2007129254, July 30, 2007.

[14] N.A.Moldovyan. *New Public Key Cryptosystems Based on Difficulty of Factorization and Discrete Logarithm problems.* 3d Int. Workshop IF&GIS'07 Proc. May 28-29, 2007, St.Petersburg, Russia. Springer LNGC. 2007. Vol. XIV. pp. 160–172.

[15] N.A.Moldovyan. *Cryptoschemes Based on New Signature Formation Mechanism.* Computer Science Journal of Moldova. 2006. vol. 14. No 3 (42). pp. 390–396.

[16] National Institute of Standards and Technology, NIST FIPS PUB 186. Digital Signature Standard, U.S. Department of Commerce, 1994.

[17] Pieprzyk J., Hardjono Th., and Seberry J. *Fundamentals of Computer Security.* Springer-Verlag. Berlin, 2003. - 677 p.

[18] Rabin M.O. *Digitalized Signatures and Public Key Functions as Intractable as Factorization.* - Technical report MIT/LCS/TR-212, MIT Laboratory for Computer Science, 1979.

[19] R.L.Rivest, A.Shamir, and L.M.Adleman. *A Method for Obtaining Digital Signatures and Public Key Cryptosystems.* Communications of the ACM, 1978, vol. 21, no 2, pp. 120–126.

[20] H.E.Rose. *A Course in Number Theory.* 2nd ed. Oxford: Clarendon Press, 1994.

[21] K.H.Rosen. *Elementary Number Theory and its Applications.* 4th ed. Reading, MA: Addison-Wesley Publishing Company, 2000.

[22] N.Smart. *Cryptography: an Introduction. McGraw-Hill Publication.* London, 2003.

[23] B. Schneier. *Applied Cryptography.* Second Edition, John Wiley & Sons, Inc. New York, 1996. 758 p.

[24] Schnorr C.P. *Efficient signature generation by smart cards.* J. Cryptology. 1991. vol. 4. pp. 161–174.

Appendix 1.

Examples of different primes having the structure $p = Nk^s + 1$.

We have composed a computer program generating the primes $p = Nk^s + 1$, where k is a prime. Our experiment has shown that for some pairs of the values s and N there exist no primes k such that $p = Nk^s + 1$ is a prime. For example, there exist no primes p for even s and $N \equiv 2 \bmod 6$. Another interesting example of the negative cases is presented by the pair $s = 4$ and $N = 4$. For odd values s there are also some exceptions.

However primes p having different length exist in the majority of the cases. The experiment has shown that there exist sufficiently large portion of the primes p (such that $2 \leq |k| \leq 320$ bits and $5 \leq |p| \leq 2048$ bits) for even s and $N \equiv Z \bmod 6$, where $Z = 0$ or $Z = 4$. For odd values s we also have many positive cases corresponding to different values N. For example, we have the following primes p.

Case $N = 10$, $s = 2$, and $k = 725884693$ gives prime:
$p = 5269085875317042491$.

Case $N = 2$, $s = 3$, and $k = 1337274629$ gives prime:
$p = 4782905620790796691857520379$.

Case $N = 4$, $s = 2$, and $k = 67433803869294133$ gives prime:
$p = 18189271617129713532911550672886757$.

Case $N = 4$, $s = 2$, and $k = 534215924076685141269133$ gives prime:
$p = 1141546614148426492824805789757003540711754286757$.

Case $N = 10000$, $s = 4$, and $k = 232667804209296877$ gives prime:
$p = 2930523206883470465880672783646732965482432585330676917965$
0044728906410001.

Case $N = 10$, $s = 2$, and
$k =$
21099217031501566983116540704831224275519342156267172191246 3
gives prime:
$p = 44517695934240579622597117963291205065768804389087691871 5$
$62708634450957516024988297600437002758657651782094950234726$
$3691.$

Case $N = 666666666666666666666664$, $s = 2$, and
$k = 353809991516667947508673$ gives prime:
$p = 83454340064683096655339600240446993579220192592930435$
$4376574202394108 57.$

Case $N = 4444$, $s = 5$, and
$k =$
63425033526343690366405826766779313666870338333903512434121 9
gives prime:
$p = 45611838462977930940136670572684327115554421236798891647 61$
$71602844372230180400997482458217991670744699006594761723 28$
$74210223268000533827948655237896653849530935472802618526 00$
$17443169267690232589235848680668262500188955199901307039 71$
$12250837616065101033961052714266266422492104636304917129 73$
$5122836127957.$

Case $N = 2222222$, $s = 5$, and
$k =$
27878674999896483124318821076504625160720702130261695915035 3
gives prime:
$p = 37423817620939885071521894476321909036491631622873355830 67$
$14049364451684595324178242570342515422784964745484530084 40$
$25753747981488549676110514907677079189817948309288602736 75$
$44888959669284542342963622725196050519509771417021849786 90$
$65745634227392624152354599321925890133857616649716234554 38$
$97670489118447.$

Appendix 2.

A numerical example with artificially small parameters.

This example illustrates the signature generation and verification procedures in the DSS based on difficulty of finding roots modulo large prime $p = Nk^2 + 1$, where $k = 132104433635297779312031$ is prime and $N = 238$,
i. e. $p = 41534763698924652690128708976232823900474001007 19$.

Suppose a user private key is $x = 3526378981324543353612$ and the hash function value calculated from the document to be signed is $H = 73568790119017231823457$. Then his public key is $y = x^k \bmod p = 38648581002193529403697748477885520183670551977 06$.

The signature is represented by a pair of numbers (R, S) which is generated as follows.

1. Generate a random number $t = 87193323415243553115136314$.

2. Calculate the value $R = t^k \bmod p$:
$R = 19651943943290548836692335934353542255535285430 48$.

3. Calculate the value E using the formula $E = RH \bmod \delta$, where $\delta = 35488784369499179$:
$E = RH \bmod \delta = 30895498554403274$.

4. Calculate the value S using the formula $S = x^E t \bmod p$:
$x^E \bmod p = 41428960321742932773705412861186030086874128465 12$;
$S = x^E t \bmod p =$
$= 34593999607864752462932100387875805936013775274 59$.

The signature verification is performed as follows.

1. Calculate $E' = (S^k y^{p-E-1} \bmod p)H \bmod \delta$:
$S^k \bmod p = 12236286329907996953803299609450507757550316562 37$;
$p - E - 1 \bmod p - 1 =$
$= 41534763698924652690128708976232514945488456974 44$;
$y^{p-E-1} \bmod p =$

$= 39407982034742155741062810189353996402481761395 23;$
$R' = S^k y^{p-E-1} \bmod p =$
$= 19651943943290548836692335934353542255535285430 48;$
$E' = R'H \bmod \delta = 30895498554403274.$

2. Compare the values E' and E.

The signature is valid, since $E' = E$.

N. A. Moldovyan,

Received July 7, 2007

Specialized Center of Program Systems "SPECTR",
Kantemirovskaya, 10, St.Petersburg 197342, Russia;
ph./fax.7-812-2453743,
E–mail: *nmold@cobra.ru*

Computer Science Journal of Moldova, vol.**16**, no.2(47), 2008

New approaches for development, analyzing and security of multimedia archive of folklore objects

Galina Bogdanova, Todor Todorov, Tsvetanka Georgieva

Abstract

We present new approaches used in development of the demo version of a WEB based client/server system that contains an archival fund with folklore materials of the Folklore Institute at Bulgarian Academy of Sciences (BAS). Some new methods for image and text securing to embed watermarks in system data are presented. A digital watermark is a visible or perfectly invisible, identification code that is permanently embedded in the data and remains present within the data after any decryption process. We have also developed improved tools and algorithms for analyzing of the database too.

1 Introduction

Computerizing of science work in the folklore sphere is one of the newest modern problems, which should be comprehended not only as folklore texts collecting, but also in their wider context connection. The national collecting and conserving center for the Bulgarian folklore in the folklore Institute of BAS is the only special organization for documenting, conservation and popularizing the folklore in the country. It has a unique collection of authentic materials on different bases: paper documents, audio- and video cassettes, reel. Section 2 presents some basic challenges in digital organization of folklore materials. Section 3 is for the organization of a web based applications. In Section 4 we consider basics of relational database management systems. Section 5 is devoted to the basics of watermarking security. In Section 6 we present basics of

error-correcting codes. Section 7 considers the structure of the demo-version of Web based archive of folklore objects. Section 8 is devoted to the security of the system. In Section 9 is presented analyzing of multimedia archive of folklore objects. In Section 10 we investigate other folklore archives and compare them with the Web based archive of folklore objects considered in this paper.

2 Contemporary methods for digital organization of folklore archives

The idea of digital preservation of folklore texts and other folklore objects and their analysis with computer means, has been used since the early stages of such technique. Folklore explorers and programmers have already tried and succeeded in accomplishing this idea. Such pieces of work were made by S.E.Nikitina (Institute of Linguistics, Russian Academy of Science), Y.I.Smirnov (Institute of the World Literature Russian Academy of Sciences), the Russian folklore republican center and the folklore archives RGGU, IVGI RGGU etc. Some Web sites in the folklore area are [10], [9], [22], [26]. Actually computerizing of science work in the folklore sphere is one of the newest modern problems, which should be comprehended not only as folklore texts collecting, but also in their wider context connection.

There are two basic types of folklore information systems:

1) Data base text and electronic indexes;

2) Searching systems adapted to work with folklore objects.

The Database creation should be very precise as a method for folklore analysis with specific structure - irregularly, very branching with the "by hand" operating.

The national collecting and conserving center for the Bulgarian folklore in the folklore Institute of BAS contains examples in different spheres of traditional and contemporary culture: verbal folklore (fairy-tale and not fairy-tale prose, proverbs, sayings and etc.), rites

and festivity (calendar, family, labor etc.), musical, dance and plastic folklore (clothes, belongings, crafts etc.). Beside materials, collected from collaborators of the Folklore Institute, in the National Center terrain records of students' expeditions, as well as unique personal archives of famous folklore people and crafts are preserved. All country regions are available as well as the Bulgarian Diaspora in Czech, Slovakia, Hungary, Moldova, Ukraine. The regional genre makes the Center's collection very valuable and significant for our national culture. Today the archive funds of the Center are consistently filled up with folklore materials.

The development of calculating technique made collecting and preserving information easier. Great attention is paid to the protection of information from unauthorized access. During the last decade computer steganography has approved as a protective measure.

3 Organization of web-based applications

Most of the web-based applications now are using a multilayered architecture. It is organized as a two - tier structure. On the first level there is an application space. It consists of a web-browser which communicates with a web server via HTTP protocol. On the top level of the application space there are the server side applications. These are web servers, CGI scripts and API's for database connection. The second level of organization is the database tools. The most important application in this level is the database server used for storage and organization of the data in the system. Also we have tools and libraries for working with these database management systems. HTML (Hypertext Markup Language) is still the more important technology for visualizing in the Web. Although HTML evolved and many improvements have been added, it is in itself still static. The next step is the dynamic Web technologies, which allow the building of active web sites.

TECHNOLOGIES FOR BUILDING ACTIVE WEB SITES

These technologies could be classified as follows:

1) Dynamic technologies from the client side

   1.1) Java applets

   2.2) Active X controls

   3.3) DHTML (Dynamic HTML)

2) Dynamic technologies from the server side

   2.1) Common Gateway Interface (CGI)

   2.2) Active Server Pages (ASP)

   2.3) Java Servlets and Java Server Pages (JSP)

   2.4) PHP

   2.5) Patented API's for Web servers(ISAPI and NSAPI)

   2.6) Server Side JavaScript (SSJS)

More detailed explanation about all these technologies is given in [4]. We use DHTML and PHP in the development of demo version of web based client/server system with folklore objects. DHTML is not a standard defined by the World Wide Web Consortium (W3C). It is a combination of technologies used to create dynamic Web sites. DHTML allows to use regular HTML, scripts, document object model (DOM), absolute positioning, dynamic styles, multimedia filters and many other technologies for dynamic text and graphic manipulation, which HTML shows on the screen. Dynamic styles are based on Cascading Style Sheets (CSS). With CSS we have a style and layout model for HTML documents. CSS was a breakthrough in Web design because it allowed developers to control the style and layout of multiple Web pages all at once. To make a global change, we could change the style, and all elements in the Web are updated automatically. The HTML DOM defines a standard set of objects for HTML, and a standard way to access and manipulate HTML objects. Internet Explorer supports two languages for writing scripts - Visual Basic Script (VBScript) and JavaScipt. PHP stands for PHP: Hypertext Preprocessor. PHP is a server-side scripting language and works like ASP and JSP: the sections with scripts are inside the HTML page. It runs on different platforms (Windows,

Linux, Unix, etc.) and supports many databases (MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, etc.). As a WEB server for PHP scripts many different servers (Apache, IIS, etc.) can be used but we choose Apache HTTP Server. It is the only one that can use PHP as an internal module. This improves performance and security in the working process.

4 Relational databases and relational database management systems (RDBMS)

Database is a collection of records stored in a computer in a systematic way, so that a computer program can consult it to answer questions. There are a number of different ways of organizing the data in the database called modeling of the database structure. The model in most common use today is the relational model, which in layman's terms represents all information in the form of multiple related tables each consisting of rows and columns. Relational databases are combination of interconnected and stored in one place data with the presence of such minimal excess that allows their use in optimal way for one or more applications [29]. The fundamental assumption of the relational model is that all data are represented as mathematical n-ary relations, an n-ary relation being a subset of the Cartesian product of n domains. In the mathematical model, reasoning about such data is done in two-valued predicate logic, meaning there are two possible evaluations for each proposition: either true or false (and in particular no third value such as unknown, or not applicable, either of which are often associated with the concept of NULL). The relational model of data permits the database designer to create a consistent, logical representation of information. Consistency is achieved by including declared constraints in the database design, which is usually referred to as the logical schema. The theory includes a process of database normalization whereby a design with certain desirable properties can be selected from a set of logically equivalent alternatives. The access plans and other implementation and operation details are handled by the DBMS

engine, and are not reflected in the logical model. This contrasts with common practice for SQL DBMSs in which performance tuning often requires changes to the logical model. RDBMS are applications that are used to manage such a relational database. There are many such applications - Oracle, Sybase, PostgreSQL, MSSQL, MySQL etc. In our system we use Microsoft SQL Server for database management. The most important advantages of MSSQL server are:

1) RDBMS with high productivity

2) Developed to work with databases with complicated structure

3) Perform well in WEB environment - high speed, huge transactions

4) Full SQL compatibility - SQL is platform independent language for database manipulation.

SQL (commonly expanded to Structured Query Language) is the most popular computer language used to create, modify, retrieve and manipulate data from relational database management systems. The language has evolved beyond its original purpose to support object-relational database management systems.

5 Information protection with a digital watermark

The problem of information protection from unsanctioned access is solved already in antiquity. Even then two basic resolving directions have differentiated and now proceed to develop: cryptography and steganography [20]. Cryptography's designation is hiding a message by writing it in cipher, while in the steganography the mere fact for the existing of the secret message is hidden The word "steganography" has Greek origin, which means "secret writing" [14]. It is accomplished by several different methods and the similarity between them is that the secret message is put in a harmless, inconspicuous object. After that this object is transported openly to the address. In cryptography

the presence of a coded message itself attracts the attention. So it could be said that cryptography and steganography are not competitive information protection fields of study, on the contrary they can be used as self supplementing: one message can be encrypted and then sent by the secret steganographic methods.

Interest in digital watermarks has grown out of an increasing interest in intellectual property and copyright protection. Messages are hidden in digital data and especially multimedia: text, audio records, images, and video. A new steganography branch has appeared - digital steganography [5], [6], [7], [11], [13].

Digital steganography can be basically divided to four directions:

1) Embedding information for secret transferring;

2) Embedding a digital water mark (DWM) (watermarking);

3) Embedding identifying numbers(fingerprinting);

4) Embedding captions structure (captioning).

With embedding identifying numbers, every copy has its own number and that way, the further use of the product can be followed and the exact violator can be determined. Embedding captions is used for medical photos signing or laying an explanation legend on a map for example. The purpose is preserving of different information in a whole and stopping potential violators. The digital watermark is a special mark, imperceptibly embedded in an image, text or other signal in order to control its use. Embedding and retrieving of information from another is of basic importance in steganography and is done by the stegosystem's principles. The basic elements of a typical stegosystem for digital watermark are shown on Fig.1.

- precursory coder – structure for proper transforming of the secret message in order to embed it in the signal container (information sequence, in which the message is put);

- stegocoder – structure for embedding the secret message in other data and reading its specialties – structure for watermark retrieving;

- stegodetector – structure for stegomessage's presence determination;

- decoder – structure for secret message' s restoring.

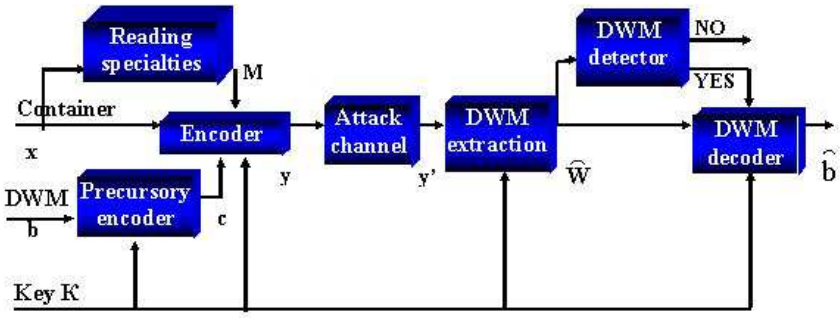

Figure 1. Stegosystem for digital watermark

Before watermark embedding appropriate transformations are necessary so that it corresponds to the container. For example, if the container is an image, then the watermark must be a two-dimensional array of bits. All transformations are done by the precursory encoder. Calculation of the general Fourier transformations for the message and the container are done in it. That enables embedding in the spectral area and that increases the stability of the watermark. An embedding key (K) is often used to increase the secrecy. Embedding and transforming messages in the container are done by the encoder.

There are different methods for that, which depend on the container's character and will be referred later. There are detectors for finding an existing watermark and for selecting it. In the first case detectors are possible with either hard or soft resolve. Metrics as Hamming distance and mutual correlation between the initial and delivered signal are used for choosing the proper resolve. When the initial signal is unknown, statistic methods are used. There are three types

of stegosystems depending on the necessary information for detection: open, half-hidden and hidden.

The hidden type I stegosystems have the best stability against outer intervention. To be stable a stegosystem must follow the next requirements:

- The system security must be completely determined by the secrecy key, so that even if the violator knows all algorithms of the system, he can not get additional information if there is a message in the container or not;
- The knowledge for the presence of a message in a container must not give the violator an opportunity to discover such in another container;
- Putting the message must not influence the container's quality;
- Possibility for discovering a secret message where it does not exist must go to zero;
- To have considerably complex algorithms for coding and decoding.

Necessary requirements for the DWM:

- The DWM must easily recover the legal owner;
- It must be steady to:
  - o common signal processing;
  - o common geometric transformations – rotation, translation, scaling and others;
  - o collusion and forgery.
- Universality – one and the same watermark for all types of multimedia;
- Unambiguous.

There are three types of stegosystems according to their steadiness to outer influence: steady, fragile and semifragile. Fragile DWM are destroyed only by insignificant modification of the container. They are used for signal authentification. The difference from the digital sign is that fragile DWM allow little modifications, which is important for the compression for example. Semifragile DWM are steady to some influences and fragile to others. As a whole all DWM can assign to this stegosystems type. For example, they can allow compression but they are fragile to elements removal.

5.1 Methods for image watermarking

5.1.1 Methods for hiding the data in the spatial area

The data is put directly into the initial image. Its basic advantage is that no additional transformations of the image are needed. DWM is embedded by manipulating the brightness or the component colors. Its basic disadvantage is the weaker steadiness. Such a method, used in the protection of the concerned folklore system, is Kutter's method [15]. Let s be a single bit to be embedded in an image $I = (R, G, B)$, and $p = (i, j)$ a pseudo-random position within I. This position depends on a secret key K, which is used as a seed to the pseudo-random number generator. The bit s is embedded by modifying the blue channel B at position p by a fraction of the luminance $L = 0,299R+0,587G+0,114B$ as: $B_{ij} = B_{ij} + (2s - 1)L_{ijq}$ where q is a constant determining the signature strength. The value q is selected such as to offer best trade-off between robustness and invisibility. In order to recover the embedded bit, a prediction of the original value of the pixel containing the information is needed. This prediction is based on a linear combination of pixel values in a neighborhood around p. The sign of the difference between the prediction and the actual value of the pixel determines the value of the embedded bit.

$$B'_{ij} = \frac{1}{4c} \sum_{k=-c}^{c} B_{i+k,j} + \sum_{k=-c}^{c} B_{i,j+k} - 2B_{ij}$$

where c is the size of the cross-shaped neighborhood. To retrieve the embedded bit the difference δ between the prediction and the actual value of the pixel is taken: $\delta = B_{ij} - B'_{ij}$. The sign of the difference δ determines the value of the embedded bit. Retrieving of the bit is made without the knowledge of an initial message. For that purpose prediction of its initial value is made according to the value of the near pixels. Accuracy can not be always guaranteed in the prediction of the secret bit value, so the embedding and retrieving functions are not mutually convertible. Some additional techniques, concerned in, are used. Also, robustness could be improved with the use of optimal error correcting codes. The method is steady to filtering, JPEG compression, geometrical transforms.

5.1.2 Spread spectrum watermarking

Based on different transformations of the container, for example Discrete Cosine Transform (DCT), it gives opportunity for greater steadiness of the DWM to transformations [7].

5.2 Method for text watermarking

5.2.1 Line-Shift Coding

This is a method, in which text lines are shifted vertically, so that the document could be uniquely coded. {-1, 1, 0} is often used as an initial alphabet, which is coded as shifting the line up, down or leaving it to its place. In most cases decoding can be made without the usage of the original document, if the constant space between the lines is known. It is easily found but steady to noise. This method is used for protection of .doc .ps in the folklore system.

5.2.2 Word-Shift Coding

This is a method of altering a document by horizontally shifting the locations of words within text lines to encode the document uniquely. The method is least visible when applied to documents with variable

spacing between adjacent words. Because of this variable spacing, decoding requires the original image – or more specifically, the spacing between words in the unencoded document.

5.2.3 Feature Coding

The image is examined for chosen text features, and those features are altered, or not altered, depending on the codeword. Decoding requires the initial document. The choice of text characteristics can be made by different criteria [6].

6 Error-correcting codes

6.1 Basics

The object of an error-correcting code is to encode the data, by adding a certain amount of redundancy to the message, so that the original message can be recovered if not too many errors have occurred [12].

Definition 1 *A* $q-$ ary code *is a given set of sequences of symbols where each symbol is chosen from a set* $F_q = \{\lambda_1, \lambda_2, \ldots, \lambda_q\}$ *of* q *distinct elements.*

The set F_q is called the alphabet and is often taken to be the set $Z_q = \{0, 1, 2, ..., q-1\}$. If q is a prime power we often take the alphabet F_q to be the finite field of order q.

Definition 2 *A binary code is a given set of sequences of 0s and 1s which are called codewords.*

Definition 3 *The (Hamming) distance between two vectors* x *and* y *of* $(F_q)^n$ *is the number of places in which they differ. It is denoted by* $d(x, y)$.

Definition 4 *Let* F_q *is the Galois field* GF(q), *where* q *is a prime power, and let* $(F_q)^n$ *is the vector space* $V(n, q)$. *A linear code* C *over* GF(q) *is a subspace of* $V(n, q)$, *for some positive integer* n.

If C is a $k-$dimensional subspace of $V(n,q)$, then it is called $(n,k,d)-$code, where n is length, k is dimension and d is the minimum distance of the code. Sometimes we denote it just (n, k) code.

Definition 5 *We call an* $(n,k,d)-$code *optimal if for fixed n, k it has the largest possible d.*

6.2 Reed - Solomon codes

Reed - Solomon (RS) codes are nonbinary cyclic codes with symbols made up of m-bit sequences, where m is any positive integer having a value greater than 2. RS(n, k) codes of m-bit symbols exist for all n and k for which

$$0 < k < n < 2^m + 2$$

where k is the number of data symbols being encoded, and n is the total number of code symbols in the encoded block.

Now let's make a more precise definition of RS codes [21]. Let α be a primitive element in GF(2^m). This means that α is an element of GF(2^m) such that each nonzero element of the field can be represented by a power of α. In these conditions for any positive integer $t \leqslant 2^m - 1$, there exists a t-symbol error-correcting RS code with symbols from GF(2^m) and the following parameters:

$$n = 2^m - 1 n - k = 2tk = 2^m - 1 - 2td = 2t + 1 = n - k + 1$$

The generating polynomial for an RS code takes the following form:

$$g(X) = g_0 + g_1 X + g_2 X^2 + \ldots + g_{2t-1} X^{2t-1} + X^{2t}$$

where $g_i \in$ GF(2^m) and $g(x)$ has $\alpha, \alpha^2, \ldots, \alpha^{2t}$ as roots.

One of the most important features of RS codes is that the minimum distance of an RS(n, k) is $n - k + 1$. Codes of this kind are called "maximum distance separable codes" (MDS). RS codes achieve the largest possible code minimum distance for any linear code with the same encoder input and output block lengths.

Also Reed-Solomon codes have an erasure-correcting capability, ρ , which is:

$$\rho = d - 1 = n - k$$

Simultaneous error-correction capability can be expressed as follows:

$$2\alpha + \gamma < d < n - k$$

where α is the number of symbol-error patterns that can be corrected and γ is the number of symbol erasure patterns that can be corrected.

There are many proposed algorithms for effective encoding and decoding of RS codes [21].

7 Structure of the application

The computer folklore system we are presenting consists of WEB interface for folklore database management. Client-server technology is used with the browser on the client computer (MS Internet Explorer has the best effectiveness), and for server Apache HTTP Server. Program technologies on the client side are ActiveX and Jscript, and on the server side – PHP scripts. The database is of relational type and is managed by RDBMS (System for database management) of MSSQL server.

The relations and the type of tables organization in the database are shown in Figure 2, described in detail in [17].

It is an open system, which is specified, changed and renovated in the process of categorization with the rising of the processed archive documents. The scheme is based on several important principles – presenting the folklore culture in its diversity and orderliness, pointing out that every culture fact is unique, ensuring efficient access and easy use of the archive funds for everyone who is interested. The classification scheme is a projection of the basic culture spheres – songs, rites, speech, music, dance, household goods – more and more independent scheme

structures are formed in these main spheres. Without the inculcation of a particular scheme categorization is impossible. Its main purpose is fast and beneficial orientation in the enormous diverse source of folklore materials. It is also a necessary step towards modern processing of the empiric data. On the scheme from Figure 2 the database tables used in the system are shown.

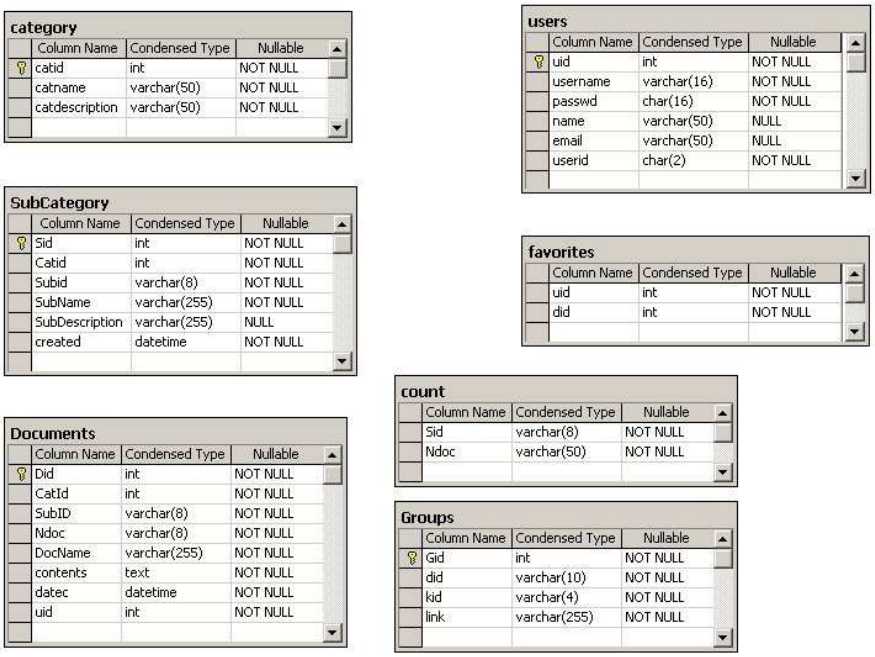

Figure 2. Database organization

- Category – information about the basic folklore categories
- SubCategory – information about the basic folklore sub-categories
- Documents – store of all folklore documents

- Users – information about all registered users
- Favorites – reference to all documents that are stored in "My Documents" category
- Count – Number of the documents in given category
- Groups – Used for documents additional grouping and external linking

The data is divided into three structural parts – audio, video and text archive and the mutual relations between the folklore parts are under review. The data is organized to allow flexibility in searching, renewing and adding.

8 Building the security of the data in the demo version of the web-based database for folklore documentation

In view of the fact that unique photo material is kept in the system, protection against illegal copying and spreading is needed. There is an opportunity for downloading partial photo materials on the client computer, but before that, it is uniquely signed, so that its origin could be proved. This is made by a specially developed ActiveX control, written in Borland Delphi and only the administrator has access to the system. It takes the graphic file as an entrance, which will be protected by the techniques described in 2.1.1, and puts a unique watermark. There is an opportunity for identifying the signed image. In 2.1.1 a primitive error correcting code based on the multiplicity of the embedding is used. It is well known that redundancy codes are far from optimality. So here we use some more effective error-correcting schemes. First we use hamming codes. They provide a mechanism that can be inexpensively implemented. In general, their use allows the correction of single bit errors per unit data, called a code word. Here we use a (15, 11) Hamming code that enables data link packets to be constructed easily by

permitting one parity byte to serve two data bytes. For error detection we use CRC. Second approach that we apply for error-correction are Read-Solomon codes [2]. RS codes are often used as "outer codes" in a system that uses a simpler "inner code". The inner code gets the error rate down and the RS code is then applied to correct the rest of the errors. Let RS(n, k) is a code over GF(2^m). Every element in this field can be represented uniquely by a binary m-tuple, called m-bit byte. To encode binary data with such a code a message of km bits is first divided into k m-bit bytes. Each m-bit byte is regarded as a symbol in GF(2^m). The k-byte message is then encoded into n-byte codeword based on the RS encoding rule. After the RS code is selected for the given case we proceed with the selection of the "inner code". According to the value of m we have selected for the RS code the same value should be selected for the dimension of the inner code. This code will correct errors on bit level in each of the m-bit bytes. Series of macros are developed; they use 2.2.1 to protect MS Word documents before they are given to the user. An additional password is also set, to prohibit him from the opportunity to change the document and that way to impede the removal of the watermark.

The data is divided into three structural parts – audio, video and text archive and the mutual relations between the folklore parts are under review. The data is organized to allow flexibility in searching, renewing and adding. For the security preventing unauthorized access to the data access levels are developed, with usernames, passwords and appropriate rights for them. We have three user roles – user, registrant and administrator. Here are the permissions that the different roles have:

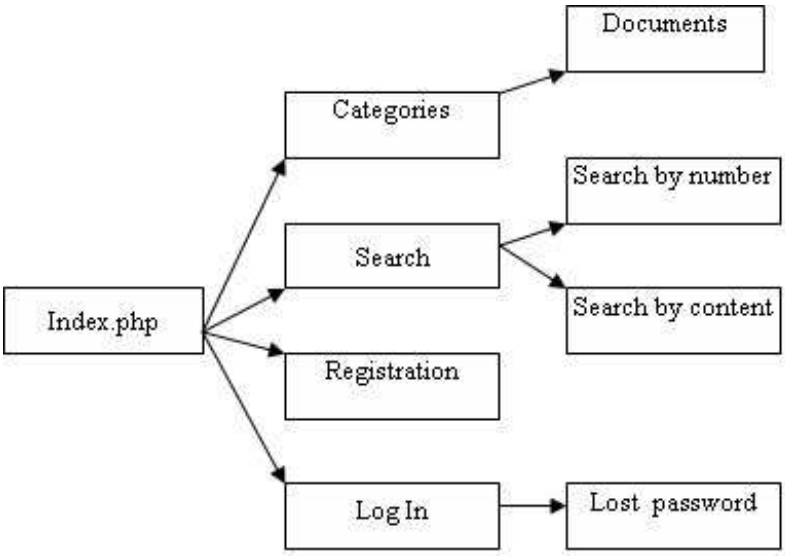

Figure 3. User permissions

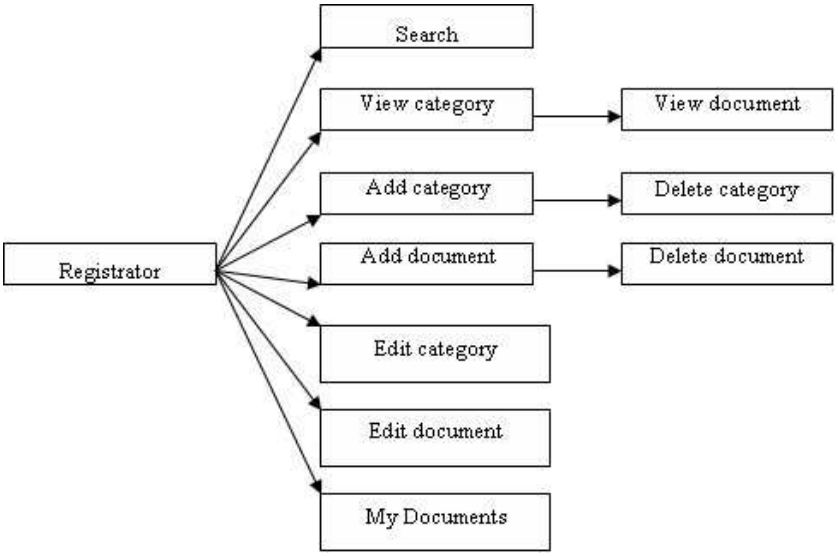

Figure 4. Registrant permissions

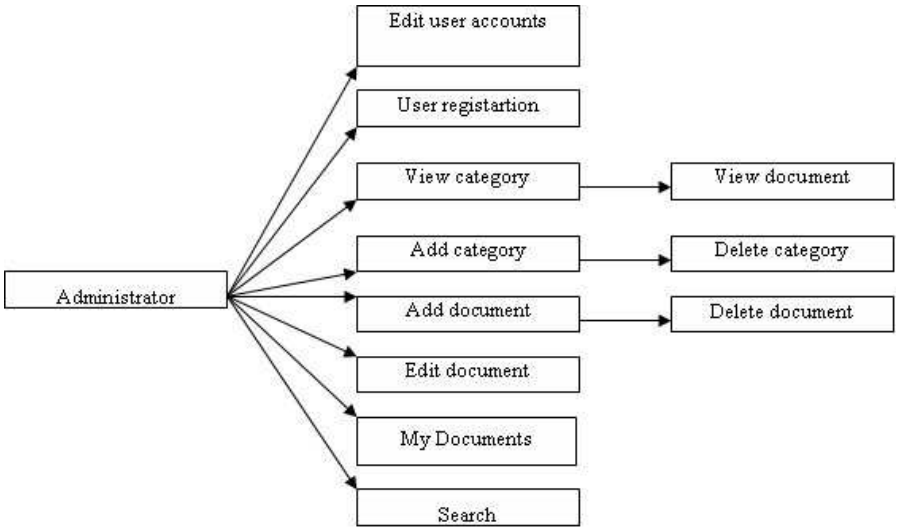

Figure 5. Administrator permissions

9 Analyzing the data of multimedia archive of folklore objects

Online Transaction Processing (or OLTP) is a class of programs that facilitate and manage transaction-oriented applications, typically for data entry and retrieval transaction processing. Some applications of OLTP include electronic banking, order processing, employee time clock systems, e-commerce, and eTrading. Online Transaction Processing has two key benefits: simplicity and efficiency. Reduced paper trails and the faster, more accurate forecasts for revenues and expenses are both examples of how OLTP makes things simpler for businesses. It also provides a concrete foundation for a stable organization because of the timely updating. OLAP is an acronym for On Line Analytical Processing. It is an approach to quickly provide the answer to analytical queries that are dimensional in nature. It is part of the broader category business intelligence, which also includes Extract transform load (ETL), relational reporting and data mining. Databases config-

ured for OLAP employ a multidimensional data model, allowing for complex analytical and ad-hoc queries with a rapid execution time.

The database in data warehouse is designed and the data is extracted from the OLTP (online transaction processing) database, transformed to match the data warehouse schema, and loaded into data warehouse database periodically by execution a batch job.

The data cube FolkloreCube is created in correspondence with the star schema of the dimensional model of the database in the data warehouse (fig. 6).

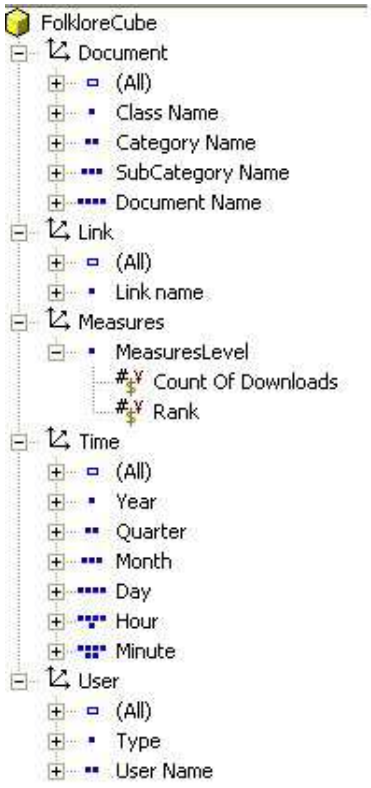

Figure 6. Folklore cube

It consists of four dimensions – Document, Link, User and Time. The first measure of the examined data cube is count of downloads of the folklore materials from the documents by the users. The users can rank the materials with the integer values between 1 and 7 that are stored in the measure rank and reflect the preferences of the users.

Applying the OLAP (Online Analytical Processing) Operations

MDX (multidimensional expressions) queries are applied to the data cube FolkloreCube providing the dimensional view of summarized data. Additional statements of MDX queries and the results from their executions are represented in [3], [8].

Discovering the Association Rules

The application for discovering the association rules in data cube FolkloreCube by using the OLAP operations is developed. An association rule shows the frequently occurring patterns (or relationships) of a set of data items in a database. An association rule is an implication of the form $X \rightarrow Y$, where $X = \{x_1, \ldots, x_m\}$ and $Y = \{y_1, \ldots, y_n\}$ are sets of items with $X \bigcap Y = \emptyset$. The rule $X \rightarrow Y$ has support s if $s\%$ of all itemsets contain $X \bigcup Y$. The rule $X \rightarrow Y$ has confidence c if $c\%$ of itemsets that contain X also contain Y. For example the following rule is generated from the data cube with daily downloads of folklore materials: {Document("*Songs*"), Time("*11/2006*")} $\rightarrow$ {Link("*somelink11*")} with support s = 0.1151 and confidence c = 0.2667. This rule means that one of the most downloaded materials from the documents in the category "*Songs*" during November 2004 is the material "*somelink11*" (with 26.67% confidence) and such downloads represent 11.51% from all downloads under study.

Discovering the Distribution Intervals of the Association Rules

In addition to the association rules the important practical applicability has their distribution in time. An algorithm that applies the OLAP operations and uses the fractal dimension to uncover the behavior of the association rules in time dimension is described. The code, which realizes the proposed algorithm, is successfully implemented.

Association rule mining aims to extract interesting correlations, fre-

quent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories [1]. We use the proposed algorithm and software implementation to explore the behavior of the database in the multimedia archive of folklore objects. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control etc., where the proposed software tools could be successfully used.

10 Comparison with existing digital archives

We make an investigation to some other folklore software systems all over the world and compare them with the one described in this paper. According to [28] and to our research of other similar systems, we can conclude that two of the main criteria for comparison are its multimedia attributes and data security.

First we make a comparison by some attributes of a system that determine it as a multimedia:

- Text
- Audio
- Graphics
- Video
- Search

Results are displayed in the Table 1.

We could conclude that there are just a few archives that meet all the criteria to be called "multimedia". Also all the archives are not very flexible in their organization and are developed just to satisfy the needs of the particular folklore organization. On the other hand the system presented in this paper has a very flexible data organization that could be easily adopted to the needs of many different folklore organizations. It has all the aspects of multimedia archives too.

Table 1. Comparison of multimedia folklore archives

System	Text	Audio	Graphics	Video	Search
[16]	+	-	+	+	+
[18]	+	+	-	+	+
[9]	+	+	+	N/A	+
[24]	+	+	+	N/A	+
[27]	+	+	+	+	+
[25]	+	-	-	-	+
[22]	+	+	+	+	-
[23]	+	+	+	+	N/A
[19]	+	+	+	+	+
[26]	+	-	-	-	-

Second part of comparison includes data security and data analysis. In fact all the systems that we research include data security just in permissions level. The newly presented system also has different user roles with different access level. But more powerful tool in it is the newly proposed watermarking scheme and the software tools for watermarking embedding. They preserve data ownership even when it's already downloaded. Also the unique for this system, that we can't find in any other system are the data analyzing tools included in it. They allow more easy and flexible research of the data in the system and to analyze working activity of the different users.

11 Conclusion

We present new approaches used in development of the demo version of a WEB based client/server system that contains an archival fund with folklore materials of the Folklore Institute at Bulgarian Academy of Sciences (BAS). We use two error-correcting schemes – one that uses Hamming codes and another that combines Reed-Solomon codes as outer code and optimal linear code as inner code. We use them as a

part of watermarking software in order to secure data in the computer system. Finally we present improved software used to analyze the database. We could conclude that this is a useful and unique computer system that has very powerful tools for data security and analysis.

References

[1] Agrawal, R., Imielinski, T., Swami, A., *Mining Association Rules between Sets of Items in Large Databases*, In Proceedings of the ACM SIGMOD International Conference on Management of Data, Washington, 1993, pages 207–216.

[2] Berger, T., Todorov, T., *Improving the watermarking process with usage of block error-correcting codes*, Serdica Journal of Computing, 2008(submitted).

[3] Bogdanova, G., Georgieva, Ts., *Finding the error-correcting functional dependency by using the fractal dimension*, In Proceedings of the the fourth international workshop on optimal codes and related topics, 2005, 20–26.

[4] Bogdanova, G., Todorov, T., Blagoev, D. and Todorova, M., *Use of Dynamic Technologies for WEB-enabled Database Management Systems*, International Conference on Computer Systems and Technologies (CompSysTech'2003), Sofia 2003, II.22-1-6.

[5] Brassil, J., Low, S., Maxemchuk, N., O'Gorman, L., *Electronic marking and identification techniques to discourage document copying*, Proceedings of IEEE INFOCOM '94, 1994 3, 1278-1287.

[6] Brassil, J., Low, S., Maxemchuk, N., O'Gorman, L., *Hiding information in document images*, Proceedings of the 29th Annual Conference on Information Sciences and Systems, 1995, 482–489.

[7] Cox, I., Kilian, J., Leighton, T., Shamoon, T., *Secure spread spectrum watermarking for multimedia*, Proceedings of the IEEE International Conference on Image Processing. Vol. 6. , 1997, 1673–1687.

[8] Demetrovic, J., Katona, G., Miklos, D., *Functional dependencies distorted by errors*, Discrete Mathematics(accepted).

[9] Fife Folklore Archives, *http://library.usu.edu/Folklo/*.

[10] Folklore Databases, *http://www.eastern.edu/library/www/webindex/arts/folklore.shtml*.

[11] Gribunin, G., Okov, I., Turincev, I., *Cifrovaia steganographia*, Solon-Press, 2002 (in Russian).

[12] Hill, R., *A first course in coding theory*, Calendar Press, Oxford, 1986.

[13] Karasev, A., *Komputernaia tainopis grafika i zvuk priobretaut podtekst*, Mir PK. - 1/97, 132–134 (in Russian).

[14] Katzenbeisser, K., Petitcolas, F., *Information hiding techniques for steganography and digital watermarking*, Artech House Books, 2000.

[15] Kutter, M., Jordan, F., Bossen, F.,*Digital signature of color images using amplitude modulation*, Proc. of the SPIE Storage and Retrieval for Image and Video Databases. V. 1997. Vol. 3022., 518–526.

[16] Living Treasures, *http://www.treasures.eubcc.bg/* .

[17] Mateeva, V., Stanoeva, I., *Klasifikacionna shema na tipologichnia katalog v Instituta za folklor*, Bulgarski folklor. kn.2-3, 2001, 96–109 (in Bulgarian).

[18] Music Multimedia Archive, *http://musicart.imbm.bas.bg/en/about.htm*.

[19] Philadelphia folklore project, *http://www.folkloreproject.org/*.

[20] Privacy Guide: Steganography, *http://www.all-nettools.com/privacy/stegano.htm*.

[21] Skallar, B., *Digital Communications: Fundamentals and Applications*, Prentice-Hall, 2001.

[22] The American Folklore Center, *http://www.loc.gov/folklife/index.html*.

[23] The Estonian Folklore Archives, *http://www.folklore.ee/rl/era/*.

[24] The Folklore Program at the University of California, Berkeley, *http://ls.berkeley.edu/dept/folklore/*.

[25] The Israel Folktale Archives (IFA), *http://www.folklore.org.il/asai.html.*

[26] The Site for American Folklore, *http://www.americanfolklore.net.*

[27] The Ukrainian Folklore Archives, *http://129.128.116.48:8890/photo_ archives/.*

[28] Velvheva, J., Petkov A., *Informacionni sistemi i tehnologii v biznesa*, Russe University press, 2002 (in Bulgarian).

[29] Welling, L., Thomson, L., *PHP and My-SQL WEB development*, Sams Publishing, 2003.

Galina Bogdanova, Todor Todorov, Tsvetanka Georgieva Received June 4, 2008

Galina Bogdanova
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Section Mathematical Foundation of Informatics
P.O.Box 323
5000 V.Tarnovo, Bulgaria
E–mail: *galina@moi.math.bas.bg*

Todor Todorov
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Section Mathematical Foundation of Informatics
P.O.Box 323
5000 V.Tarnovo, Bulgaria
E–mail: *todor@moi.math.bas.bg*

Tsvetanka Georgieva
University of Veliko Tarnovo
Department of Information Technologies
3 G. Kozarev str. 5000 Veliko Tarnovo, Bulgaria
E–mail: *cv.georgieva@uni − vt.bg*

Computer Science Journal of Moldova, vol.**16**, no.2(47), 2008

Specific features in automatic processing of the formations with prefixes*

Mircea Petic

Abstract

This article contains the information about the defining and analysis of some rules which will permit the automatic retrieving of the romanian prefixed formations and enriching the lexical resources by automatic derivation with the romanian prefixes *ne-* and *re-*. There are also described some peculiarities of the romanian prefixes.

Keywords: lexical resources, automatic prefixation, derivatives, algorithm for word extraction

Introduction

Linguistic electronic resources form the fundamental support for drawing up automatic tools for processing the linguistic information.

The need of the linguistic resources enriching is satisfied not only by word borrowing from other languages, but also by the use of some exclusively internal processes. Derivation means creation of a new word or a word with other meaning by adding some prefixes or suffixes to existent lexical bases. Prefixes are productive from the lexical point of view, when after removing the affix remains an attested word, and non-productive, when the remaining sequence after this operation is not attested as a word, though etymologically we recognize the prefix [1] in the word. The units from which the derivatives are formed, are called bases. In this article, as derivatives affixes, prefixes will be studied.

*The article is carried out as a part of the INTAS project, ref. nr. 05-104-7633

The scope of this article is the defining and analysis of some rules which will permit the automatic retrieving of the formations with prefixes and enriching the lexical resourses by automatic derivation with the romanian prefixes ***ne-*** *and* ***re-****.*

1 The classification of the prefixes

The 86 Romanian simple prefixes described in [2] can generate 5680 derivatives. These derivatives can be classified in four fundamental categories, such as:

- inherited derivatives from Latin, for example, (a) *închide* (in engl. (to) *close*), (a) *deschide* (in engl. (to) *open*), (a) *rămîne* (in engl. (to) *remain*);
- borrowing from other languages, for example, *deservire* (in engl. *service*), *nonsens* (in engl. *nonsense*), *prefabricat* (in engl. *prefab*);
- imitations by foreign models, for example, *concetăţean* (in engl. *fellow citizen*), (a) *întrevedea* (in engl. (to) *discern*);
- Romanian internal creations, for example, (a) *dezrobi* (in engl. (to) *emancipate*), *nefericit* (in engl. *unhappy*), (a) *înţărca* (in engl. (to) *wean*).

The compound prefixes obtained from two simple prefixes as a unique element of derivation are: *apar-*, *metem-* and *meten-*, *ram-* and *ran-*, *rim-* and *rin-*, *sco-*, *sper-*. It is worth mentioning that compound prefixes, analysed together with the wordbase and not relating to a formation prefixed before with one of the prefixes, we do not confuse with a double successive prefixation, for example, in *pre/străne*pot (in engl. *the son or daughter of the / great-grandson*), or (a) *re/înscrie* (in engl. (to) *re/write down*).

Developed prefixes, that result from combination of a simple prefix with a non-prefixal element, an insignificant fragment of a prefix or

a root, are: *destr-* (destrauri), *pres-* (presfira), *zăs-* (zăstimp, in engl. *period*). Sometimes it is difficult to distinguish a compound prefix from a developed one. Thus, a complex prefix *năs-*, based on a simple prefix *nă-*, may be considered compound with simple prefix *s-* (*z-*) in formations such as *năzbate* and developed in *năzduh* or *năzvăţa* (in engl. to *spoil*), where *z-* is recognized because of false analysis of the word *văzduh* (in engl. *air*) or (a) *dezvăţa* (in engl. (to) *unlearn*) [1].

As to the prefix's etymology, 12 prefixes are inherited from Latin (*des-*, *în-*, *stră-*), 13 – from Slavic (*ne-*, *răs-*), 18 came from ancient Greek, being taken by a Slavic or Latino-Romanic intermediate (*anti-*, *arhi-*, *hiper-*, *hipo-*) [3]. 29 prefixes with a multiple etymology are neologic inherited from Latin, French and/or Italian (*ante-*, *circum-*, *co(n)-*, *contra-*, *ex-*, *extra-*, *non-*, *post-*, *re-*, *ultra-*).

2 Some quantitative characteristics of the prefixes

Traditionally prefixes are described by a limited set of features: etymology, derivational model (with pointing the part of speech of the obtaining word and of its bases, for example, the verb (a) *dezdoi* (in engl. (to) *unbend*) is constructed from the numeral *doi* (in engl. *two*) and the prefix *dez-*) and the meaning of the obtaining word. Step by step derivational models began to be characterized by a set of qualitative attributives, such as: utility, regularity, productivity, frequency, etc. There aren't clear definitions of those notions. That is why we can only suppose the real meaning of those terms. Only formulating a set of quantitative definitions allow us to obtain some of the quantitative characteristics for the derivational affixes [4].

By number of the registered derivatives in [2], the most productive prefixes are *în-* (their number is 571), *ne-* (449), *des-* (395), *re-* (338).

In [5] it had been established some quantitative characteristics for these romanian prefixes, namely: number of words which begin with these prefixes, number of the derivatives with these prefixes, the repartition of letters which follow after the mentioned prefixes, the distri-

bution of the parts of speech (noun, adjective, verb) for every prefix mentioned above.

The romanian prefix *în-* can be attested in four different phonological forms, namely *in-*, *im-*, *în-* and *îm-*. One of them is a phonetic variant *îm-* before *b*, *p* consonant, for example, (a) *îmbătrîni* (in engl. (to) *grow old*), (a) *împărţi* (in engl. (to) *divide*), etc. Sometimes it is possible that the romanian prefixes *în-* and *îm-* are attested as the prefixes *in-* and *im-*. The romanian prefix *ne-* has no other literary phonetic variants. The prefix *des-* has the following literary phonetic variants: *dez-*, before the roots that begin with a (semi)-vowel or a voiced consonant and *de-* obvious before the roots that begin with *s*, *ş*. The prefix *re-* is a neologic prefix and sometimes substitues the ancient variant *ră-*. As a source for establishing these quantitative characteristics served [5], that contains about 30 thousands words divided in flection groups depending on their forming way. These words are grouped by part of speech, but the noun is classified also into gender. It permits to establish the announced quantitative characteristics mentioned above. These characteristics has been obtained by some programs developed in C programming language.

Taking into account all phonological forms of prefix *în-* for every form (*in-*, *im-*, *în-* and *îm-*), some specific quantitative characteristics have been found. Thus, it has been calculated the number of words that began with the particles *in-*, *im-*, *în-* and *îm-*. The total number of these words is 1145 [5].

The same situation is also with the prefix *re-* which has another phonological form *ră-*, but it is an old one. For the prefix *des-* it has been calculated separately the quantitative characteristics for phonologic forms *des-* and *dez-*. As the romanian prefix *ne-* has no phonological variants, in the calculations only this particle has been considered. The details of quantitative characteristics for these prefixes are presented in Table 1.

Taking into consideration the information obtained and arranged in Table 1 we can say that there are four letters, namely *ă*, *k*, *q* and *y*, before which the particles mentioned above have not been found. In addition, the words in which the presence of the prefix is recognized

Table 1. The quantitative characteristics for the phonetic variants of the studied prefixes [5]

pre-fixes	pho-netic vari-ants	num-ber of words	num-ber of deriva-tives	%	part of speech	total let-ters	conso-nants	vowels
in	în	583	107	18	V,N,A	21	17	4
	in	360	47	13	N,V,A	14	9	5
	im	126	15	12	N,V,A	7	2	5
	îm	79	19	24	V,N,A	2	2	0
ne		216	36	17	N,A,V	19	14	5
des	des	191	39	20	V,N,A	13	8	5
	dez	101	48	47	V,N,A	12	7	5
re		454	81	18	N,V,A	20	15	5
average				21				

where V denotes the verb, N – the noun and A – the adjective.

by relating it to a wordbase existent in Romanian language represent about a fifth part of the words that began with these particles. In general, more often we find consonant after the studied particles. The words that begin with the mentioned particles more often are nouns as a part of speech, then come the words that are verbs and finally adjectives. Actually, some prefixes contain more phonological forms and it makes more difficult to establish some quantitative characteristics.

3 Retrieving analysable formations with prefixes from a lexicon

For all categories of formations only the words in which the prefix presence was distinguished by referring to a wordbase existent in the Romanian language had been taken into considerations [2]. They can be grouped in derivatives of the type: *analyzable*, *semianalyzable*, and

non-analyzable. In the analyzable formations both are distinguished either the prefix or the wordbase. For example, in the word *poimîine* (in engl. *the day after tomorrow*), *poi-* is the prefix, and *mîine* (in engl. *tomorrow*) is an adverb (wordbase).

As a source for analyzable formations with simple prefix extracting a lexicon[1] had served which contains not only graphic representations of the words, but also its part of speech. This lexicon contains reusable resources of Romanian language (it has approximately 100000 of wordbases). It needs to be mentioned that a word can have several entrances for different parts of speech [7].

Besides the lexicon it was used a list of 86 simple prefixes that are registered in [2]. Also phonological forms of those prefixes had been added. Being important the peculiarity of those prefixes and derivatives described above the algorithm for automatic extraction of the analysable formations with simple prefixes had been developed. Lately it was implemented in a program written in C programming language.

Taking into account the peculiarity of the Romanian simple prefixes and of the available lexicon a simple but distinctively used for automatic extraction of the analysable formations with simple prefixes algorithm had been elaborated. The essence of the algorithm is the following: it reads a prefix and when listing the words from a lexicon, chooses those that can be integrated in the established constrains, namely if after removing the prefix it remains an attested in the dictionary word and its part of speech coincides with the initial one. Corresponding to this algorithm, it could be added also the instructions for counting both the analysable formations and of the words which begin with the particles that coincide with the respective prefixes [8].

Using the program, the analyzable formations with simple prefixes had been found and extracted for all prefixes mentioned in [2]. In Table 2 the concrete numbers of the analyzable formations for the ten most numerous prefixes obtained with described program comparatively with those in the source [2] are given. The results presented in this table are expectable for some prefixes, but for the other are really amazing.

[1]The lexicon is available on the site http://imi201.math.md/elrr/

Table 2. The number of analysable formations (NAF)

prefixes	NAF program	NAF source [2]
ne-	1500	449
re-	970	338
de-	702	250
in-	632	217
a-	610	276
s-	474	73
pre-	441	186
co-	338	19
con-	282	106
anti-	257	281

Verifying the analyzable formations with the prefixes *ne-* and *re-* we can say that the precision of a good word selection is very high, the cases of the word of the kind *rece* (in engl. *chill*) are singular. The same can be said also about the prefixes *pre-*, *con-* and *anti-*.

For the other prefixes many words had been obtained that were not analyzable formations with these prefixes, for example for the prefix *s-*: *scară* (in engl. *ladder*), *seră* (in engl. *hothouse*), and so on. But the number of analyzable formations with the prefix *anti-* is not greater that those registered in [2]. So it can be presumed that some of the words are not present in any of lexicographic sources.

In order to retrieve analysable formations with compound prefixes it was used the same algorithm described in [2] and a program in C programming language has been developed on its base.

The concrete data obtained by the mentioned above program for 10 compound prefixes, comparatively with those presented in [2], are given in Table 3. In this table we can bring into evidence the big number of analysable formations obtained by program comparatively to the number recorded in [2] for the compound prefixes *ra-* and *sco-*. It can be explained by confusing the prefixation with double successive

Table 3. The number of analysable formations (NAF)

the prefix	NAF program	NAF source [2]
ra-	64	2
sco-	22	2
ram-	4	1
apar-	4	1
ran-	3	1
rin-	3	1
sper-	2	1
metem-	1	1
meten-	1	1
rim-	0	1

prefixation with simple prefixes *a-* and respectively *co-* after that with *re-* (*r-*) and, respectively, *s-*. In other cases the expectable results were obtained, except the prefix *rim-* for which we didn't find the analysable formation *rimbomba*, which isn't present in the lexicon.

The obtained results suggest the idea that the algorithm for automatic extraction of the analysable formations with simple prefixes is not universal. It can be used in automatic extraction of the analyzable formations with long simple prefixes. For short simple prefixes the information about only graphic representation and part of speech proved to be insufficient for several simple prefixes. They need, probably, the semantic information about analyzable formations with such prefixes.

Although, we certainly can say that for long simple prefixes, the presented algorithm satisfies with a high precision the automatic extraction of the analyzable formations with simple prefixes. The results proved that the most numerous simple prefixes in analyzable formations registered in [2] coincide with the results of the program [9].

Nevertheless, the algorithm mentioned above will not help us to find the derivatives like: *deschis* (in engl. *opened*), *închis* (in engl. *closed*), (a) *combina* (in engl. (to) *combine*) and (a) *îmbina* (in engl. (to)

join), *interbelic* (in engl. *interwar*) and *antebelic* (in engl. *prewar*). These words aren't analysable formations. They are semi-analysable formations. The prefix derivatives of this type are the words in which we can distinguish only the prefix as opposed to the other prefixed formations or compound words with a common root, inexistent as an independent word. So the method by which we recognize the analysable formations is not valid for semianalysable formations.

4 Automatic prefixation

According to quantitative characteristics discussed in section 3, the more frequent ones are *în-*, *ne-*, *des-*, *re-*. As the prefixes *în-* and *des-* have more phonetic variants and need more information in order to formulate prefixation rules, afterwards we will consider only the affixation by the romanian prefixes *ne-* and *re-*.

4.1 The prefix *ne-*

According to [2] in The Dictionary of the Romanian language (DRL) there are registered 449 analysable formations, considering only one formation from a group (though its number is greater), having a common base: for example, *nemişcare* (in engl. *immobility*), but not *nemişcat* (in engl. *motionless*), *nemişcător* (in engl. *motionless*). Although, DRL has only a part of all existent formations with the prefix *ne-*, the number of derivatives from the DRL is small and it does not illustrate the real productivity of this romanian prefix (as a rule, the dictionary does not include obvious formations). Since the derivation with *ne-* forms an open system, it does not contain the formations from other sources.

It is important to note the preference of this prefix to the adjectives of participle origin (in our case that ends in: *-at*, *-it*, *-ut*, *-ît*, because the others face more difficulties in participle recognition), and also those derivatives with the suffixes *-tor*, *-bil*, *-os* in relation with verbal bases. In DRL, the formations from the adjectives derivated by other suffixes are, as a rule, fortuitous. The creations with *ne-* from the primary

adjective are, in general, not numerous and unusual.

The formations with *ne-* from the verbs are not numerous. The derivations from verbs, being not specific for the prefix *ne-*, can be from adjectives of participle origin or some derivatives with suffixes. The prefix *ne-* is grammaticalized in relation with gerund, participle and supine of the verbs, replacing completely the negation *nu* (engl. *not*).

The derivation with romanian prefix *ne-* constitues an open system. Wordbases to which this prefix can be attached are of various origin, where the prefix *ne-* can be attached, valid in literary language in general, it does not function in popular and familiar ones, where in spontaneous constructions, the prefix *ne-* can deny any kind of word. The prefix *ne-* is productive in all literary styles of the language.

Thus, from those mentioned above we can infer the following rules for the prefix *ne-*:

- from the adjectives derivated by suffixes *-tor*, *-bil*, *-os* we form adjectives derivated by the prefix *ne-*;
- from the participle ended with *-at*, *-it*, *-ut*, *-ît* we form adjectives derivated by prefix *ne-*;
- from the gerund we form adjectives derivated by prefix *ne-*.

4.2 The prefix *re-*

According to [2] in DRL the number of analysable derivatives with *re-* is 283. To these words we can add 55 derivatives from other sources (for example: Dictionary of neologisms, Morphological Dictionary, Actual Romanian Language, etc), the most numerous derivatives are those verbal obtained from verbs (analysable formations). Many of semi-analysable formations with the romanian prefix *re-* is related with those semianalysable derivatives with romanian prefixes *în-* (*in-*), *con-*.

The prefix *re-* is not attached, as a rule, to the words which begin with r+vowel [2]. In fact, there are some derivatives of this case: *reromanizare* (in engl. *reromanization*), *rerafinare* (in engl. *repurifying*), *reruralizare* (in engl. *reruralization*).

The wordbases at which the prefix *re-* is attached are of different origin, both ancient (latine, slave, hungarian, turkish, greek), and modern (neologic).

Actually, the romanian prefix *re-* can be attached to any verbal wordbase. Another observation is that the nouns formed from the present infinitive of the verb by adding the suffix *-re* form derivative by prefix *re-*. The derivatives with this prefix can be found in all literary styles of the Romanian language.

So, we can define the following rules for the romanian prefix *re-* from the infinitive of the verbs we form:

- verbs derivatives with prefix *re-*;
- nouns derivatives both with prefix *re-* and the suffix *-re*.

4.3 The methodology of automatic prefixation

The rules formulated above need knowledge only about graphic representation of a word and its part of speech. The source [6] contains 28932 words which are divided into flection groups depending on its way of forming. Dividing into a set of other flection groups, as it was observed from the rules above, does not influence the process of forming the derivatives with the romanian prefixes *ne-* and *re-*. It is important to mention that the obtained derivatives based on formulated rules above will inherit the flection group of the wordbase of the derivative, except only the case of noun forming from the verb.

Examining the words from the lexicon and concatenating the prefix respectively to those ones which correspond to the categories established by the rules above, an algorithm of analysable derivation with romanian prefixes *ne-* and *re-* is implemented in a unit capable to generate new words.

4.4 The results

The developed program based on the algorithm mentioned above enriched the lexicon with 417 derivatives with the romanian prefix *ne-*

and 9014 derivatives with the prefix *re-*. In Table 4 there are the results obtained for the analysable derivatives rules with the romanian prefixes *ne-* and *re-* for adjectives and verbs. Taking into consideration that the lexicon had already 216 words that began with *ne-*, the growth is about 1,9 times. In the same way the number of the words that begin with *re-* is 454, so the growth is 19 times. As a result, the growth of derivatives is 9430 words with the romanian prefixes *ne-* and *re-*, that represents about 1/3 of the initial lexicon. In the process of flection of these words with the help of the programs described in [10] we will obtain 250340 of flected forms.

Table 4. The results obtained for analysable derivation rules with romanian prefixes *ne-* and *re-* for adjectives and verbs

prefixes	part of speech	suffixes	nr. of derivatives
ne-	Adjective	-tor	72
		-os	46
		-bil	20
	Participle	-at, -it, -ut, -ît	261
	Gerund	-ind	17
re-	Verb		4507
	Nouns from verb	-re	4507
Total number of wordbases with prefixes *ne-* and *re-*			9430

Thouth the statistics presented in Table 4 is important, there are some moments that can be brought into the evidence. The number of the derivatives obtained by the romanian prefixes *ne-* and *re-* includes also those that are already contained in [5] and in DEX.

So, there is one derivative prefixed by *ne-* that is already included in [5] and 20 that are in DEX. In this case the number of true new words automatically generated is 397, that represents 95,43% of the initially generated words.

Analogous, the number of already existent words with romanian

prefix *re-* in [5] among those automatically generated is 27, and in DEX 298. Thus, the number of really new words is 8698, that constitutes 96,49% from those automatically generated by the romanian prefix *re-*.

If we will exclude the words that had already been prefixed with the affixes *ne-* and *re-* the number of the remained derivates will constitute 391 and, respectivelly, 8504. Although among the remained words there are also the ambigous ones. There are 362 perfectly valid words with prefix *ne-* and 7834 with prefix *re-*, that represent 91.18% and respectivelly 92.12% of new automatically generated words.

5 Conclusions

As only two romanian prefixes *ne-* and *re-* have been studied in new words generation, it proved their good productive properties in automatic affix derivation. In addition, those derivational rules, except the fact that the information about the part of speech and graphic representation is needed, require also the semantic and etymologic information of the wordbases to which they are attached. It was ascertained that in the existent lexicon it is difficult to distinguish the derivatives and the words that begin with the same sequence of letters of the correspondent prefix. In general, the derivatives generated by the correspondent rules would inherit the flection group from the derivatives wordbase, except the case when the verb is transformed in the noun.

References

[1] Iorgu Iordan. *Limba română contemporană*. Bucureşti, Editura Academiei, 1970.

[2] Formarea cuvintelor în limba română. (vol. Colectiv), Bucureşti, Editura Academiei, 1970.

[3] Al. Graur, Mioara Avram (redactori responsabili). *Formarea cuvintelor în limba română*, vol. II, Bucureşti, Editura Academiei, 1978.

[4] B.I. Bartkov, T.B. Bartkov, E.A. Golovatskaia, L.K. Karashchuk. *Qualitative derivatology of contemporary english prefixes of negation.* http://old.festu.ru/ru/structure/library/library/vologdin/v2000-I/173.htm (in russian)

[5] M. Petic. *Unele caracteristici cantitative ale celor mai productive prefixe în limba română.* In Proceedings of the Vth International Conference on Microelectronics and Computer Science, Chişinău, september 2007. pp. 161–164.

[6] A. Lombard, C. Gâdei. *Dictionnaire morphologique de la langue roumain.* Bucureşti, Editura Academiei, 1981, 232 p.

[7] Ciubotaru C, Cojocaru S.,.Boian E., Colesnicov A., Malahova L., Demidova V., Burlaca O. *Resurse Lingvistice Reutilizabile.* În Lucrările atelierului Resurse lingvistice şi instrumente pentru prelucrarea limbii Române. Iaşi, noiembrie 2006, pp. 75–79.

[8] M. Petic. *Automatic extraction of the analysable formations with simple prefixes.* In Proceedings of the Second International Conference of Young Scientists Computer Science and Engineering-2007, Lvov, october 2007, pp. 215–217.

[9] M. Petic. *Prefixe compuse în formaţiile analizabile.* In International Conference of Young Researchers. Scientific Abstracts, Chişinău, november 2007, pp. 215.

[10] E. Boian, S. Cojocaru. *The inflexion regularities for the Romanian Language.* Computer Science Journal of Moldova, 4, 1(7), 1995, pp. 40–58.

M.Petic, Received June 12, 2008

M.Petic
Institute of Mathematics and Computer Science,
Academy of Sciences of Moldova
5, Academiei str., Chisinau,
MD 2028, Moldova
E–mail: *mirsha@math.md*

Computer Science Journal of Moldova, vol.**16**, no.2(47), 2008

The representation method of a complex software system dynamic project

Nicolae Magariu

Abstract

The system of transition diagrams having the capacity to define the dynamic links between components of complex software system presented at different levels of specification is proposed. The model of complex software development based on the applying of transition diagrams systems is considered. The advantages of the model are analyzed.

Key words: software design, transition diagram, dynamic project of software system.

1 The problem

The necessity of intensification of human society economical development on the one hand, and advanced performances of modern computers on the other hand, force automation of the most complex Information Handling Processes (IHP) in the field of economics. This automation can be realized by means of Complex Information System (CIS) [1]. CIS development needs the big intellectual efforts of a group of developers in the course of several years. In the context of CIS development a special attention is given to CIS design phase, especially to CIS dynamic aspects design: components and subcomponents behavior specification, correlation of system's static and physic aspects with its dynamic aspects, etc. One of the well known means, which is applied to specify dynamic aspects of simple programs, is the transition diagram [2, 14]. An attempt to specify dynamic aspects of a CIS with the help of single transition diagram causes serious difficulties. Applying

a set of transition diagrams to specify CIS dynamic aspects can be a good solution of the posed problem. A model of applying the Systems of Transition Diagrams (STD) in CIS dynamic projects elaboration is proposed.

2 Preliminary

STD had been used for the first time in 1963 by American scientist Melvin E. Conway when he had developed a separable compiler [4]. Conway had specified the syntax of COBOL language utilizing STD, and had proposed an original algorithm of compilation which he had named diagrammer. A diagram of the STD has a double destination. It defines the syntax of language construction on the one hand and specifies the model of abstract automata for this construction implementation on the other hand.

As automata model a diagram consists of nodes and edges, where nodes represent states and edges define transitions from one state to another [2,5, 14]. A diagram can have one initial node (or state), one or more final nodes and no one or several intermediate nodes. An initial or intermediate state defines the opportunity to respond to defined events occurrence. A final state defines the passing interruption. The transition from one state to other is pointed by an edge. Edges can be marked by terminal symbols of the language or by name of a diagram. This method of interlinking between diagrams characterizes a Conway STD. A program unit can be associated with every edge. This unit is executed when the corresponding edge is passed. The STD includes the main diagram -"program" and several diagrams for presentation of all syntactic constructions of the language. The diagrammer is to pass full main diagram path having read the symbols of the program which are saved in input line. It passes the full path by steps. One step includes: - reading of one symbol from the input line and selection of the edge for transition to the other state; - execution of the program unit associated with the selected edge; - transition to the other state pointed by the selected edge. It finishes passing when the current state is a final state or the transition from current state isn't possible. It

starts the input line analysis from the initial state of the main diagram and recognizes program constructions in nondeterministic way - with returns in input line. Conway determines the possibility of STD and diagrammer application for other languages compilation as well. The STD can be constructed by using the formal description of the language syntax.

David Bruce Lomet had studied the Conway diagrammer and had proved diagrammer's equivalence to a restricted Deterministic Push-Down Acceptor (DPDA) called a nested DPDA [5]. He had established that the class of nested DPDA's is capable of accepting all deterministic context-free languages.

The author has been studied Conway's SDT and diagrammer independently of D. B. Lomet. There was elaborated a special class of STD which allows the diagrammer functions in deterministic way. This class of STD was applied in time of the APL interpreter construction [6].

APL is an interactive language and the syntax of its constructions is very simple [7]. User variables can obtain as a value a numerical array or an array of *char* type with arbitrary number of dimensions. APL operators can be nullary, unary or binary. Operands of the operators can be some expressions, evaluated to arrays, with arbitrary number of dimensions. Semantics of the APL operators is very consistent and it can be described in C language using tens or hundreds of instructions. Almost all operators in APL have equal priority (with small exceptions). The user can create and give to APL system an expression for execution or can create a User Defined Functions (UDF). A UDF prototype has the structure similar to the APL basic operator's structure. A UDF can be made up from expression instructions or branching instructions. Jumps can be made inside UDF only. A UDF can't be defined inside another UDF. All UDFs necessary for solving a problem or a family of problems are stored in a special Work Space (WS). An expression can contain invocations of UDF. To solve a problem by means of APL system, one needs to create the set of UDF in WS and give the APL expression to APL system. The execution of this expression leads to execution of UDFs, which are invoked directly or indirectly from this expression. So, the STD of the APL expression

defines the order of execution of a UDF from WS. One can say that the APL expression defines a control message to realize functionality, which is necessary for a user and the STD of APL expression defines the rules of execution of this expression. The diagrammer interprets the APL expression according to the rules defined by STD.

This context suggests the idea about using the STD in designing and construction of CIS [8]. In this case the process of data handling is considered as the execution of an actions sequence. One event provokes an action or an activity of data handling. A user function can be defined as a sequence of events. The STD defines the correctness of events sequence and the action of data handling which corresponds to each event of the sequence. Thus the STD represents the dynamical project of a CIS. An input line contains a sequence of events names – specification of the functional requirement of the user. The execution of actions of data handling which corresponds to this sequence of events has to give the user result. The diagrammer represents the model of interpretation control of actions corresponding to events pointed in an input line.

Ed Yourdon had researched a software system modeling by means of Transition Diagrams Systems of States (TDSS) [3]. He elaborated the method of creation of TDSS in the context of modeling the software systems oriented to data flaw. TDSS are the other form of dynamic project representation. The events and the program units aren't associated to edges and the links between diagrams aren't pointed evidently in TDSS.

3 A transition diagrams and STD as a dynamic projects

Two types of diagrams are used in automaton modeling: transition diagrams and state diagrams or machine [9]. Transition diagrams are used in modeling of G.H. Meally automaton and state diagrams are used in modeling of E.F Moore automaton.

The diagrams application in designing of automaton for different

objects control (for example, flying apparatus, seagoing ships, etc.) is studded in a number of publications [10,11]. The elaboration of such automatons begins with the specification of the object and their states. The object is described by means of finite set of variables. Every variable represents a property of the object. The value of each variable can be changed in time of object activity. An object state is specified by concrete values of a subset of variables. After this the automata model construction follows. This model is represented by means of one or several diagrams. For every state of the diagram there are specified: - the signals (or events names) which can be received from the object; - the procedure of control signals generation for every received event name; - the states in which the automata can be passed. If several diagrams are used under automata model construction then the association of these diagrams is defined. The method of diagram association is defined in particular way proceeding from feature of the concrete problem. As a rule such modeling is realized by means of state diagrams.

On the base of automata model the functional automata schema represented by means of logical operations or the program which describes the automata behavior is constructed.

The method of program construction on the base of the automata diagram is studied in the works [11,12]. In specification of such programs the separation of states describing from control process describing is proposed.

On the base of accumulated experience in construction of program for objects control the original style of programming was developed – Automaton Oriented Programming (AOP) [11,12]. In AOP the models of data processing are represented as a model of automata and the program unit describes the work of finite state automata. The main features of the AOP are:

1. The model of data processing is specified as a model of automata represented as a transition matrix, transition diagram or state diagram. States of the automata are defined depending on the values of subset of variables from the set of variables which is de-

fined when problem analyzing. Thus the states of automata differ one from another through different values of the variables which describe the problem. In the time of automata step execution some variables obtain new values. In modeling of composed data processes several diagram may be used. The method of diagram association is specified in particular way depending on problem peculiarities.

2. The program unit contains the loop. The body of this loop represents the program code fragment which describes a work step of the automata. This program code fragment can be represented as a subprogram – function. The program execution is driven by events produced by external or internal actors. AOP can bee realized by means of structured or object oriented programming.

The methods of transition diagram application in designing of languages processors is described in many publications [2,4,14]. The methodology of interpreter modeling by means of transition diagrams is a good base for development of methodology of CIS dynamic project construction.

The methodology of AOP originates a new programming paradigm – Event-Driven Programming (EDP) [15-17]. The general static model of event-driven application is shown in Fig 1.

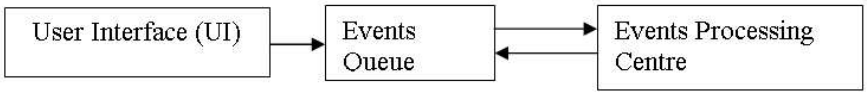

Figure 1. The general static model of event-driven application

The UI component offers to user the means of event generation: windows with menus, buttons, etc. The event generation is realized by the user through different ways: mouse clicking a menu line, a button,

etc. The ,,Events Queue" component ensures the events saving in a queue. The ,,Events Processing Centre" component takes the events from the queue and processes it. The model of events processing is specified by a transition diagram and the interpretation of this component functions is driven by this diagram. The events can be processed in parallel way [16].

Currently the EDP is developed by means of OOP [16]. The author used this technique in elaboration of the "Electronic Atlas of Moldova Republic" application prototype [18]. But EDP can be developed by means of structured programming and STD [19]. See compartment 5.

4 Deterministic STD

From theoretical point of view STD can be considered as the association of the transition diagrams realized by means of the standard method of association. Therefore the STD structure can be modified easyly.

From practical point of view it is very important that the STD should provide a deterministic transition from one to another state.

A TD of STD uses two disjoint vocabularies: vocabulary of events names and vocabulary of diagrams names. The set of events can be defined from specification of functional requirements to the application. This set is finite always. One can define the events sequence for each functional requirement. The detailed analysis of the construction method of STD is not provided in the context of this paper. Two general models of STD construction may be pointed out.

Model 1 includes the following common steps:

St1) The functional requirements are defined as sequences of events names.

St2) The transition diagrams of STD are constructed on the basis of sequences of events names having the common prefixes or/and suffixes.

Model 2 provides for the elaboration of the events set and the STD in the same time as the top-down functional designing of a CIS.

As a rule the STD constructed by applying any method is a non-deterministic one. In this case it is necessary to check STD and if it is not deterministic then transform it into deterministic one. To define

the STD which models the deterministic process, the formal notations were proposed [6]:

Σ – vocabulary of events names;

N – vocabulary of diagrams names;

d_i – the name of i-th diagram of the STD ($i \in 0...n-1$) and $d_i \in N$;

IL – input line;

Σ_{ij} – the set of events names, which mark edges going from j-th node of i-th diagram;

N_{ij} – the set of diagram names, which mark edges going from j-th node of i-th diagram;

$F(d_i)$ – the set of simple events which are the first events of the sequences that can be interpreted under passing d_i diagram ($i = 0...n-1$). The construction method of $F(d_i)$ set ($i = 0...n-1$) is:

$$F(di) = \Sigma_{i0} \cup (F(d_l)), \text{ for all } \ d_l \in N_{i0}$$

These notations helped to define exactly the limitations for diagrams structure which ensure the deterministic work of the diagrammer. The limitations for diagrams structure are:

1) $N \neq \emptyset$;

2) Σ and N vocabularies don't contain useless symbols;

3) STD diagrams don't contain unmarked edges;

4) There can't be two edges, outgoing from the same node and marked with the same symbol;

5) For every d_l diagram ($d_l \in N_{ij}$) the intersection of Σ_{ij} and $F(d_l)$ is the empty set;

6) For 2 diagrams d_i and d_l ($d_i, d_l \in N_{ij}$) the intersection $F(d_i) \cap F(d_l)$ is the empty set.

The sets $F(d_i)$ (i =0...n-1) can be calculated for concrete STD before the diagrammer starts execution.

It was proved, that if STD diagrams satisfy enumerated limitations, then in every state and current event the diagrammer can choose only one possible transition [6].

On the base of the diagrammer constructed in [6] the deterministic model of events sequence interpretation was elaborated. It was

named AIM (Application Interpretation Model). The general description of the AIM was made up by means of pseudo language based on C language. See Appendix 1.

The knowledge about limitations for diagrams structure and AIM simplify the process of STD construction.

5 Applying of the STD in the development of software systems

The CIS's dynamic project representation by means of STD correlates very well with its top-down designing. Top-down design methodology of the complex program systems recommends initially to develop a general structure of a system - its components (subsystems), and after that it recommends to do detailed modeling of every subsystem. In this case a dynamic project can be represented by a STD, which contains a main diagram named "PS". This diagram specifies the behavior of program system's subsystems. The STD also must contain at least one diagram for every component. These diagrams specify the behavior of subcomponents of the appropriate components.

Let us consider an example of a generalized project of complex software system, which consists of subsystems $c1$, $c2$, $c3$, $c4$, $c5$. Let these subsystems realize the following functionalities, necessary to user: $c1$ – data initializations and adaptations, $c2$ – calculations according to the method M1, $c3$ – calculations according to the method M2, $c4$ – comparative analysis of the obtained results, $c5$ – report generation. Suppose that $c1$, $c2$ and $c3$ components need an intensive interaction with the user. In this case, the behavior of the systems may be specified by the main diagram "PS", shown in Fig. 2.

Symbols "another problem", "repeat" and "exit" are the names of the simple events and symbols $c1$, $c2$, $c3$, $c4$, $c5$ are the names of diagrams which specify the components behavior. These names may be considered as names of composed events. To simplify understanding, the names of the program units associated with events are not shown on the diagram edges.

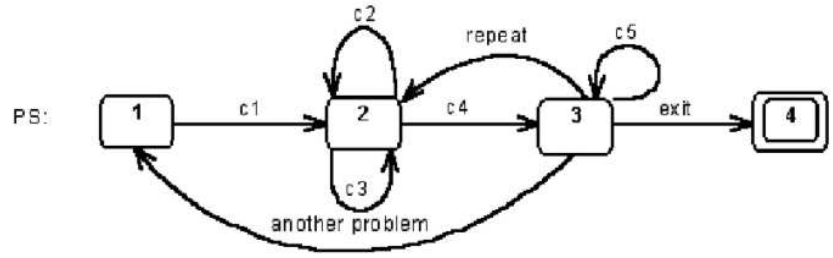

Figure 2. The main diagram of the SDT

Suppose that the diagram, which corresponds to component $c1$, has the structure, shown in Fig.3. The symbols $e0$, $e1$, $e2$, ..., eN are the names of simple events in this diagram. The $F(c1)$ set includes **e0** symbol only.

The transition diagrams for components $c2$, $c3$, $c4$, $c5$ of the CIS are constructed similarly.

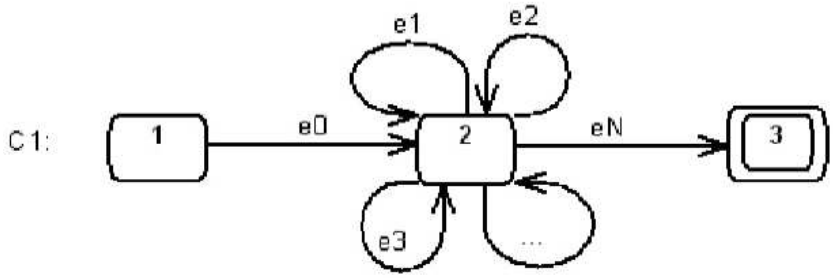

Figure 3. The transition diagram for subsystem $c1$

The correctness of one diagram structure can be tested. In this case the pointed diagram is used as a main diagram and AIM is interpreted

for selected sequence of events names. For example, we can apply the AIM by using the diagram c1 as a main diagram. For that we must fix the $c1$ as the main diagram and we mast prepare the sequences of events (user functionality) accepted by $c1$ diagram. Then it is necessary to interpret this sequence of events according to the AIM. Let the 'e0e1e1e3eN' be prepared to satisfy the user function 1 and the sequence 'e0e2e1e2e1e3eN' – the user function 2. We can make sure that the AIM accepts these sequences.

It is easy to observe that to modify the application it is necessary to modify the STD, but the AIM remains the same. It is very important to notice that the modification of the STD which isn't violating the foregoing limitations don't break the AIM work. Therefore the AIM can be implemented one time and can be reused when constructing the other application.

Practically it is important to outline the classes of problems which can be solved by application of a STD. Evidently the STD can't be used for the systems development for parallel data handling. But it can be used with success on elaborating the structured software systems for management automation.

If the dynamic project is represented by a single transition diagram and the user interaction is required not frequently then one can consider that application of the STD and AIM is less efficient.

If the IHP is very complex and his execution requires an intensive interaction with the user or other actors, then applying the STD is efficient enough. In this case it is very important the generation mode of the sequence of events. There can be used three modes or strategies of driving the AIM's work:

(1) Any functionality of a CIS is defined preliminarily by a finite sequence of the events. This sequence is placed in the queue at the beginning of AIM work. The AIM interprets automatically events from the queue.

(2) The event is placed in queue under every transition to new state of a diagram. The AIM waits the event in every state.

(3) AIM's work is partially driven by finite subsequences of the events, and partially by the event generated under transition into a

new state.

To provide the strategies (2) and (3) it needs to execute some actions under entering the state and under exiting the state. Thereby AIM functioning can be adapted in correspondence with the strategy of AIM driving.

To raise the efficiency of event-driven CIS elaboration process the model of framework was proposed [19]. The static model of the framework is shown in Fig. 4

The ,,UI constructor" component helps to construct the user interfaces of applications. The ,,Librarian" component helps to create and modify the library of program units. The ,,STD Constructor" component helps to construct the inner representation of deterministic STD. The ,,CIS integrator" component integrates together three components (AIM, UI, and STD) and creates an interpretable CIS. The "CIS interpreter" component customizes and starts running the CIS.

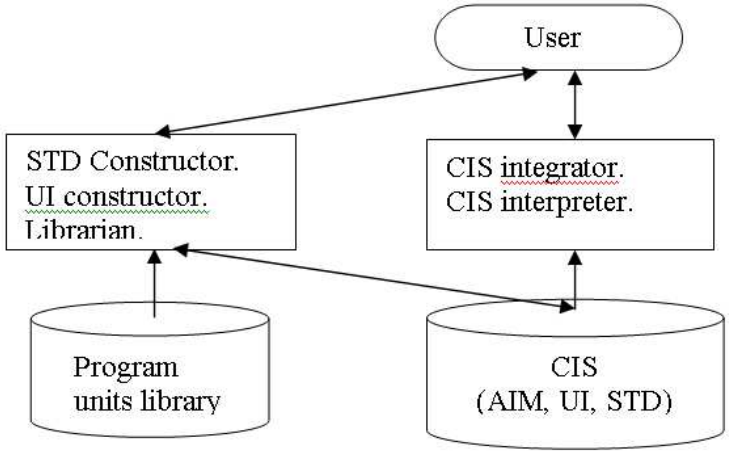

Figure 4. The static model of the framework

The analysis of the functioning principles of the framework per-

mits to affirm that the elaboration process of an event-driven CIS with applying the STD will minimize CIS elaboration time and cost.

6 Conclusions

One can mention that using the STD on software system development offers some important advantages:

1. Application of STD represents a more systematized technique of modeling of complex software system behavior than other known methods. It permits the effective distribution of work among developers and fast assembling of CIS.

2. The AIM represents an invariant part of an application. Therefore it can be constructed once and reused on developing of other CIS. This fact provides the possibility to expand the quality of software systems and reduces time of their construction.

3. The structure of a software system can be modified and extended fast and easy.

4. The automation of the STD creation can reduce the time of CIS construction.

5. STD and AIM represent an efficient mechanism for elaborating event-driven software systems.

To construct quickly the compound software system, the instrumental framework may be constructed using the results stated in this paper.

CIS design by means of STD is based on functional refinement of the project. To improve effectiveness of this type of modeling it needs to be provided with an adequate systematic method of data design and control.

References

[1] Stepan A., Petrov Gh., Iordan V. *The fundamental for design and implementation of software systems.* /MIRTON ed., Timisoara, 1995. 282 p. (In Romanian)

[2] Ciubotaru C.S., Magariu N.A. *Scanners elaboration.* (The workbook.) /State university of Moldova, Chisinu, 1984. 28p. (In Russian)

[3] http://yourdon.com/strucanalysis/wiki/index.php?title=Chapter 13, *#STATE-TRANSITION DIAGRAM NOTATION*, Retrieved January, 2008

[4] Melvin E. Conway. *Design of a separable transition-diagram compiler.* //Communications of the ACM, Volume 6, Issue 7 (July 1963). pp. 396–408.

[5] David Bruce Lomet. *A Formalization of Transition Diagram Systems.* //Journal of the ACM (JACM) V. 20 , Issue 2 (April 1973). pp. 235–257.

[6] Magariu N.A. *The application of transition diagrams on interactive programming system implementation.* //Mathematical researches of Moldova Science Academy, issue 107. The theory and practice of programming. Chisinau, Stiinta, 1989. pp.100–110. (In Russian)

[7] Magariu N.A. *The APL programming language.* /Radio i sviazi, Moscva, 1983. 87 p. (In Russian)

[8] Magariu N. *The using of transition diagrams on systems construction of data processing.* //Proceedings of republican scientific conference "Informatics and computer engineering", Chisinau, 1993, pp. 61–62. (In Romanian)

[9] Ziubin V.E. *The programming of information and management systems based on finite automaton.* (The workbook.)/ State university of Novosibirsck. Novosibirsk, 2006. 96 p. (In Russian)

[10] David Harel. *Statecharts: A visual formalism for complex systems.* //Science of Computer Programming 8, North Holand, 1987. pp. 231–374.

[11] *The software technology based on finite automaton* / http://www.softcraft.ru/auto.shtml#ap. Retrieved January, 2008

[12] *The programming based on automaton* / http://is.ifmo.ru/. Retrieved January, 2008 (In Russian)

[13] Cuznetov B. P. *The psychology of programming based on automaton.* // BYTE/Russian, 11, 2000 and www.softcraft.ru/design/ap/ap01.shtml. Retrieved January, 2008 (In Russian)

[14] Hopcroft J., Motwani R., Ullman J. *Introduction to Automata Theory, Languages and Computation.* /MA: Addison-Wesley, 2001. 521 p.

[15] Shalito A. A., Tokkeli N.I. *The automaton implementation under programming of event-driven sytems.* // A Programer, 4. St. Petersburg State institute of fine mechanics and optics, 2002. pp. 74–80. (In Russian)

[16] Stephen Ferg. *Event-Driven Programming: Introduction, Tutorial, History.* Version 0.2 – 2006-02-08. / http://Tutorial_EventDrivenProgramming.sourceforge.net. Retrieved January, 2008

[17] N.N.Nepeivoda. *Programming styles and methods.* Internet University of Information Technologies intuit.ru, 2005, 320p. (In Russian)

[18] Magariu N., Grico I. *The design dimension of the electronic geographic atlas of Moldova Republic.* //Scientific annals of "A.I. Cuza" university of Iaşi. Number 6. The proceedings of GIS symposium. V. XLVI. Iaşi, 2000. pp.55–60. (In Romanian)

[19] Magariu N. *Some modern peculiarities of development of complex application.* // Microelectronics and Computer Science, int. conf. (5, 2007, Chişinau), Proceeding of the 5^{th} International Conference on ,,Microelectronics and Computer Science", sept.19-21. 2007, Chişinau, Moldova/ com. or: I. Balmuş, V. Ababii,... Ch., U.T.M, 2007. ISBN 978-9975-45-047-8 (vol2). pp. 149–152. (In Romanian)

APPENDIX 1. The Application Interpretation Model.

// AIM

//The initial work state of the AIM is fixed.

i = 0; //The main diagram was selected

j = 0; // The initial nod of diagram was selected

{The content of the input line *IL* is fixed};

k = 1; //k – the index of a symbol from the *IL*.

while (the current node is not the final node of the main diagram)

{ ***while*** **(**IL_k $F(d_l)$ and d_l N_{ij}**)**

{The current state is memorized in stack;

The initial state of diagram $d_l$is fixed as current state; // Values of $i$and $j$variables are modified.

}

if **(**IL_k Σ_{ij}**)**

{ The program unit associated with the edge, marked with IL_k is executed;

The transition through the edge, marked with IL_k is realized and the value of j variable is modified;

k = k + 1;

}

else

{ Message delivery "The sequence of control symbols is wrong"; **exit** (1)}

while **(**The current node is the final node of a diagram that differs from the main diagram**)**

{ The state from the top of the stack is determined as the current state;

The transition through the edges, marked with name of passed diagram, is made;

}

}

{The interpretation is stopped};

Nicolae Magariu

Received February 4, 2008

State University of Moldova,
Mathematics and Computer Science Department,
Research Laboratory "Software System Designing"
E–mail: *magariu@usm.md*

Computer Science Journal of Moldova, vol.**16**, no.2(47), 2008

Using of P2P Networks for Acceleration of RTE Tasks Solving*

Adrian Iftene

Abstract

In the last years the computational Grids have become an important research area in large-scale scientific and engineering research. Our approach is based on Peer-to-peer (P2P) networks, which are recognized as one of most used architectures in order to achieve scalability in key components of Grid systems.

The main scope in using of a computational Grid was to improve the computational speed of systems that solve complex problems from Natural Language processing field. We will see how can be implemented a computational Grid using the P2P model, and how can be used SMB protocol for file transfer. After that we will see how we can use this computational Grid, in order to improve the computational speed of a system used in RTE competition [1], a new complex challenge from Natural Language processing field.

Keywords: Computational Grid, P2P Network, SMB protocol, RTE competition

1 Introduction

To achieve their envisioned global-scale deployment, Grid systems [2, 3] need to be scalable. Peer-to-peer (P2P) techniques are widely viewed as one of the prominent ways to reach the desired scalability. Resource discovery is one of the most important functionalities of a Grid system and, at the same time, one of the most difficult to scale. Indeed, the

*A preliminary version of this paper was presented at 9^{th} SACCS International Conference, Iasi, Romania, 16-17 October 2007

duty of a resource discovery system (such as the Globus MDS [4]) is to provide system-wide up-to-date information, a task which has inherently limited scalability. To add to the challenge, Grid resource discovery systems need to manage not only static resources, but also resources whose characteristics change dynamically over time, making the design critical.

The popularity of distributed file systems has continued to grow significantly in the last years, due to the success of peer-to-peer services like file sharing applications, VoIP applications, Instant messaging and TV on the Internet. Peer-to-peer systems offer a decentralized, self-sustained, scalable, fault tolerant and symmetric network of machines providing an effective balancing of storage and bandwidth resources. The peer-to-peer system described in this paper is **completely distributed** (*requires no form of centralized control, coordination or configuration*), **scalable** (*its nodes control only a certain region which is a part of the whole system and does not depend on the total number of nodes in the system*), and **fault tolerant** (*it can avoid some failures*).

Solving complex problems has become a usual fact in the Natural Language Processing domain, where it is normal to use large information databases like lexicons, semantic relations, dictionaries.

We will see how we can configure a peer-to-peer system in order to find the global solution for the Third Recognising Textual Entailment competition (RTE3) task. The system architecture is based on the peer-to-peer model, using the SMB protocol for file transfer. The system functionality was optimized by using a caching mechanism and a quota for synchronizing the termination of all processes.

After the peer-to-peer network is configured, one computer becomes the initiator and builds the list of available neighbors. Subsequently, the initiator has the following additional roles: split the initial problem into sub-problems, send the sub-problems to the list of neighbors for solving, receive the partial output files and build the final solution.

2 Computational GRID

Grid computing is a phrase in distributed computing which can have

several meanings [5]:

- A local computer cluster which is like a "grid" because it is composed of multiple nodes.
- This computer can offer online computation or storage as a metered commercial service, known as utility computing, computing on demand, or cloud computing.
- It can permit the creation of a "virtual supercomputer" by using spare computing resources within an organization.
- Also, it can create a "virtual supercomputer" by using a network of geographically dispersed computers. Volunteer computing, which generally focuses on scientific, mathematical, and academic problems, is the most common application of this technology.

These varying definitions cover the spectrum of "distributed computing", and sometimes the two terms are used as synonyms.

Functionally, one can also speak of several types of grids:

- *Computational grids* (including CPU Scavenging grids) which are focuses primarily on computationally-intensive operations.
- *Data grids* or the controlled sharing and management of large amounts of distributed data.
- *Equipment grids* which have a primary piece of equipment e.g. a telescope, and where the surrounding Grid is used to control the equipment remotely and to analyze the data produced.

Usually, a *computational Grid* consists of a set of resources, such as computers, networks, on-line instruments, data servers or sensors that are tied together by a set of common services which allow the users of the resources to view the collection as a seamless computing/information environment. The standard Grid services include [6]:

- security services which support user authentication, authorization and privacy

- information services, which allow users to see what resources (machines, software, other services) are available for use,
- job submission services, which allow a user to submit a job to any compute resource that the user is authorized to use,
- co-scheduling services, which allow multiple resources to be scheduled concurrently,
- user support services, which provide users access to "trouble ticket" systems that span the resources of an entire grid.

Our P2P system can be considered like a computational Grid, focused on complex linguistic operations. All standard services were implemented except the security services which are supported by our operating system.

3 P2P Networks

The term Peer-to-Peer (P2P) refers to a class of systems (hardware and software) which employ distributed resources to perform a function in a decentralized manner. Each node of such a system has the same responsibility.

Basic goals are decentralization, immediate connectivity, reduced cost of ownership and anonymity. P2P systems are defined in [7] as "*applications that take advantages of resources (storage, cycles, content, human presence) available at the edges of the Internet*". The P2P architecture can help to reduce storage system costs and allow cost sharing by using the existing infrastructure and resources of several sites. Considering these factors, the P2P systems could be very useful in designing the future generation of distributed file systems.

3.1 Design Issues in P2P Networks

Peer-to-Peer systems have basic properties that separate them from conventional distributed systems. We describe in this section different design issues and their effect in system performance [8].

Symmetry: Symmetry comes among the roles of the participant nodes. We assume that all nodes have the same characteristics.

Decentralization: P2P systems are decentralized by nature, and they have mechanisms for distributed storage, processing, information sharing. In this way scalability, resilience to faults and availability are increased.

Operations with Volunteer Participants: An important design issue is that the participants can neither be expected nor enforced. They haven't a manager or a centralized authority and can be inserted or removed from the system at any time.

Fast Resource Location: One of the most important P2P design is the method used for resource location. Because resources are distributed in diverse peers, an efficient mechanism for object location becomes the deciding factor in the performance of such system.

Load Balancing: Load balancing is very important in one robust P2P system. The system must have optimal distribution of resources based on capability and availability of node resources.

Anonymity: With scope to prevent censorship we need to have anonymity in our system. Our network must to assure anonymity for producer and for consumer of information.

Scalability: The number of peers from network should be any value (hundreds, thousands or millions), and that not affect the network behavior. Unfortunately actual systems are not scalable over few hundreds or thousands of nodes.

Persistence of Information: Methods from our system should be able to provide persistent data to consumers, the data stored in the system is safe, protected against destruction, and highly available in a transparent manner.

Security: Security from attacks and system failure are design goals for every system. The system should be able to assure data security, and this can be achieved with encryption, different coding schemes, etc.

4 Peer to Peer GRID Concepts

P2P systems are divided into two categories in [6]:

- *File sharing utilities* how we already see in FreeNet, which can be characterized as providing a global namespace and file caching and a directory service for sharing files in a wide-area distributed environment. In the most interesting cases, the resources and all services are completely distributed. Each user client program, which can access local files, is also a data server for files local to that host.
- *CPU cycle sharing* of unused user resources usually managed by a central system, which distributes work in small pieces to contributing clients. Examples of this include Seti-athome, Entropia [11] and Parabon [12].

Both of these cases are interesting distributed system architectures. From the perspective of Grid computing there are several compelling features to these systems. First, the deployment model of P2P systems is purely user-space based. They require no system administrator.

Security, when it exists in these systems is based on users deciding how much of their resource they wish to make public to the collective or to the central resource manager. Also P2P systems are designed to be very dynamic, with peers coming and going from the collective constantly as users machines go on and off the network. P2P systems are also doing a good job of bypassing firewalls. This stands in contrast to large-scale scientific Grid systems, which manage large expensive resources and must be constantly maintained and managed by the system administration staff.

5 The System

5.1 CAN

The system P2P architecture is based on CAN model [15]. In order to implement a P2P architecture that respects all these issues, we use

the Content Addressable Network (CAN) presented in [9]. The basic operations performed on a CAN are the insertion, lookup and deletion of nodes. Our implementation is based on [10] and it provides a completely distributed system, scalable and fault-tolerant.

CAN Construction

Our design is over a virtual d-dimensional Cartesian coordinate space. This space is completely logical and we haven't any relation between it and any physical system. In this d-dimensional coordinate space, two nodes are neighbors if their coordinates are the same along d-1 dimensions and are different over one dimension. In this way, if all zones from this space are approximate the same every node has 2d neighbors. To allow the CAN to grow, a new node that joins to this space must to receive its own portion of the coordinate space.

CAN Insertion

The process has three steps:

- The new node must find a node that is already in CAN (source node).
- After that, it knows CAN dimensions and it generates a d-point in this space (this point is in a node zone - destination node). We route from source node to destination node following the straight-line path through the Cartesian space. The destination node then splits its zone in half and assigns one half to the new node.
- In the last step, the neighbors of the split zone must be notified about new node from CAN.

CAN Deletion

When nodes leave from our CAN, we need to ensure that the remaining neighbors take their zones. If the zone of a neighbor can be merged with the node's zone to produce a valid single zone, then this is done.

If not, then the zone is split in zones accordingly with neighborhoods structure. In some cases it is possible like this zone to remain non-allocate, but with first occasion it is used.

5.2 Transfer Protocol

The transfer protocol we used in our approach is based on CIFS (Common Internet File System) [13], the Microsoft version of SMB (Server Message Block) [14]. The main scope of using this transfer protocol is to manage the download and upload of files between nodes from our P2P network. Using our protocol, advantages come from possibility to retry in case of failures and in possibility to use bandwidth partially in order to by-pass network overloaded.

SMBs have a specific format that is very similar for both requests and responses. After connecting at the network level, the client is ready to request services from the server. However, the client and server must first identify which protocol variant they each understand.

The client negotiates the protocol which will be further used in its communication with the server. Once a protocol has been established, the client can proceed to login to the server, if required. One of the most important aspects of the response is the UID of the logged on user. This UID must be submitted with all subsequent SMBs on that connection to the server. The client sends a SMB specifying the network name of the share that they wish to connect to, and if all is proper, the server responds with a TID that the client will use in all future SMBs relating to that share.

Having connected to a tree, the client can now open a file with an open SMB, followed by reading it with read SMBs, writing it with write SMBs, and closing it with close SMBs.

Download and Upload files

Our CIFS implementation respects the following design issues:

- Possibility to split and transfer files in blocks;

- Possibility to retry and resume when connection is lost, in case of power failures;
- Upload/Download files to/from one computer which need user/ password login;
- Possibility to use the bandwidth partially, in order not to overload the network.

Download a file

In order to download a file, the following steps are performed in our implementation:

1. **Login**: First the dialect of CIFS to use is negotiating and the Protocol to use is obtained. Secondly, the connection was set up to the server using credentials and set the UID assigned by the server to the client.
2. **TreeConnect** - connect to the specified folder and set TID identifier
3. **Open File** - open file for read and set FID
4. **ReadFromFile** - read from remote file starting with position offset, length bytes and put the result in buffer

Upload a file

In order to upload a file, the first two steps described above are performed, followed by:

3'. **Open File** - create file or open file for write, append and set FID (flags should be FILE_OPEN_IF)

4'. **WriteToFile** - write to remote file starting with position offset, length bytes from buffer

5.3 System Architecture

The system presented below consists of more core modules (CMs) and databases of linguistic resources. In order to solve the task from RTE competition we must connect to a computer from this computational Grid in order to initiate the problems solving. The system main components and transfer protocols were implemented using Microsoft platform and the C# language.

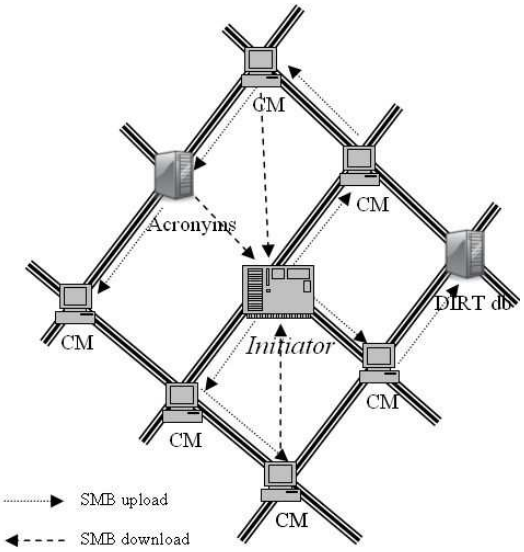

Figure 1. *P2P Network*

Between CMs, the upload and download operations are done using a special component based on the SMB protocol.

Any computer from this network can initiate the solving of the RTE task and it becomes the *Initiator*. First of all it checks its list with neighbors in order to know the number of computers which can be involved in the problem solving (the future CMs). After that it updates all these CMs with the last version of the TE module (this module identifies if we have textual entailment between text and hypothesis).

In parallel, all pairs are sent to the LingPipe and Minipar modules [16, 17], which send back the pairs on which the central module can run the TE module. All necessary files are automatically uploaded from the initiator to the other computers and eventually, in the end the partial results from the other computers are automatically downloaded to the initiator.

After that, the *Initiator* automatically splits the initial problem (consisting of 800 pairs) in sub-problems (a range between the number of the first and last pair) according to the number of its neighbors and using a dynamic quota. At first, this quota has an initial value, but in time it is decreased and eventually becomes the default minimal value. The main goal of using this quota is to send to any neighbor a number of problems according to its computational power and how busy it is in the running moment. In order to accomplish this, each computer is given an initial number of problems for solving and when it finishes, it receives automatically another quota according to the remaining number of problems.

We run in a network with different system configurations of computers (dual core - running two processes and normal - running one process) and with different degrees of business in the execution moment. For that we run Disk Defragmenter on some computers (those marked with *) to see how the initiator distributes the sub-problems to its neighbors.

Table 1. Number of Problems Solved by Computers

quota	**100**	**100**	**100**	**20**	**20**	**20**
Dual1_1	165	163	176	171	178	209
Dual1_2	197	136*	179	145*	159*	221
Dual2_1	175	175	155*	185	138*	108*
Dual2_2	152	183	163	180	129*	130*
Normal	111	143	127	119	196	132

The table above demonstrates the correctness of our supposition regarding the quota mechanism. In all cases presented, the processor solves fewer problems. After splitting the initial problem in sub-

problems, the "Client Module" from the initiator creates threads for each sub-problem. It also creates for every neighbor a sub-client in order to communicate with server components from neighbor and solve the sub-problem (see Figure 2).

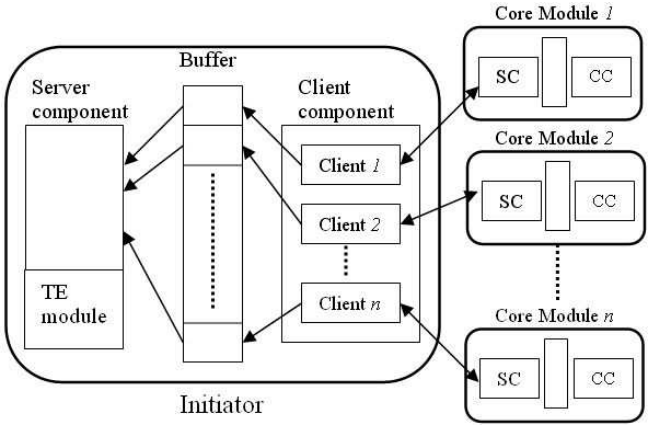

Figure 2. *Client Server architecture*

This sub-client is responsible for sending sub-problem, for receiving the results and saving them to a file with partial results, and informing the initiator server when the solving of sub-problem is finished. In order to inform the server about ending of solving process the client module uses the buffer zone in order to inform the server about that. The client uses the buffer like a "producer" which puts here information about ending of processes, and the server uses the buffer like a "consumer" [18]. The server takes the information and sees how the sub-problem is solved on the corresponding computer and starts the solving of a new sub-problem when the previous one is finished.

5.4 Results

The idea to use a computational Grid came after the first run of the first system build for RTE3 [19]. This first run had no optimization and

took around 26 hours on one computer. In order to observe the system changes faster, we improved this system and during the competition period we used a system which took around 8 hours for a full run.

This initial system was installed on four computers and after manual splitting of problems in sub-problems a single run took only two hours. After all system finished its runs we manually built the final solution from the four partial solutions.

Now, in the current solution the *Initiator* is responsable for splitting of problems in sub-problems, and for collecting of partial results in order to build the final result. Also, a common caching mechanism is used for the large databases of Dirt and WordNet in order to increase the computational speed. For the building of the caching resource, the system was run on one computer. After that, the computational Grid was configured on a P2P network with 5 computers (3 normal and 2 with dual core processor) and a run of the system now takes 6.7 seconds (see Table 2).

Table 2. Details on the Duration of Runs

No	Run details	Duration
1	One computer without caching mechanism	5:28:45
2	One computer with caching mechanism, with empty cache at start	2:03:13
3	One computer with full cache at start	0:00:41
4	5 computers with 7 processes	0:00:06.7

Another big problem was synchronizing the termination of processes. When we manually split the initial problem in sub-problems, if another process (like Windows update or a scanning computer after viruses) starts on some computer, then this computer works slowly. In this case, we have up to 30 minutes delay due to the fact that we have to wait for it to finish solving its sub-problems in order to build the final solution.

Now, using the quota mechanism for synchronizing the ending of all processes, the difference between the time when the first computer from the network finishes its work, and the last computer is around 0.26

seconds. Also, the powerful computers from the network have time to finish more problems while the slow ones solve less.

6 Conclusions and Future Work

In this paper we have shown how to build the system for RTE competition. This system is based on a P2P arhitecture in order to observe the system changes faster. This approach seems as most appropriate given the complexity of the task and the fact that the impact of any change brought to the system must be quantified quickly (the competition has a duration of 3 days). To our knowledge, there were no similar efforts to optimize RTE systems from the computational speed point of view (presently, there are systems whose run takes days to complete).

One important further development of the system will be a web-service allowing users to run in a GRID environment NLP applications. For that we will use for the middleware part of the GRID, the Globus Toolkit [20], a software "work-in-progress" which is being developed by the Globus Alliance [21]. The toolkit provides a set of software tools to implement the basic services and capabilities required to construct a computational GRID, such as security, resource location, resource management, and communications. Our scope is to offer basic linguistic GRID services, and after that, based on these services we will build complex services. Like basic services we already implement in Java GRID services for lemmatization, POS, tokenizing, name entity recognition, WordNet (in English and in Romanian). As complex services we will be focused on special services necessary in Question Answering and in Textual Entailment.

Acknowledgement

The author thanks the members of the NLP group in Iasi for their help and support at different stages of the system development.

The work on this project is partially financed by the CEEX GRAI (Grid Computing and Artificial Intelligence) project number 94 and by

Siemens VDO Iasi.

References

[1] RTE competition: http://www.pascal-network.org/Challenges/RTE3.

[2] G. C. Fox, D. Gannon, *Computational Grids*, IEEE Comput Sci Eng. Vol 3, No. 4, pp. 74–77, 2001.

[3] D. Gannon, and A. Grimshaw, *Object-Based Approaches*, The Grid: Blueprint for a New Computing Infrastructure, Ian Foster and Carl Kesselman (Eds.), pp. 205–236, Morgan–Kaufman, 1998.

[4] Globus MDS. http://www.globus.org/toolkit/mds.

[5] Grid Computing: http://en.wikipedia.org/wiki/Grid_computing.

[6] D. Gannon, R. Bramley, G. Fox, S. Smallen, A. Rossi, R. Ananthakrishnan, F. Bertrand, K. Chiu, M. Farrellee, M. Govindaraju, S. Krishnan, L. Ramakrishnan, Y. Simmhan, A. Slominski, Y. Ma, C. Olariu, N. Rey–Cenvaz, *Programming the Grid: Distributed Software Components, P2P and Grid Web Services for Scientific Applications*, In Cluster Computing journal, Volume 5, Number 3, Pp. 325–336. 2002

[7] S. Androutsellis-Theotokis and D. Spinellis, *A survey of peer-to-peer files sharing technologies*, White paper, Electronic Trading Research Unit (ELTRUN), Athens, University of Economics and Business, 2002.

[8] H. Ragib, A. Zahid, *A survey of peer–to–peer storage techniques for distributed file system*, National Center for Supercomputing Applications Department of Computer Science, April 2005.

[9] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, *A scalable content addressable network*, TR-00-010, 2000.

[10] A. Iftene, C. Croitoru, *Graph Coloring using Peer–to–Peer Networks*, In Proceedings, of 5th International Conference RoEduNet IEEE. Pp. 181–185. Sibiu, Romnia. 1-3 June, 2006.

[11] Entropia Distributed Computing, see http://www.entropia.com.

[12] Parabon Computation, see http://www.parabon.com.

[13] CIFS or Public SMB Information on Common Internet File System: http://support.microsoft.com/kb/199072.

[14] Server Message Block: http://en.wikipedia.org/wiki/Server_Message_Block.

[15] A. Iftene, A., Balahur-Dobrescu, and D. Matei, D. *A Distributed Architecture System for Recognizing Textual Entailment.* In proceedings of 9th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing. Timisoara, Romnia. September 26–29, 2007.

[16] LingPipe: http://www.alias-i.com/lingpipe/.

[17] D. Lin, *Dependency-based Evaluation of MINIPAR*, In Workshop on the Evaluation of Parsing Systems, G ranada, Spain, May, 1998.

[18] G. R. Andrews. *Foundations of parallel and distributed programming.* Addison-Wesley Longman Publishing Co., Inc., 1999.

[19] A. Iftene, A. Balahur-Dobrescu. *Hypothesis Transformation and Semantic Variability Rules Used in Recognizing Textual Entailment.* In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, Pp. 125-130, Prague, June 2007.

[20] Globus Toolkit: http://www.globus.org/toolkit/

[21] Globus Alliance: http://www.globus.org/alliance/

Adrian Iftene

"Al. I. Cuza" University,
Faculty of Computer Science,
General Berthelot Street, No. 16,
Code 700483, Iasi, Romania
E–mail: *adiftene@info.uaic.ro*

Received December 2, 2007

Computer Science Journal of Moldova, vol.**16**, no.2(47), 2008

An Approach of Diagnosis Based On The Hidden Markov Chains Model

Karim Bouamrane, Amel Djebbar, Baghdad Atmani

Abstract

Diagnosis is a key element in industrial system maintenance process performance. A diagnosis tool is proposed allowing the maintenance operators capitalizing on the knowledge of their trade and subdividing it for better performance improvement and intervention effectiveness within the maintenance process service. The Tool is based on the Markov Chain Model and more precisely the Hidden Markov Chains (HMC) which has the system failures determination advantage, taking into account the causal relations, stochastic context modeling of their dynamics and providing a relevant diagnosis help by their ability of dubious information use. Since the FMEA method is a well adapted artificial intelligence field, the modeling with Markov Chains is carried out with its assistance. Recently, a dynamic programming recursive algorithm, called 'Viterbi Algorithm', is being used in the Hidden Markov Chains field. This algorithm provides as input to the HMC a set of system observed effects and generates at exit the various causes having caused the loss from one or several system functions.

Key words: Diagnosis, Markov Chain, Hidden Markov Chain (HMC), FMEA, Viterbi Algorithm.

1 Introduction

Under the effect of multiple causes as wear or deformations etc, equipment tends to worsen in the course of time. These deteriorations can fail down system operations (breakdown), decrease outputs, reduce or

decrease quality and may be increase operation costs. Today, industrial materials availability control makes possible industry action on the production regularity, manufacturing costs, its competitiveness and commercial success.

Selling more and better is not only a better installation control mode proposition but also a fast owner mode intervention and installation maintenance of equipments that guaranties an optimum production satisfaction. Maintenance computerization aims to optimize reliability and equipment availability. Belavilacqua [3] shows that 80 percent of the production systems downtime, following random breakdowns is devoted to the diagnosis maintenance and only 20 percent of this time is used for repair itself. In order to satisfy the new market requirements, each enterprise must provide an identification and an effective and rapid localization of failures and their causes. Thus, each time a failure occurs, it will be identified and immediately corrected. This performance realization requires expert technician or specialized maintenance engineer supervision, knowing the equipment perfectly. Since the production process became complex, including several factors and situations, the diagnosis requires important technical skills and a great experiment, being difficult to find in only one human. For this reason, the Markov Chains are one of the tools used for the maintenance diagnosis. Article continuation is organized in four sections. Section 2 presents a diagnosis definition. Section 3 defines the various existing methods of diagnosis. Section 4 details the developed diagnosis approach. Section 5 deals with an illustrative example.

2 Definition of the diagnosis

Diagnosis is defined as a process of exact failure cause localization. Once the failure is detected, it is the responsibility of the maintenance engineer to recognize the effects, to analyze information, to interpret the various error messages and indications, and to leave with the true diagnosis of the situation in term of components having caused the failure and the reason of their failures. When the diagnosis is completed, the replacement or the repair of the component at the origin of the

failure is the following defect correction stage. The companies are thus confronted with this double economic challenge:

- to increase the productivity by increasing availability of their equipments production;
- to reduce the maintenance costs.

Diagnosis methods are mainly classified according to the used knowledge type [2],[4],[9],[10],[11],[12],[13],[16],[17],[18].

3 Markovian approach for diagnosis

Among the methods of diagnosis quoted previously, we choose the diagnosis by Markov Chain and more precisely the Hidden Markov Chain (HMC) because they make it possible to deal with dubious information problem. The interest to add a hidden layer lies in the ridge in which the failing component is not directly observable. On the other hand, what are observed are the effects having generated this failure [5] [1].

3.1 Discrete time Markov Chain

It is a Markovian discrete states process where observations and states are identical. States are related by probabilistic transitions. A following Markov Chain (MC) models operation:

1. A set of states N: a finite state system where each state is represented by a node. The trajectories between states are symbolized by directed arcs. We denote n_i *the* n^{th} state visited by the system,
2. A distribution: $\pi i = P(n0 = i)$ on the initial state. With a known probability, the state can then evolve, and pass from $n_i = i$ *to* $n_{i+1} = j$. Generally, this probability depends on the sequence of states occurrence since the initial moment. But in the case of the first order Markov models, dependence stops at a previous moment n.

$$P(n_{i+1}/n_0 = i_0, \ldots, n_{i-1} = i_{i-1}, n_i = i) = P(n_{i+1} = j/n_i = i) \qquad (1)$$

3. The transitions probabilities of the Markov chain are specified by a matrix of transition $A = a_{ij}$

$$a_{ij} = P(n_{i+1} = j/n_i = i), \forall i, j \in n \tag{2}$$

with

$$\Sigma a_{i,j} = 1, \forall i, j \in n \tag{3}$$

Thus, the built model takes the name of Markov Chain.

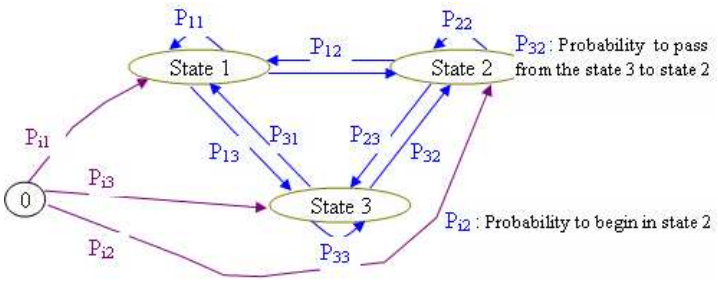

Figure 1. Markov Chain Model

Definition 1: A Markov chain is defined by: $\lambda = (N, \Pi, A)$ With N a finished spaces state, Π a probability distribution on the initial state and A the transition probabilities matrix.

3.2 Hidden Markov Chain

It is a Markov chain where the states are not the same as observations. Each state will generate an observation. It is said that the states are hidden (they are not directly observed). In a Hidden Markov Model (HMM), the states $N = (n_1, n_2, \ldots, n_n)$, are non observable, however they emit observable signals $O = (O_1, O_2, \ldots, O_n)$ which are balanced by their probability. It is characterized by:

1. The states $N = \{n_1, n_2, \ldots, n_n\}$,
2. The transition matrix $A = \{a_{ij} = P(n_j/n_i\}; \Sigma a_{i,j} = 1$

3. The initialization vector $\Pi = \{\pi_i = P(n_i)\}; \Sigma\pi_i = 1$

4. Probabilities that the state n_i emits the signal of observation o_k. They are gathered in an emission matrix $B = \{b_i(o_k) = P(o_k/n_i)\}; \Sigma a_{ij} = 1$

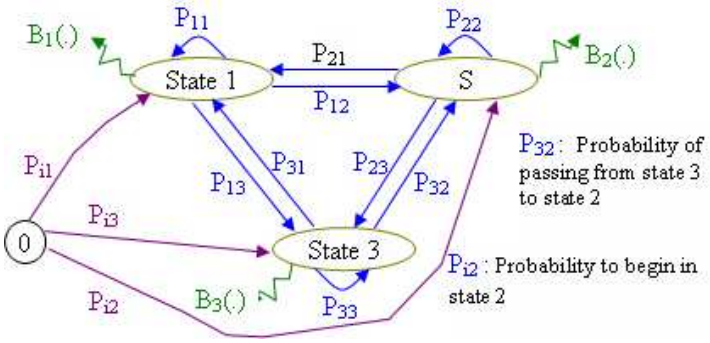

Figure 2. Hidden Markov Chain Graph

Definition 2: A Hidden Markov Chain is defined by:

$$\lambda = (N, \Pi, A, B) \tag{4}$$

With N – a finite state, Π – an initial distribution probability state, A – the transition probabilities matrix and B – the observation probabilities matrix. A Hidden Markov Chain is a Markov chain of order one with non observable states (emission of a signal) and B – the observation probabilities matrix.

3.3 Application of Markovian modeling to the diagnosis

Since Markov chains and more specifically Hidden Markov chains Model (HMM) can deal with uncertain information problems, we opted for their use as a solution for the diagnosis problem. The use of a hidden layer in our case lies in the ridge that the component failure is not directly observable. But, what were observed are effects that produced this failure. In diagnosis, the order one ergodic Markov model, figure

(3), lets any state be reached from any other one within a finite number of transitions. This type of model is more general and interesting when the model represents a following process statements evolution.

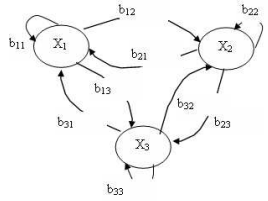

Figure 3. The Ergodic Topology

The HMM that we consider for the diagnosis has the following characteristics [6], figure (4):

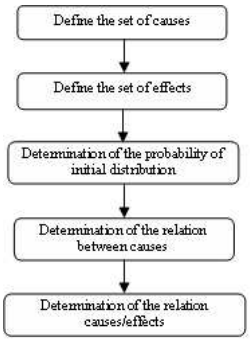

Figure 4. General diagram of Diagnosis by HMM

1. each state is a breakdown state;

2. each observation is an observed effect on the system;

3. the initial probability Π of the HMM states: this distribution on the set N of the states is supposed known; it represents the confidence which we have on the operation system;

4. the transition probabilities matrix between states: it depends on the connections of cause and effects between the breakdown states.

5. Matrix emission probabilities: it depends on the relationship between the failure effects and causes.

The diagnostic system based on the model Hidden Markov chains for an industrial process is constructed in two steps:

Step 1: Construction of the model for each system. This step is carried out using a solution of the evaluation problem and learning to estimate efficiently the optimal parameters of the model in order to maximize the probability of the observed sequence.
Step 2: Assess the following statements. This step is carried out using a solution to the problem of recognition. Once the parameters of HMM constructed and optimized, the defining of the set of statements that have caused the system breakdown is still pending.

The diagnostic system by Hidden Markov Chains works like figure (5):

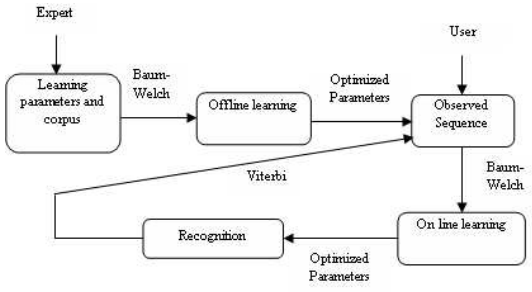

Figure 5. Diagnostic System Functioning

Phase 1: Once the various parameters associated with HMM are entered, the diagnostic system performs offline learning with a corpus of learning (sequence of observations) provided by the expert and using Baum-Welch algorithm.

Phase 2: Once the maintenance operator enters the failure of the observed effects in the system, an online learning will be done using the Baum-Welch algorithm.
Phase 3: Finally, the system will determine the sequence of the optimal failure statements in the sense of the post maximum likelihood, and this is estimated recursively by the Viterbi algorithm [8].

The approach Characteristic The Markovian approach performance depends on the system knowledge a priori richness. Indeed, the real process modeling based on HMM is effective if the model parameters are correctly estimated because the latter have a direct influence on the obtained results. The Markovian model combinative explosion phenomenon is well known: the number of states increases exponentially according to the breakdown number.

4 Illustrative example

The case study was carried out on the ANABIB[1] Company equipment in order to improve its maintainability. Indeed, being given the heavy investment in this sector, the machines often function in two teams. Any breakdown can cause non recoverable delays for the production service. In the context of an effectiveness preoccupation, we choose the most critical company equipment. It is about a machine illustrated in figure (6) which, starting from a steel reel, gives a structural shapes named "Skelp". It ensures the forming and welding operation.

We had carried out cutting of the machines into a subset and then into bodies by specifying each element functions and by using the various functional analysis methods [19] [20] [21]. Then, we were brought to identify for each body all its modes of failures, their possible causes, their respective effects and repair measurements to be applied using FMEA analysis [14]. This phase was not easy because the company does not have Computer-Assisted Maintenance and Management software.

[1]ANABIB: Public Economic Enterprise of the iron and steel sector in Algeria

Figure 6. Forming-Welding machine

The principal failures of this equipment are as follows: 1. wear ball. 2. wear bearing. 3. wear roll.

Effects observed: 1. Bad forming. 2. Oil leakage.

Since Markovian methods are used, work consists in a correct problem specification. That is done by the states set determination which represents in this case the set of the failures quoted previously, the calculus of the transition probability which is defined by the failure rate hh such as hh= 1/COAT (Correct Operation Average Time) and the calculus of appearance probability of each effect determined by the frequency factor of FMEA table. Thus, we can build:

1. The transition matrix A with the states set as lines and columns.

$$A = \begin{array}{|c|c|c|} \hline 0.3 & 0.5 & 0.2 \\ \hline 0 & 0.3 & 0.7 \\ \hline 0 & 0 & 1 \\ \hline \end{array}$$

2. The observation matrix B with the various states as lines and the observed effects set as columns.

$$B = \begin{array}{|c|c|}\hline 1 & 0 \\ \hline 0.5 & 0.5 \\ \hline 0 & 1 \\ \hline \end{array}$$

The initial probabilities vector Π such as $P(\pi_i) = 1/\text{numbers of causes}$.

$$\Pi = \begin{array}{|c|}\hline 0.33 \\ \hline 0.33 \\ \hline 0.33 \\ \hline \end{array}$$

If we supposed that the observed sequence O = bad forming, bad forming, oil leakage, oil leakage after application of the Viterbi algorithm, we obtained:
- The matrix δ :

$$\delta = \begin{array}{|c|c|c|c|}\hline 0.33 & 0.09 & 0 & 0 \\ \hline 0.16 & 0.08 & 0.02 & 0.03 \\ \hline 0 & 0 & 0.05 & 0.05 \\ \hline \end{array}$$

- The matrix ψ :

$$\psi = \begin{array}{|c|c|c|c|}\hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ \hline 0 & 1 & 1 & 2 \\ \hline \end{array}$$

- The sequence of states having generated $N*$ is wear ball, wear roll, wear bearing, wear bearing.

5 Conclusion

In the new economic context, the survival of small, average and large company depends on their ability to maintain their productivity. To

achieve this goal, the company is brought to control the system production and to improve products quality.

We discussed in this article the necessary tool to develop a Markov Chain diagnostic system and more particularly the Hidden Markov chain model. We proposed the hidden Markov chain model in order to be able to mitigate the difficulties related to the realization of a diagnosis system. Indeed, the use of an FMEA analysis for Markovian modeling makes it possible to facilitate the construction of the used model for the diagnosis.

The FMEA Analysis for the presentation of the effect/cause relations as for the determination of the probabilities of the signals emitted by the various considered states and the HMM adapted to take into account of incomplete and dubious information makes it possible to obtain coherent results and to provide a relevant help to the diagnosis.

Finally, the use of the Viterbi recursive dynamic programming algorithm enabled us to determine the failures sequence affecting the system. From a prospective point of view, this work will continue with the development of various tools for diagnosis quoting the Bayesian networks and the expert systems in order to determine the most relevant tool by using a multi criterion method for outclassing using various criteria such as the response time, the development cost.

References

[1] T.Alani, Guellif, H., *Modèles de Markov cachés: thérie et techniques de base*, Rapport de recherche de l'INRIA, 1994.

[2] A.Benveniste., *Markov nets: probabilistic models for distributed and concurrent systems*, IEEE; 2004.

[3] M.Bevilacqua., *The analytic hierarchy process applied to maintenance strategy selection, realibility engineering and system safety*, 2000.

[4] J.N Chatain., *Diagnostic par système expert*, Editions Hermès, 1993.

[5] J. Dequier., *Chaînes de Markov et applications*, Thèse, Université de Grenoble, 2005.

[6] A.M.Djebbar, K.Bouamrane et B.Beldjilali, *Une approche de diagnostic basé sur le modèle de Chaînes de Markov Cachées*, Conférence Internationale sur informatique et ses Applications (CIIA06), 14, 15 et 16 Mai 2006 Saida.

[7] B. Dubuisson., *Diagnostic, intelligence artificielle et reconnaissance des formes*. Traité IC2: Information - Commande - Communication. Hermes, 2001.

[8] G.D. Forney, *The Viterbi Algorithm*, Proceedings of the IEEE, volume 61, No. 3, March 1973.

[9] J.Gertler., *Fault detection and diagnosis in engineering systems*, INC, New York, 1998.

[10] N.Hernandez-gress., *Système de diagnostic par réseaux de neurones et statiques : application à la détection d'hypovigilance du conducteur automobile*, Thèse de doctorat, Institut polytechnique de Toulouse, 1998.

[11] K.Balakrishnan, Honavar V., *Intelligent Diagnosis Systems*, Journal of Intelligent Systems, 1998.

[12] J.Laurens., Hernandez-gress N., Titli A., Esteve D., *Conception et réalisation d'un système de diagnostic temps réel à base de logique floue : application au diagnostic de l'état de vigilance d'un conducteur automobile*, Rapport LAAS no.96314, 1996.

[13] M.A.Maalej, Delcroix V., Piechowiak S., *Les réseaux bayésiens pour la recherche des diagnostics*, Colloque PENTOM 2003, pp. 277–287, ISBN 2-905725-51-6, Valenciennes, 2003.

[14] M.N Ngote., *Présentation de la méthodologie AMDEC*, Ecole nationale de l'industrie minérale, 2002.

[15] L.R.Rabiner, *A tutorial on Hidden Markov Models and selected applications in speech recognition*, In the proceedings of IEEE, volume 77, No. 2:257–285, February 1989.

[16] L. Tromp., *Surveillance et diagnostic de systèmes industriels complexes : une approche hybride numérique/symbolique*, Thèse, Université de Rennes 1,2000.

[17] R.Valette, Kunzle L.A., *Réseaux de Petri pour la détection et le diagnostic*, Rapport LAAS 94463,1994.

[18] G.Zwingelstein., *Diagnostic des défaillances*, Editions Hermès,1995.

[19] http://www.adullact.org/documents/AnalyseFonctionnelle-v1.3.pdf

[20] http://perso.wanadoo.fr/philippe.fichou/documents/Systemes2004.pdf

[21] http://www.inh.fr/enseignements/idp/idp2005/besoin/pieuvresecateur.pdf

Karim Bouamrane, Amel Djebbar,
Baghdad Atmani

Received November 26, 2007

Computer Science Department,
Faculty of Science, University of Oran,
BP 1524 El M'Naouer 31000 Oran, Algeria;
E–mail: *bouamrane.karim@univ − oran.dz*, *bbatmani@yahoo.fr*

Computer Science Journal of Moldova, vol.**16**, no.2(47), 2008

Evaluation of the traffic coefficient in priority queueing systems

Gh. Mishkoy, A. Iu. Bejan, O. Benderschi

Abstract

Methodology and algorithms of evaluation of the traffic coefficient in priority queueing systems with zero and nonzero switchover times are presented. In the case of zero switchover times the calculation of the traffic coefficient is straightforward. In contrast, it relies heavily on the efficient numerical evaluation of the busy period's Laplace-Stieltjes transform in the case when switchover times are not all degenerate zero. Examples for both cases are provided.

This work was done under support of the SCOPES grant IB7320-110720. The second author also acknowledges the support of James Watt scholarship and Overseas Research Students Awards Scheme.

Keywords: priority queues, switchover times, traffic coefficient.

1. Introduction

It is a usual practice to represent and to study real world phenomena processes by mathematical models. Among the latter there are models of priority queueing systems. The theory of priority queueing systems is concerned with the phenomena of prioritized servicing—the incoming requests should be classified by their importance and served according to assigned priority labels. In comparison with other queueing models the priority queueing systems have a more complicated structure, which limits the possibility of their exact analytical analysis. Thus, many results are derived considering the stationary behaviour of the system.

The traffic coefficient is an important measure of the performance of a queueing system and it is responsible for the workload of the system. Analysis of queueing systems delivers formulae for system performance characteristics—many of such analytical expressions involve the traffic coefficient ρ. In the case of priority queueing systems $M|G|1|\infty$ and $M_r|G_r|1|\infty$ with zero switchover times one can easily evaluate ρ via analytic formulae using the rates of incoming flows and mean values of corresponding service times. However, in the case of priority systems with random switchover times, one should be able to evaluate the Laplace-Stieltjes transforms (LST's) of the system busy period in order to estimate the value of the traffic coefficient. Generally, this can only be done numerically.

2. Priority Queueing Systems with Switchover Times

2.1 Description

Consider a queueing system with a single server and r classes of incoming requests, each having its own flow of arrival and waiting line. We call the requests from the i^{th} queueing line L_i i-requests. The i-requests have a higher priority than j-requests if $1 \leq i < j \leq r$. The server gives a preference in service to the requests of the highest priority among those presented in the system.

Suppose that the time periods between two consecutive arrivals of the requests of the class i are independent and identically distributed with some common cumulative distribution function (cdf) $A_i(t)$ with mean $\mathbb{E}[A_i]$, $i = 1, \ldots, r$. Similarly, suppose that the service time of a customer of the class i is a random variable B_i with a cumulative distribution function $B_i(t)$ having mean $\mathbb{E}[B_i]$, $i = 1, \ldots, r$.

It is assumed that the server needs some additional time to proceed with the switching from one priority waiting line of requests to another. This time is considered to be a random variable, and we say that C_{ij} is the time of switching from the service of i-requests to the service of j-requests, if $1 \leq i, j \leq r$, $i \neq j$.

We adopt the classification and the terminology introduced in [1, 2]. We also explain some additional notions and notations.

Definition 1. *By a kk-busy period call the period of time which starts when a k-request enters the empty system and finishes when there are no longer k-requests in the system. Denote the kk-busy period by* Π_{kk}.

Definition 2. *By a k-busy period call the period of time which starts when an i-request enters the empty system,* $i \leq k$, *and finishes when there are no longer k-requests in the system. Denote the k-busy period by* Π_k.

Note, that an r-busy period is nothing but the system's busy period Π, i.e. $\Pi \equiv \Pi_r$.

The following two notions are due to the fact that both servicing and switching can be interrupted under preemptive service and switching policies.

Definition 3. *By a k-cycle of service call the period of time which starts when server begins the servicing of a k-request, and finishes when this request leaves the system. Denote the k-cycle of service by* H_k. *If the servicing of a certain k-request is not interrupted then the corresponding realization of* H_k *coincides with the time* B_k *this request was being serviced.*

Definition 4. *By a k-cycle of switching call the period of time which starts when the server begins the switching to the line of k-requests, and finishes when the server is ready to provide service to these requests. Denote the k-cycle of switching by* N_k. *If the switching from the* i^{th} *line to the* k^{th} *line is not interrupted, then the corresponding realization of* N_k *coincides with the time* C_{ik} *this ik-switching lasted.*

Let $\Pi_{kk}(t)$, $\Pi_k(t)$, $H_k(t)$ and $N_k(t)$ be the cumulative distribution functions of kk-busy periods, k-busy periods, k-cycle of service and k-cycle of switching, correspondingly. Let also $\pi_{kk}(t)$, $\pi_k(t)$, $h_k(t)$ and $\nu_k(t)$ be their Laplace-Stieltjes transform, i.e.

$$\pi_{kk}(s) = \int_0^\infty e^{-st} d\Pi_{kk}(t), \ldots, \ \nu_k(s) = \int_0^\infty e^{-st} dN_k(t).$$

Finally, let $\beta_i(s)$ be the Laplace-Stieltjes transform of $B_i(t)$, i.e.

$$\beta_i(s) = \int_0^\infty e^{-st} dB_i(t).$$

From now on throughout the text we assume that C_{ij} do not depend on i and only depend on j, i.e. $C_{ij} \equiv C_j$, $i = 1, \ldots, r$. Denote the Laplace-Stieltjes transform of C_j with cdf $C_j(t)$ by $c_j(s)$:

$$c_j(s) = \int_0^\infty e^{-st} dC_j(t).$$

2.2 Priority Queueing Systems with Poisson incoming flows

The queueing systems with Poisson incoming flows are of great importance in the theory and practice. In this case the interarrival times are exponentially distributed, i.e. $A_i(t) = 1 - e^{-a_i t}$, $i = 1, \ldots, r$, where a_1, a_2,..., a_r are some positive real numbers with the physical meaning of the flow arrival rates. The compound flow of the requests with priority not greater than k is Poisson with the arrival rate $\sigma_k = \sum_{i=1}^{k} a_i$.

Using the extended Kendall notation we write $M_r|G_r|1|\infty$ to denote a priority queueing system with Poisson incoming flows of requests and random switchover times.

3. Traffic coefficient and its calculation

Analysis of queueinq systems delivers formulae for systems performance characteristic—many of such analytical expressions involve the traffic coefficient ρ.

3.1 Zero switchover times

In the case of priority queueing systems $M_r|G_r|1|\infty$ with degenerated switchover times one can easily evaluate ρ via analytic formulae using

the rates of incoming flows and mean values of corresponding service times.

For instance, the traffic coefficient of the system $M_r|G_r|1|\infty$ can be calculated as follows [1]:

$$\rho = \sum_{i=1}^{r} a_i b_i, \tag{1}$$

where

- for the service scheme "repeat again"

$$b_i = \frac{1}{\sigma_{i-1}} \left[\frac{1}{\beta_i(\sigma_{i-1})} - 1 \right] \tag{2}$$

- for the service scheme "resume"

$$b_i = \mathbb{E}[B_i] \tag{3}$$

- for the service scheme "loss"

$$b_i = \frac{1}{\sigma_{i-1}} \left[1 - \beta_i(\sigma_{i-1}) \right]. \tag{4}$$

In the case when $\rho > 1$, the following takes place: $\pi(0) < 1$, and $\Pi(t)$ is an improper cumulative distribution function, i.e.

$$\lim_{t \to \infty} \Pi(t) < 1,$$

which means that the busy period is of indefinite length with a positive probability. However, if $\rho < 1$, then $\pi(0) = 1$ and the cdf $\Pi(t)$ of the busy period Π is proper. These comments motivate the presence of the quantity $\pi(0) \equiv \pi_r(0)$ in our further examples.

Remark 1. *It is easy to see that the value of the traffic coefficient in the system $M_r|M_r|1|\infty$ running under the service scheme "repeat again" is the same as the value of the traffic coefficient of the same system running under the service scheme "resume". To see this, one*

should compare (2) and (3). The LST's of the exponential service times B_i can be written as follows:

$$\beta_i(s) = \frac{1}{s\mathbb{E}[B_i] + 1},\ i = 1, \ldots, r. \tag{5}$$

Calculation of b_i in (2) using (5) shows that

$$b_i = \frac{1}{\sigma_{i-1}} \left[\frac{1}{\beta_i(\sigma_{i-1})} - 1 \right] = \frac{1}{\sigma_{i-1}} \left(\sigma_{i-1}\mathbb{E}[B_i] + 1 - 1 \right) = \mathbb{E}[B_i],$$

i.e., the traffic coefficients for the priority systems with exponential switchover times under the service schemes "resume" and "repeat again" coincide (for both schemes ρ should be calculated using (1) with the same values of a_i, $i = 1, \ldots, r$).

Example 3.1.1. *Consider the system $M_{10}|M_{10}|1|\infty$ with zero switchover times. In this case*

$$\beta_i(s) = \frac{1}{s\mathbb{E}[B_i] + 1},\ i = 1, \ldots, 10.$$

Let $a_i = 1$ and $\mathbb{E}[B_i] = 0.05$, $i = 1, \ldots, 10$. The results of our calculations of the traffic coefficient ρ and the LST of the busy period using the given algorithms can be found in Table 1. The approximation error ϵ is taken to be equal to 0.001.

preemptive service schemes:	*repeat again*	*resume*	*loss*
ρ	0.5	0.5	0.413914
$\pi_{10}(0)$	0.999959	0.999959	0.999989

Table 1. Calculation of the traffic coefficient ρ and $\pi_{10}(0)$ in Example 3.1.1.

Example 3.1.2. *Consider $M_{10}|G_{10}|1|\infty$ with zero switchover times running under the service scheme "repeat again". Let a_i = 45,*

i=1,...,10. The service times of the requests from the queueing priority lines L_1, L_2, L_3, L_4 are exponential Exp(20), the service times of the requests from the lines L_5, L_6, L_7, L_8 are uniformly distributed on the interval [0,1], and, finally, the service times of the requests from the lines L_9 and L_{10} are of Erlang type Er(2, 20). The approximation error ϵ is taken to be equal to 0.001*. Thus, the following holds:*

$$\beta_i(s) = \frac{20}{20+s},\ i = 1,2,3,4,$$
$$\beta_i(s) = 1 - e^{-s},\ i = 5,6,7,8,$$
$$\beta_i(s) = \left(\frac{20}{20+s}\right)^2,\ i = 9,10.$$

For this system the traffic coefficient ρ was numerically estimated to be equal to $104.0625 \gg 1$*, whereas* $\pi(0) \equiv \pi_{10}(0) = 0.429542 < 1$*. This clearly shows that the system is under the heavy traffic regime.*

3.2 Nonzero switchover times

Let us assume that at least one random variable C_j is not constant zero. In this case the the traffic coefficient can be calculated using the following formula [1]:

$$\rho = \sum_{k=1}^{r} a_k b_k,$$

where

$$b_1 = -\frac{\beta'(0) + c_1'(0)}{1 - a_1 c_1'(0)},$$

and

- for the service scheme "repeat again"

$$b_k = f_1 \dots f_{k-1} \frac{1}{\sigma_{k-1} c_k(\sigma_{k-1})} \left[\frac{1}{\beta_k(\sigma_{k-1})} - 1\right] \qquad (6)$$

- for the scheme "loss"

$$b_k = f_1 \dots f_{k-1} \frac{1}{\sigma_{k-1} c_k(\sigma_{k-1})} [1 - \beta_k(\sigma_{k-1})] \tag{7}$$

- for the service scheme "resume"

$$b_k = f_1 \dots f_{k-1} \frac{1}{c_k(\sigma_{k-1})} \mathbb{E}[B_k]. \tag{8}$$

Here

$$f_1 = 1,$$

$$f_k = 1 + \frac{\sigma_{k-1} - \sigma_{k-1}\pi(a_k)}{\sigma_{k-1}} \left(\frac{1}{c(\sigma_{k-1})} - 1 \right),$$

where $\sigma_0 = 0$ and $\pi_k(t)$ is the Laplace-Stieltjes transform of the busy period $\Pi(t)$.

Value of ρ can be evaluated numerically. In what follows we present algorithms of numerical evaluation of the LST of the k-busy periods and the traffic coefficient ρ. For simplicity we only give the algorithms for the systems $M_r|G_r|1|\infty$ under the service scheme "repeat again" and preemptive switchover policy. In order to calculate ρ one needs to be able to evaluate the LST of the busy period (r-busy period). This can be done numerically using the following sample algorithm [3].

Algorithm 1 $BPLST_E$ (***for the system*** $M_r|G_r|1$ ***with switchover times under the preemptive service scheme "repeat again"***)

Input: r, s^*, $E > 0$, $\{a_k\}_{k=1}^r$, $\{\beta_k(s)\}_{k=1}^r$, $\{c_k(s)\}_{k=1}^r$.
Output: $\pi_k(s^*)$
Description:
IF (k==0) THEN $\pi_0(s^*) := 0$; *RETURN*
$k := 1$; $q := 1$; $\sigma_0 := 0$;
Repeat
inc(q);
$\sigma_q := \sigma_{q-1} + a_q$;
Until $q == r$;

Repeat

$\nu_k(s) := c_k(s^* + \sigma_{k-1})\{1 - \frac{\sigma_{k-1}}{s^*+\sigma_{k-1}}\,[1-$

$-c_k(s^* + \sigma_{k-1})]\,\pi_{k-1}(s^*)\}^{-1};$

$h_k(s^*) := \beta_k(s^* + \sigma_{k-1})\{1 - \frac{\sigma_{k-1}}{s^*+\sigma_{k-1}}\,[1-$

$-\beta_k(s^* + \sigma_{k-1})]\,\pi_{k-1}(s^*)\nu_k(s^*)\}^{-1};$

$\pi_{kk}^{(0)}(s^*) := 0;\ n := 1;$

Repeat

$\pi_{kk}^{(n)}(s^*) := h_k(s^* + a_k - a_k\pi_{kk}^{(n-1)});$

$inc(n);$

Until $|\pi_{kk}^{(n)}(s^*) - \pi_{kk}^{(n-1)}(s^*)| < E;$

$\pi_{kk}(s^*) = \pi_{kk}^{(n)}(s^*);$

$$\pi_k(s^*) := \frac{\sigma_{k-1}\pi_{k-1}(s^* + a_k)}{\sigma_k} + \frac{\sigma_{k-1}}{\sigma_k}(\pi_{k-1}(s^* + a_k - a_k\pi_{kk}(s^*)))-$$

$$-\pi_{k-1}(s^*+a_k)\nu_k(s^*+a_k[1-\pi_{kk}(s^*)])+\frac{a_k}{\sigma_k}\nu(s^*+a_k-a_k\pi_{kk}(s^*))\pi_{kk}(s^*);$$

$inc(k);$

Until $k == r;$

End of Algorithm 1 $BPLST_E$**.**

Remark 2. *Algorithm 1 is convergent. However, it does not provide one with the absolute error of the approximation. In this algorithm some quantity* E *is used to judge on the convergence of the Cauchy sequence* $\{\pi_{kk}^{(n)}(s^*)\}_{n=0}^{\infty}$.

Improved algorithms which evaluate the LST of the busy period with a certain precision are discussed in [3, 4]. We present next Algorithm $BPLST_\epsilon$, which was introduced and discussed in [3]. It is an

improved algorithm of numerical evaluation of the LST of the busy period with a given precision and it is based on the acceleration scheme of solving the Kendall equation [5].

Algorithm 2 $BPLST_\epsilon$ (***for the system*** $M_r|G_r|1$ ***with switch-over times under the preemptive service scheme "repeat again"***)

Input: r, s^*, $\epsilon > 0$, $\{a_k\}_{k=1}^r$, $\{\beta_k(s)\}_{k=1}^r$, $\{c_k(s)\}_{k=1}^r$
Output: $\pi_k(s^*)$
Description:
IF (k==0) THEN $\pi_0(s^*) := 0$; *RETURN*
$k := 1$; $q := 1$; $\sigma_0 := 0$;
Repeat
inc(q);
$\sigma_q := \sigma_{q-1} + a_q$;
Until $q == r$;

Repeat
$\nu_k(s^*) := c_k(s^* + \sigma_{k-1})\{1 - \frac{\sigma_{k-1}}{s^*+\sigma_{k-1}}\,[1 - c_k(s^* + \sigma_{k-1})]\,\pi_{k-1}(s^*)\}^{-1}$;
$h_k(s^*) := \beta_k(s^* + \sigma_{k-1})\{1 - \frac{\sigma_{k-1}}{s^*+\sigma_{k-1}}\,[1 - \beta_k(s^* + \sigma_{k-1})]\,\pi_{k-1}(s^*)\nu_k(s^*)\}^{-1}$;
$\pi_{kk}^{(0)}(s^*) := 0$; $n := 1$;
Repeat
$\underset{\sim}{\pi}_{kk}^{(n)}(0) := 0$; $\widetilde{\pi}_{kk}^{(n)}(0) = 1$;

Repeat
$\widetilde{\pi}_{kk}^{(n)}(s^*) = h_k(s^* + a_k - a_k\widetilde{\pi}_{kk}^{(n-1)}(s^*))$;
$\underset{\sim}{\pi}_{kk}^{(n)}(s^*) = h_k(s^* + a_k - a_k\underset{\sim}{\pi}_{kk}^{(n-1)}(s^*))$;
inc(n);
Until $\frac{\widetilde{\pi}_{kk}^{(n)}(s^*) - \underset{\sim}{\pi}_{kk}^{(n-1)}(s^*)}{2} < \epsilon$;

$$\pi_{kk}(s^*) := \frac{\widetilde{\pi}_{kk}^{(n)}(s^*)+\underset{\sim}{\pi}_{kk}^{(n-1)}(s^*)}{2};$$

$$\pi_k(s^*) := \frac{\sigma_{k-1}\pi_{k-1}(s^*+a_k)}{\sigma_k} + \frac{\sigma_{k-1}}{\sigma_k}(\pi_{k-1}(s^*+a_k-a_k\pi_{kk}(s^*)))-$$

$$-\pi_{k-1}(s^*+a_k)\nu_k(s^*+a_k[1-\pi_{kk}(s^*)])+\frac{a_k}{\sigma_k}\nu(s^*+a_k-a_k\pi_{kk}(s^*))\pi_{kk}(s^*);$$

$inc(k)$;
Until $k == r$;

End of Algorithm 2 $BPLST_\epsilon$**.**

The following is the model algorithm of calculation of the traffic coefficient for the priority queueing systems with switchover times [3].

Algorithm 3 WLCOEEF (*for the system* $M_r|G_r|1$ *with switchover times under the preemptive "repeat again" service scheme*)

Input: r, $\{a_k\}_{k=1}^r$, $\{\beta_k(s)\}_{k=1}^r$, $\{c_k(s)\}_{k=1}^r$.
Output: ρ
Description:
$k := 1$; $\rho := 1$; $\sigma_0 := 0$; $\sigma_1 := a_1$;
$f_1 := 1$; $p := 1$;
$b_1 := -(\beta'(0) + c_1'(0))/(1 - a_1c_1'(0))$;
$\rho := a_1b_1$;

Repeat
$inc(k)$;
$\sigma_k := \sigma_{k-1} + a_k$;

$b_k := p\frac{1}{\sigma_{k-1}c_k(\sigma_{k-1})}\left(\frac{1}{\beta_k(\sigma_{k-1})} - 1\right)$;

$\rho := \rho + a_k b_k$;

$f_k := 1 + \frac{\sigma_k - \sigma_{k-1}\pi_{k-1}(a_k)}{\sigma_{k-1}} \left(\frac{1}{c_k(\sigma_{k-1})} - 1\right)$;

$p := f_k p$;

Until $k == r$;

End of Algorithm 3 WLCOEEF.

Remark 3. *Calculation of ρ in Algorithm 3 requires calculation of $\pi_{k-1}(a_i)$, $k = 2, \ldots, r$, which, in turn, can be realized using Algorithm 1 or Algorithm 2. For a different scheme of service or switching policy one should employ the corresponding formulae for the LST's $h_k(s)$, $\nu_k(s)$, $\pi_{kk}(s)$, $\pi_k(s)$ (see [1]).*

Example 3.2.1. *Consider the system $M_5|M_5|1$ with all interarrival times being distributed exponentially Exp(10) and all service times being distributed exponentially Exp(200). The switchover times C_k are all distributed as Exp(100), $k = 1, \ldots, 5$; the service scheme is "repeat again". The results of calculations for such systems can be found in Table 2. The quantity ϵ is taken to be* 0.001.

	$BPLST_E$	$BPLST_\epsilon$	*difference*
$\pi_1(10)$	0.859847	0.858993	0.000854
$\pi_2(10)$	0.839391	0.838905	0.000486
$\pi_3(10)$	0.813851	0.813453	0.000398
$\pi_4(10)$	0.781973	0.781613	0.000360
ρ	0.442225	0.442271	-0.000046
$\pi_5(0)$	0.999874	0.999385	0.000489

Table 2. Calculation results for k-busy periods and the traffic coefficient for the system from Example 3.2.1

Example 3.2.2. *Consider the system $M_{10}|M_{10}|1$ with all interarrival times being distributed exponentially Exp(a_k), $k = 1, ..., 10$, and all*

service times being distributed exponentially Exp(200). The switchover times C_k are all distributed as Exp(100), $k = 1, \ldots, 10$. The results of calculations for such systems can be found in Tables 3, 4. The quantity ϵ is taken to be 0.001. *In this case:*

$$\beta_k(s) = \frac{1}{s\mathbb{E}[B_k] + 1},$$

$$c_k(s) = \frac{1}{s\mathbb{E}[C_k] + 1},$$

$$k = 1, \ldots, 10.$$

a_k	$\rho(BPLST_E)$	$\rho(BPLST_\epsilon)$	*difference*
1	0.064175	0.064175	0.000000
10	1.806396	1.807565	-0.001169
100	7213724.500000	7212584.000000	1140.500000

Table 3. Calculation results for the traffic coefficient for the systems from Examples 3.2.2 (scheme "repeat again").

a_k	$\rho(BPLST_E)$	$\rho(BPLST_\epsilon)$	*difference*
1	0.062948	0.062948	0.000000
10	1.375752	1.376590	-0.000838
100	1323211.375000	1322987.875000	223.500000

Table 4. Calculation results for the traffic coefficient for the systems from Example 3.2.2 (scheme "loss").

Example 3.2.3. *Consider the system $M_{10}|G_{10}|1$ with all interarrival times being distributed exponentially Exp(a_k), $k = 1, ..., 10$, and all service times being distributed Er$(3, 200)$. The switchover times C_k are all distributed as Exp(100), $k = 1, \ldots, 10$. The results of calculations*

for such systems can be found in Tables 5, 6. The quantity ϵ *is taken to be* 0.001*. In this case:*

$$\beta_k(s) = \left(\frac{1}{s\mathbb{E}[B_k]+1}\right)^3,$$

$$c_k(s) = \frac{1}{s\mathbb{E}[C_k]+1},$$

$$k = 1, \ldots, 10.$$

a_k	$\rho(BPLST_E)$	$\rho(BPLST_\epsilon)$	*difference*
1	0.176818	0.176821	-0.000003
10	12.101274	12.102625	-0.001351
100	348237984.000000	348191168.000000	46816.000000

Table 5. Calculation results for the traffic coefficient for the systems from Example 3.2.3 (scheme "repeat again").

a_k	$\rho(BPLST_E)$	$\rho(BPLST_\epsilon)$	*difference*
1	0.165731	0.165734	-0.000003
10	4.723875	4.724642	-0.000767
100	2153345.5000000	2153149.750000	195,750000

Table 6. Calculation results for the traffic coefficient for the systems from Example 3.2.3 (scheme "loss").

Example 3.2.4. *Consider the system* $M_{10}|G_{10}|1$ *with interarrival times being all distributed as Exp(10) and with the times of service of the requests from the lines* L_1*,* L_2 *and* L_3 *being distributed exponentially Exp(200), from the lines* L_4*,* L_5 *and* L_6 *being distributed uniformly* $U[0,1]$*, and from the lines* L_7*,* L_8*,* L_9*,* L_{10} *being distributed as*

$Er(3, 200)$. *The switchover times* C_k *are all distributed as Exp(100),* $k = 1, \ldots, 10$. *The quantity* ϵ *is taken to be* 0.001.

In this case:

$$\beta_k(s) = \frac{1}{s\mathbb{E}[B_k] + 1},\ k = 1, 2, 3;$$

$$\beta_k(s) = \frac{e^{-as} - e^{-bs}}{b - a},\ a = 0,\ b = 1,\ k = 4, 5, 6;$$

$$\beta_k(s) = \left(\frac{1}{s\mathbb{E}[B_k] + 1}\right)^3,\ k = 7, 8, 9, 10;$$

$$c_k(s) = \frac{1}{s\mathbb{E}[C_k] + 1},\ k = 1, \ldots, 10.$$

scheme	$\rho(BPLST_E)$	$\rho(BPLST_\epsilon)$	*difference*
repeat again	5.184965	5.188527	-0.003562
loss	2.020857	2.022199	-0.001342

Table 7. Calculation results for the traffic coefficient for the systems from Example 3.2.4.

4. Conclusions

We presented a model algorithm of the numerical evaluation of the traffic coefficient in priority queueing systems (Algorithm 3 WLCOEFF). This algorithm makes use of the LST of busy period of the system—this should also be calculated numerically, using algorithms similar to the model Algorithm 1 $BPLTS_E$ and Algorithm 2 $BPLTS_\epsilon$. However, it was found from our experience that (i) the number of priority flows r should not exceed 10-12 for satisfactory fast calculations, and (ii) the calculation of the LST periods with algorithm Algorithm 1 $BPLTS_E$ was performed without clear idea about the absolute error of the evaluation. Therefore, there was a necessity of further optimization of this

numerical algorithm in order to achieve fast performance and high level of precision of calculations. Algorithm 2 $BPLST_\epsilon$ served this purpose. However, one should also notice that for some systems the algorithm $BPLTS_E$ calculates the busy period's LST with the same precision as $BPLTS_\epsilon$ does (throughout the considered examples for the systems for which 'difference' in tables is of the same order as E and ϵ).

There is a necessity of further optimization in order to consider greater number of priority waiting lines. Such work is being done currently [3, 4, 6].

References

[1] Klimov, G. P., Mishkoy, G. K. 1979. *Prioritetnye sistemy obsluzhivaniya s orientatsiei* (Priority Queueing Systems with Switchover Times). Moscow University Press. In Russian.

[2] Mishkoy, Gh., Giordano, S., Bejan, A., Benderschi, O. 2007. Priority queueing systems with switchover times: generalized models for QoS and CoS network technologies. *Comput. Sci. J. Moldova* nr. 2, vol. 15(44) 217–242.

[3] Mishkoy, Gh. K., Rykov, V. V., Giordano, S., Bejan, A. Iu. 2008. Multidimensional analogs of the Kendall equation for priority queueing systems: computational aspects. *Automation and Remote Control* N.5, Moscow, pp. 82–95.
WEB: `www.vitrum.md/andrew/A&T_eng_tb_published_2008.pdf`

[4] Bejan, A. 2007. Switchover time modelling in priority queueing systems. *PhD thesis*. State University of Moldova.

[5] Bejan, A. 2006. Numerical treatment of the Kendall equation in the analysis of priority queueing systems. *Bul. Acad. Ştiinţe Repub. Mold. Mat.* **51**(2) 17–28.

[6] Mishkoy, G., Giordano, S., Andronati, N., Bejan, A. 2006. Priority queueing systems with switchover times: Generalized Models

for QoS and CoS Network Technologies and Analysis. *Technical report*.
WEB: http://www.vitrum.md/andrew/PQSST.pdf

Gh. Mishkoy, A. Iu. Bejan, O. Benderschi, Received January 30, 2008

Gheorghe Mishkoy
Academy of Sciences of Moldova,
Free International University of Moldova
E–mail: *gmiscoi@ulim.md*

Andrei Iu. Bejan
Institute of Mathematics and Computer Science,
Academy of Sciences of Moldova,
Heriot-Watt University and
the Maxwell Institute for Mathematical Sciences,
Edinburgh, Scotland, UK
E–mail: *A.I.Bejan@ma.hw.ac.uk*

Olga Benderschi
State University of Moldova
E–mail: *obenderschi@yahoo.com*

Computer Science Journal of Moldova, vol.**16**, no.2(47), 2008

On stability of a Pareto-optimal solution under perturbations of the parameters for a multicriteria combinatorial partition problem

V.A. Emelichev, E.E. Gurevsky

Abstract

Abstract. We consider a multicriteria variant for the well-known partition problem. A formula of the stability radius for an efficient solution was obtained.

Mathematics Subject Classification 2000: 90C27, 90C29, 90C31.

Keywords and phrases: multiobjectivity, partition problem, Pareto set, efficient solution, stability, stability radius of an efficient solution.

Many problems of design, planning and management in technical and organizational systems have a pronounced multicriteria character. Multiobjective models that appeared in these cases are reduced to the choice of the "best" (in a certain sense) values of variable parameters from some discrete aggregate of the given quantities. Therefore recent interest of mathematicians to multicriteria discrete optimization problems keeps very high which is confirmed by the intensive publishing activity (see, e. g., bibliography [1], which contains 234 references). One of the important directions of study such problems is stability analysis of solutions under perturbations of the initial data. Various questions of stability analysis and regularization for incorrect discrete optimization problems generate numerous directions of research. Nowadays owing to the fundamental investigations of academician I. V. Sergienko and his colleagues [2–11], characteristics of stable problems, necessary and sufficient conditions of existence of Pareto-optimal solutions, methods of

regularization for incorrect problems and many other properties of multicriteria integer problems under uncertainty of initial data are rather well-known.

Nowadays many specialists study quantitative characteristics of stability of both scalar (single criteria) and vector (multicriteria) discrete optimization problems. Not touching on this wide spectrum of questions, we refer the reader to the extensive bibliography [12] as well as to the works [13–18] which contain the most typical recent results.

We continue to research the cycle of works devoted to the quantitative analysis of stability [19–26] for the Pareto-optimal solutions of the combinatorial problems with various types of vector criteria. We consider a multicriteria variant of the well-known partition problem. A formula of the stability radius of an efficient solution in the case of l_∞-metric is obtained.

The partition problem is a classical extremal combinatorial problem. It is stated as follows: it is needed to partition the finite set of numbers into two nonintersecting subsets such that the sums of numbers of these subsets are differed minimally from each other. In the case where the elements of the set are positive this problem is equivalent to the problem of the scheduling theory that consists in distribution of independent works in two identical processors such that to minimize the time when the last work should be finished [27]. In the scheduling theory this problem is marked as $P|\cdot|C_{\max}$.

We consider a multicriteria (vector) variant of the partition problem.

We define a vector function (vector criterion)

$$f(x, C) = (|C_1 x|, |C_2 x|, \ldots, |C_m x|) \to \min_{x \in Q^n}$$

on the set of n-vectors $Q^n,\ n \geq 2,\ Q = \{-1, 1\}$, where C_i denotes the i-th row of matrix $C = [c_{ij}]_{m \times n} \in \mathbf{R}^{m \times n},\ m \geq 1, x = (x_1, x_2, \ldots, x_n)^T$.

Under the multicriteria partition problem $Z^m(C)$ we understand the problem of finding the set of efficient solutions (Pareto set)

$$P^m(C) = \{x \in Q^n : \pi(x, C) = \emptyset\},$$

where

$$\pi(x, C) = \{x' \in Q^n : f(x, C) \geq f(x', C) \ \& \ f(x, C) \neq f(x', C)\}.$$

Under the stability of the efficient solution x^0 we understand the property of preserving Pareto optimality of x^0 under "small" perturbations of the elements of matrix C. We will model such perturbations by adding a "perturbing" matrices to C.

For each number $k \in \mathbf{N}$, we endow the space $\mathbf{R}^k$ with metrics l_1 and l_∞ :

$$||z||_1 = \sum_{j \in N_k} |z_j|, \ ||z||_\infty = \max_{j \in N_k} |z_j|, \ z = (z_1, z_2, \ldots, z_k) \in \mathbf{R}^k,$$

where $N_k = \{1, 2, \ldots, k\}$. Under the norm of matrix we understand the norm of vector composed from all its elements. For any number $\varepsilon > 0$ we define the set of perturbing matrices

$$\Omega(\varepsilon) = \{C' \in \mathbf{R}^{m \times n} : ||C'||_\infty < \varepsilon\}.$$

According to the definitions from [21–26], under the stability radius of $x^0 \in P^m(C)$ we understand the number

$$\rho^m(x^0, C) = \begin{cases} \sup \ \Xi, & \text{if } \Xi \neq \emptyset, \\ 0, & \text{if } \Xi = \emptyset, \end{cases}$$

where

$$\Xi = \{\varepsilon > 0 : \forall \ C' \in \Omega(\varepsilon) \ (x^0 \in P^m(C + C'))\}.$$

Thus, the stability radius is a limit level of independent perturbations of the elements of C, such that the Pareto optimality of the solution is preserved.

We will use the following implication

$$\exists q \in Q \ \ \forall q' \in Q \ \ (qz > q'z') \Rightarrow |z| > |z'|, \tag{1}$$

which holds for any numbers $z, z' \in \mathbf{R}$.

Suppose

$$\operatorname{sg} z = \begin{cases} 1, & \text{if } z \geq 0, \\ -1, & \text{if } z < 0, \end{cases}$$

$$K(x^0, x) = \{i \in N_m : \ |C_i x^0| \leq |C_i x|\},$$

$$\alpha_i(x^0, x) = \min\{\beta_i(x^0, x, q) : \ q \in Q\},$$

$$\beta_i(x^0, x, q) = \frac{|C_i(qx^0 + x)|}{||qx^0 + x||_1}.$$

It is evident, that $K(x^0, x) \neq \emptyset$ if $x^0 \in P^m(C)$.

Theorem. *Stability radius of an efficient solution x^0 of the problem $Z^m(C),\ m \geq 1$ is expressed by the formula*

$$\rho^m(x^0, C) = \min_{x \in Q^n \setminus \{x^0, -x^0\}} \max_{i \in K(x^0, x)} \alpha_i(x^0, x). \tag{2}$$

Proof. Denote by φ the right side of (2). It is easy to see, that $\varphi \geq 0$.

At first we will prove the inequality $\rho^m(x^0, C) \geq \varphi$. Suppose $\varphi > 0$ (otherwise the inequality $\rho^m(x^0, C) \geq \varphi$ is evident). Let $C' \in \Omega(\varphi)$. Then by the definition of φ for any $x \in Q^n \setminus \{x^0, -x^0\}$ there exists $k \in K(x^0, x)$ such that

$$||C'||_\infty < \varphi \leq \alpha_k(x^0, x). \tag{3}$$

Taking into account $\alpha_k(x^0, x) > 0$, we have

$$|C_k x^0| < |C_k x|.$$

From this, assuming

$$\sigma_k = \operatorname{sg} C_k x,$$

we obtain

$$C_k(qx^0 + \sigma_k x) = |C_k(\sigma_k q x^0 + x)|, \ q \in Q.$$

Therefore, using (3), we derive

$$(C_k + C'_k)(qx^0 + \sigma_k x) = |C_k(\sigma_k q x^0 + x)| + C'_k \sigma_k (\sigma_k q x^0 + x) \geq$$
$$\geq |C_k(\sigma_k q x^0 + x)| - ||C'||_\infty \cdot ||\sigma_k q x^0 + x||_1 >$$

$$> |C_k(\sigma_k q x^0 + x)| - \beta_k(x^0, x, \sigma_k q)||\sigma_k q x^0 + x||_1 = 0.$$

Thus, we have

$$(C_k + C'_k)\sigma_k x > (C_k + C'_k)q x^0,\ q \in Q.$$

Taking into account (1) for any $x \in Q^n \setminus \{x^0, -x^0\}$ we obtain

$$|(C_k + C'_k)x| > |(C_k + C'_k)x^0|,$$

and for $x = \pm x^0$ we have

$$|(C + C')x| = |(C + C')x^0|,$$

which imply $x^0 \in P^m(C + C')$.

Resuming the said above, we conclude that for any $C' \in \Omega(\varphi)$ the inclusion $x^0 \in P^m(C + C')$ holds. Hence $\rho^m(x^0, C) \geq \varphi$.

It remains to prove the inequality $\rho^m(x^0, C) \leq \varphi$. By the definition of φ, there exists $x^* \in Q^n \setminus \{x^0, -x^0\}$, such that for any $i \in K(x^0, x^*)$ the following inequalities hold:

$$0 \leq \alpha_i(x^0, x^*) \leq \varphi. \tag{4}$$

Let $\varepsilon > \varphi$. We will prove that there exists $C' \in \Omega(\varepsilon)$ with condition $x^0 \notin P^m(C + C')$.

Suppose

$$N(x^0, x^*) = |\{j \in N_n :\ x^0_j = 1\ \&\ x^*_j = -1\}|,$$

$$M(x^0, x^*) = |\{j \in N_n :\ x^0_j = x^*_j\}|,$$

$$\sigma^*_i = \operatorname{sg} C_i x^*.$$

It is easy to see, that

$$M(x^0, x^*) = M(x^*, x^0),$$

$$2(N(x^0, x^*) + N(x^*, x^0)) = ||x^0 - x^*||_1, \tag{5}$$

$$2M(x^0, x^*) = ||x^0 + x^*||_1. \tag{6}$$

To construct the rows $C'_i,\ i \in N_m$ of the needed matrix C', we consider four possible cases.

Case 1: $i \in K(x^0, x^*),\ \beta_i(x^0, x^*, -1) < \beta_i(x^0, x^*, 1)$. Then under (4) the following inequalities hold:

$$|C_i(x^0 + x^*)| > 0,$$

$$\beta_i(x^0, x^*, -1) \le \varphi < \varepsilon.$$

Therefore if we consider a perturbing row

$$C'_i = (c'_{i1}, c'_{i2}, \ldots, c'_{in}),$$

obtained by setting

$$c'_{ij} = \begin{cases} \sigma_i^* \delta_i, & \text{if } x_j^0 = 1,\ x_j^* = -1, \\ -\sigma_i^* \delta_i, & \text{if } x_j^0 = -1,\ x_j^* = 1, \\ 0 & \text{in other cases,} \end{cases}$$

$$\varphi < \delta_i < \varepsilon,$$

then we have $||C'_i||_\infty = \delta_i$, and taking into account (5), we derive

$$\sigma_i^*(C_i + C'_i)x^0 - \sigma_i^*(C_i + C'_i)x^* =$$

$$= \sigma_i^* C_i(x^0 - x^*) + 2\delta_i(N(x^*, x^0) + N(x^0, x^*)) \ge$$

$$\ge -|C_i(x^0 - x^*)| + \delta_i||x^0 - x^*||_1 >$$

$$> -|C_i(x^0 - x^*)| + \beta_i(x^0, x^*, -1)||x^0 - x^*||_1 = 0,$$

$$\sigma_i^*(C_i + C'_i)x^0 + \sigma_i^*(C_i + C'_i)x^* = \sigma_i^* C_i(x^0 + x^*) =$$

$$= |C_i(x^0 + x^*)| > 0.$$

Therefore we obtain

$$\sigma_i^*(C_i + C'_i)x^0 > \sigma_i^*(C_i + C'_i)qx^*,\ q \in Q.$$

From this, using (1), we find

$$|(C_i + C'_i)x^0| > |(C_i + C'_i)x^*|. \tag{7}$$

Note, that the inequality (7) is coordinated with condition $x^* \in Q^n \backslash \{x^0, -x^0\}$.

Case 2: $i \in K(x^0, x^*)$, $\beta_i(x^0, x^*, -1) > \beta_i(x^0, x^*, 1)$. Then under (4) we have

$$|C_i(x^* - x^0)| > 0,$$

$$\beta_i(x^0, x^*, 1) \leq \varphi < \varepsilon.$$

Therefore, constructing the row C'_i by the rule

$$c'_{ij} = \begin{cases} -\sigma_i^* \delta_i, & \text{if } x_j^0 = x_j^* = 1, \\ \sigma_i^* \delta_i, & \text{if } x_j^0 = x_j^* = -1, \\ 0 & \text{in the other cases,} \end{cases}$$

where $\varphi < \delta_i < \varepsilon$, we obtain $||C'_i||_\infty = \delta_i$ and, using (6), we derive

$$-\sigma_i^*(C_i + C'_i)x^0 - \sigma_i^*(C_i + C'_i)x^* = -\sigma_i^* C_i(x^0 + x^*) + 2\delta_i M(x^0, x^*) >$$

$$> -|C_i(x^0 + x^*)| + \beta_i(x^0, x^*, 1)||x^0 + x^*||_1 = 0,$$

$$-\sigma_i^*(C_i + C'_i)x^0 + \sigma_i^*(C_i + C'_i)x^* = \sigma_i^* C_i(x^* - x^0) = |C_i(x^* - x^0)| > 0.$$

Thus, the following inequalities hold:

$$-\sigma_i^*(C_i + C'_i)x^0 > \sigma_i^*(C_i + C'_i)qx^*, \ q \in Q.$$

Therefore, using (1), we obtain (7).

Case 3: $i \in K(x^0, x^*)$, $\beta_i := \beta_i(x^0, x^*, -1) = \beta_i(x^0, x^*, 1) = \alpha_i(x^0, x^*)$.

Consider two possible variants.

At first let $\beta_i = 0$. Then

$$C_i x^0 = C_i x^* = 0. \tag{8}$$

It is easy to see, that taking into account $x^* \neq \pm x^0$, we may choose $k, p \in N_n$ such that

$$x_k^* = x_k^0, \quad x_p^* \neq x_p^0.$$

Therefore, if we define the elements of the row $C'_i = (c'_{i1}, c'_{i2}, \ldots, c'_{in})$ by

$$c'_{ij} = \begin{cases} x^0_k \delta_i, & \text{if } j = k, \\ x^0_p \delta_i, & \text{if } j = p, \\ 0 & \text{in other cases ,} \end{cases}$$

where

$$0 \leq \varphi < \delta_i < \varepsilon,$$

then we have, that $||C'_i||_\infty = \delta_i$ and under (8) the inequality (7) holds.

Now let $\beta_i > 0$. Then, repeating all the argumentations from case 1, we obtain (7).

Case 4: $i \in N_m \backslash K(x^0, x^*)$. Then, assuming $C'_i = (0, 0, \ldots, 0) \in \mathbf{R}^n$, we have (7).

So we obtain the matrix C' with norm

$$||C'||_\infty = \max\{\delta_i : \ i \in N_m \setminus K(x^0, x^*)\} < \varepsilon.$$

Summarizing what has been proven in four cases we see, that for any $\varepsilon > \varphi$ there exists $C' \in \Omega(\varepsilon)$ such that $x^0 \notin P^m(C + C')$. Hence, $\rho^m(x^0, C) \leq \varphi$.

Theorem is proved.

Remark 1. If we impose on $C = [c_{ij}] \in \mathbf{R}^{m \times n}$ the condition of preserving positivity for all its elements during perturbing, then the stability radius of $x^0 \in P^m(C)$ is equal to

$$\min\{\varphi, c_{\min}\},$$

where φ is the right side of (2), $c_{\min} = \min\{c_{ij} : \ (i, j) \in N_m \times N_n\}$.

Efficient solution x^0 is called stable if $\rho^m(x^0, C) > 0$ and special if the following condition holds:

$$\nexists \ x \in Q^n \backslash \{x^0, -x^0\} \quad (f(x, C) \leq f(x^0, C)).$$

The following statement follows directly from the theorem.

Corollary. *Solution $x^0 \in P^m(C)$ is stable if and only if it is special.*

Remark 2. As a rule (see, e. g., [22,26]) the strict efficiency (Smale optimality [28]) of a solution of a multicriteria discrete optimization

problem is a sufficient condition of the solution stability. But, it is easy to see, that our problem does not have strictly efficient solutions. Nevertheless efficient solutions can be stable (see corollary).

References

[1] Ehrgott M., Gandibleux X. *A survey and annotated bibliography of multiobjective combinatorial optimization.* OR Spectrum, 2000, **22**, N 4, p. 425–460.

[2] Sergienko I.V. *Mathematical models and decision methods of discrete optimization problems.* Kiev, Naukova dumka, 1988 (in Russian).

[3] Sergienko I.V., Kozeratskaya L.N., Lebedeva T.T. *Stability investigation and parametric analysis of discrete optimization problems.* Kiev, Naukova dumka, 1995 (in Russian).

[4] Sergienko I.V., Shilo V.P. *Discrete optimization problems. Challenges, decision methods, investigations.* Kiev, Naukova dumka, 2003 (in Russian).

[5] Kozeratskaya L.N., Lebedeva T.T., Sergienko T.I. *Integer programming problems with a vector criterion. Parametric analysis and investigation of stability.* Soviet Mathematics Doklady, 1990, **40**, N 1, p. 98–100.

[6] Kozeratskaya L.N., Lebedeva T.T., Sergienko I.V. *Stability of discrete optimization problems.* Cybernetics and Systems Analysis, 1993, **29**, N 3, p. 367–378.

[7] Kozeratskaya L.N., Lebedeva T.T., Sergienko T.I. *Regularization of integer vector optimization problems.* Cybernetics and Systems Analysis, 1993, **29**, N 3, p. 455–458.

[8] Sergienko I.V., Lebedeva T.T., Semenova N.V. *Existence of solutions in vector optimization problems.* Cybernetics and Systems Analysis, 2000, **36**, N 6, p. 823–828.

[9] Lebedeva T.T., Sergienko T.I. *Comparative analysis of different types of stability with respect to constraints of a vector integer-optimization problem.* Cybernetics and Systems Analysis, 2004, **40**, N 1, p. 52–57.

[10] Lebedeva T.T., Semenova N.V., Sergienko T.I. *Stability of vector integer optimization problems with quadratic criterion functions.* Theory of Stochastic Processes, 2004, **10(26)**, N 3–4, p. 95–101.

[11] Lebedeva T.T., Semenova N.V., Sergienko T.I. *Stability of vector problems of integer optimization: relationship with the stability of sets of optimal and nonoptimal solutions.* Cybernetics and Systems Analysis, 2005, **41**, N 4, p. 551–558.

[12] Greenberg H.J. *An annotated bibliography for post-solution analysis in mixed integer and combinatorial optimization.* In: D.L. Woodruff, ed. Advances in Computational and Stochastic Optimization, Logic Programming and Heuristic Search, Boston, Kluwer Academic Publishers, 1998, p. 97–148.

[13] Sotskov Yu.N., Leontev V.K., Gordeev E.N. *Some concepts of stability analysis in combinatorial optimization.* Discrete Applied Mathematics. 1995, **58**, N 2, p. 169–190.

[14] Chakravarti N., Wagelmans A. *Calculation of stability radius for combinatorial optimization.* Operations Research Letters, 1998, **23**, N 1, p. 1–7.

[15] Libura M., van der Poort E.S., Sierksma G., van der Veen J.A.A. *Stability aspects of the traveling salesman problem based on k-best solutions.* Discrete Applied Mathematics, 1998, **87**, N 1–3, p. 159–185.

[16] Gordeev E.N. *Stability analysis in optimization problems on matroids in the metric l_1.* Cybernetics and Systems Analysis, 2001, **37**, N. 2, p. 251–259.

[17] Emelichev V.A., Girlich E., Nikulin Yu.V., Podkopaev D.P. *Stability and regularization of vector problems of integer linear programming.* Optimization, 2002, **51**, N 4, p. 645–676.

[18] Libura M., Nikulin Y. *Stability and accuracy functions in multicriteria combinatorial optimization problem with $\sum$–MINMAX and $\sum$–MINMIN partial criteria.* Control and Cybernetics, 2004, **33**, N 3, p. 511–524.

[19] Emelichev V.A., Krichko V.N. *Stability radius of an efficient solution to a vector quadratic problem of Boolean programming.* Computational Mathematics and Mathematical Physics, 2001, **41**, N 2, p. 322–326.

[20] Bukhtoyarov S.E., Emelichev V.A., Stepanishina Y.V. *Stability of discrete vector problems with the parametric principle of optimality.* Cybernetics and Systems Analysis, 2003, **39**, N 4, p. 604–614.

[21] Emelichev V.A., Krichko V.N., Nikulin Y.V. *The stability radius of an efficient solution in minimax Boolean programming problem.* Control and Cybernetics, 2004, **33**, N 1, p. 127–132.

[22] Emelichev V.A., Kuz'min K.G., Leonovich A.M. *Stability in the combinatorial vector optimization problems.* Automation and Remote Control, 2004, **65**, N 2, p. 227–240.

[23] Emelichev V.A., Kuz'min K.G. *Stability analysis of a strictly efficient solution of a vector problem of Boolean programming in the metric l_1.* Discrete Mathematics and Applications, 2004, **14**, N 5, p. 521–526.

[24] Emelichev V.A., Kuz'min K.G. *Stability radius for a strictly effective solution to a vector minimization problem for threshold functions in l_1 metric.* Cybernetics and Systems Analysis, 2004, **40**, N 3, p. 358–362.

[25] Emelichev V.A., Kuz'min K.G. *The stability radius of an efficient solution to a vector problem of Boolean programming in the l_1 metric.* Doklady Mathematics, 2005, **71**, N 2, p. 266–268.

[26] Emelichev V.A., Kuzmin K.G., Nikulin Yu.V. *Stability analysis of the Pareto optimal solution for some vector Boolean optimization problem.* Optimization, 2005, **54**, N 6, p. 545–561.

[27] Lawler E.L., Lenstra J.K., Rinnoy Kan A.H.G., Shmoys D.B. *Sequencing and scheduling: algorithms and complexity.* Handbooks in Operations Research and Management Science. Volume 4. Logistics of Production and Inventory, 1993, p. 445–522.

[28] Smale S. *Global analysis and economics V: Pareto theory with constraints.* Journal of Mathematical Economics, 1974, **1**, N 3, p. 213–221.

V.A. Emelichev, E.E. Gurevsky,

Belarusian State University,
av. Nezavisimosti, 4,
220030 Minsk, Belarus.
E–mail: *emelichev@bsu.by*

Received April 22, 2008

Computer Science Journal of Moldova, vol.**16**, no.2(47), 2008

Summary of researches being performed in the Institute of Mathematics and Computer Science on computer science and information technologies

Artiom Alhazov, Elena Boian, Liudmila Burtseva,
Constantin Ciubotaru, Svetlana Cojocaru,
Alexandru Colesnicov, Constantin Gaindric, Galina Magariu,
Iurie Rogojin, Tatiana Verlan

Evolution of the informatization notion (which assumes automation of majority of human activities applying computers, computer networks, information technologies) towards the notion of *Global Information Society* (GIS) challenges the determination of new paradigms of society: automation and intellectualization of production, new level of education and teaching, formation of new styles of work, active participation in decision making, etc.

To assure transition to GIS for any society, including that from Republic of Moldova, requires both special training and broad application of progressive technologies and information systems. Methodological aspects concerning impact of GIS creation over the citizen, economic unit, national economy in the aggregate demands a profound study. Without systematic approach to these aspects the GIS creation would have confront great difficulties.

Collective of researchers from the Institute of Mathematics and Computer Science (IMCS) of Academy of Sciences of Moldova, which work in the field of computer science, constitutes the center of advanced researches and activates in those directions of researches of computer science which facilitate technologies and applications without of which the development of GIS cannot be assured. The works of the collective

deal with methodological aspects of GIS development. The advanced technologies and systems which serve as scientific support for transition to GIS for such domains as health protection, education, economy, research, ecology etc are elaborated and developed in these works. Let us enumerate the domains of interest: actual problems of information society development, information decision support systems, theoretical bases of information systems, formal calculation models, information systems in computational algebra, and information systems in computational linguistics.

The problems of GIS creation are constantly in the view of European Council, one of the last recommendations of which is ,,i2010 eGovernment Action Plan - Accelerating eGovernment in Europe for the Benefit of All – (25/04/2006)". In this document there is an attempt to synchronize the efforts of European countries in electronic government, education, and keeping and protection of multi-linguism.

The access to cultural patrimony of the own country and of other countries in his native language is to be guaranteed for each citizen from every European country so as he would be able to understand and appreciate the value and immensity of this patrimony as an integrant of European patrimony. Each country has its own specifics, which compels search of proper ways of transition to GIS on the assumption of its own economic, cultural potential, nation's traditions etc.

Different aspects of these problems were examined in the works [13, 71, 100, 102, 103, 109, 129].

The book "Considerations on information society development in Moldova" [12] was the first work in Moldova in which a complex approach to the problem **in** and **for** this country is presented. The conceptual bases, principles, policies and main directions in GIS creation were elucidated in it including electronic education, science and culture.

In other works there were examined the problems of evaluation of degree of integration into GIS, proposing a set of integration indices which highlight the specifics of our country. Internet in rural environment and digital divide overcoming constitute the subjects of another works [122, 130, 146].

A group of researchers on computer science from IMCS was invited to take part in writing the collective monograph *"Information technologies communication and human development: Opportunities and challenges"*, published by Idea Group Inc.(USA), in which two chapters of these researchers are included: one is devoted to the general problems of digital inequality and another – the same problem for Moldova [8,9]. These chapters have been selected for inclusion in exclusive, authoritative reference publication ,,Information Communication Technologies: Concepts, Methodologies, Tools and Applications" published by IGI Global (formerly "Idea Group Inc.") in 2008. This six-volume set includes the most dependable, exhaustive research on the subject.

Some proposals based on research results of this group were included into National Strategy of information society construction ,,Electronic Moldova" and Actions Plan adopted by Government (Decree N 255 from 9.03.2005).

When elaborating applicable information systems the experience of the researchers from IMCS accumulated in the framework of design and automation of programming systems construction had favorable impact [11,72,76,78,82,88,98,121]. The elaborated and utilized methods when developing programming systems (formal grammars, automata, compiling mechanism, text editors) were subsequently used successfully for elaboration of interfaces and tools for database management, linguistic applications.

In the knowledge based society it is just *decision support systems* (DSS) that become very useful and claimed information tools for the factors of decision making of any level [89, 117]. In the monograph [12] "Decision taking. Methods and technologies" there are explicated new approaches to the process of decision making based on information technologies.

Some works [21, 93, 96] are devoted to the researches and elaborations of some DSS used for vehicle transport. The actual problem for any society, including that to which we are aspiring, is distribution of some always limited funds to the set of activities and projects so as to achieve maximal efficiency. The works [75, 110] are dedicated to the elaboration of DSS for optimal funds allotment.

Medical assistance being one of the first application subjects of DSS henceforth represents a perspective area for such applications. About 9% of European Union gross domestic product (GDP) is apportioned for population health support. Raising of the level of diagnostics and adequate medical treatment is the goal of the DSS at ultrasound investigations [62, 108, 132, 148, 149, 158] made by the researchers from IMCS. The goal of this system is to help medic-echographist in the process of ultrasonographic investigations, in obtained images interpretation, in obtaining conclusions adequate to the patient's state of the health and finally to prescribe an adequate treatment.

In the process of investigation of systems or complex phenomena it is necessary for correct decisions making to base on the knowledge of systems behavior under their interaction with certain external factors. Frequently the experiments accomplishment with real systems involving is impossible, because the real system does not exist yet; exists but is not accessible for real experiments; the experiments with real system are expensive or dangerous. In such situations the simulation model which is exposed to experiments is being created. This permits to economize resources, avoid unreasoned expenses, reduce designing periods, minimize risks and to avoid possible catastrophic situations.

The works [76, 87, 90, 111, 114] present the research results which led to the elaboration of *integrated environment for simulation*, which includes possibilities to visualize and analyze experiments results. In articles [91, 113, 126] the experience of simulation application to solution of some concrete problems and to create demonstrative models with training purpose is explicated.

Presence of convenient facilities for experiments results analysis in the case of mathematical models application is as important as in the case of simulation. This permits to find regularities in the behavior of investigated system and to interpret correctly the experiment results.

The solution of ecological problems continues to remain an extremely actual subject for Republic of Moldova. At the same time just in this domain the possibilities offered by information technologies had not been turned to advantages in full measure yet. An approach in this direction is made in the works [128, 131, 150, 151], which explicate

the results of environment creation for analysis of experiments results for mathematical model of physical and biochemical processes of solid waste decomposition.

Such approach was applied when elaborating information-analytical system "Scientific Potential of Moldova" [127], endowed with possibilities to administrate and analyze information about researchers and scientific centers of the country.

One of the top-priority requirements of social development at the stage of transition to GIS is the unhampered access to modern information technologies. To provide all members of community with adequate access to Internet resources, to technologies of information search and extraction, speech perception, text insonification, automatic translation etc, it is necessary the development of national linguistic engineering, which gives technological resources for natural language processing and provides with full-fledged access to Internet. Lack of such technologies at national level hampers our country to join global information resources.

The human-computer communication in *natural language* is a large-scale problem for which it is recognized that still during a long time we will be able only to approach to its solving. The contribution of the collective of IMCS to the development of the technology for natural language processing is the essential segment. The results in this direction can be classified into three domains.

Linguistic tools. There were developed encoders and converters for text and graphical information, programs for automatic word inflection, programs for word division into syllables, facilities for linguistic resources administration, electronic multifunctional dictionaries creation, for correctness and completeness of linguistic resources check-up, and also visualization means, interfaces and support for elaboration of adaptable computer-aided learning courses [44, 83, 84, 86, 101, 106, 112, 119, 120, 144, 145].

Linguistic resources. There were created reusable linguistic resources (http://imi201.math.md/elrr) for natural language, which include annotated corpus for all parts of speech and kernel, containing

circa 70000 basic words-lemmas (entries), accompanied by syntactical and morphological information. This kernel generates computer lexicon, gives the list of all word inflexions for basic word, executes translations into English and Russian languages, posts list of synonyms. General number of word inflexions generated on the basis of lexicon constitutes circa 700 thousands of words [58, 68, 107, 115, 116, 124,154].

Linguistic applications. The spelling checker RomSP was elaborated and integrated into text editor MS Word. It allows finding and correcting erroneous words, to give prompting list, gives the possibility to supplement own word base using the tools for automatic word inflection. In collaboration with editorial group LITERA the compact disk with spelling checker RomSP with linguistic base of about 1 000 000 words had been published.

Also on the bases of developed linguistic resources there were created electronic dictionaries of synonyms, Romanian-English and Romanian-Russian translations, and adaptable system of computer-aided learning course [45, 74, 81, 155, 157].

Elaboration of computation lexicon, accumulation of linguistic resources in electronic format allows making a forward step in natural language application to realization of interfaces for information systems for the purpose of assuring of friendly mode of work for the user [10,80].

Information technologies became tools that influence and propel researches in other domains of science. A set of works [1, 4, 7, 16, 38, 47, 79, 85, 99, 104, 123, 147] presents the results of researches in *development of a symbolic computation system.*

Researches of the group of authors from IMCS had been centered towards two directions: extension of calculating capacity of SCS and creation of intelligent interfaces. There were developed architectural principles of intelligent interfaces construction for SCS, the application of which facilitates "human-computer" interaction, permits extension of the circle of users. Such interfaces possess capacities to adapt to users' needs, to train; they are able to take the initiative in communication with the user (doing this in natural language or in a subset of that) guiding him with the purpose to facilitate

reaching his objectives in a way as rapid and comfortable as possible. Some of these principles were applied during implementation of the Computer Algebra System Bergman interface (elaborated by Jörgen Backelin, Stockholm University, and improved in collaboration with IMCS)(http://www.math.su.se/bergman) assuring overtaking the problem from user, its adaptation to the user's preferences (executed in three modes: a priori, on user's initiative and on system's initiative), errors prevention, creating of calculation environment – by all of this in ultimate authority contributing to create for any user apart his personal entourage oriented to the class of problems which constitute his occupations.

The explosion of demand for information technologies faces exhaustion of capabilities of conventional computers. This is why research started in the last years to find possibilities to increase computer performance using new, non-conventional approaches. Starting in 1994 the domain of *biocomputing* appeared, in which the contribution of researchers from IMCS was carried out in two directions: research of the formal computational models and creating algorithms of solving computationally difficult problems applying models of biocomputing.

Research in theoretical foundations of informational systems and formal computational models is done in two directions: computing devices built according to mechanical/linguistic principles (automata and grammars, Turing machines, register machines, insertion-deletion systems) and devices constructed according to biological principles (cellular automata, membrane systems, networks of evolutionary processors). Both directions contribute to theoretical bases of creation of informational systems of either special or general use with special requirements on the computing speed, robustness and size. Natural (biological) computing follows big ambitions: constructing computing devices from biological components (molecules, DNA, neurons, cells, etc.). Laboratory experiments showed that prototypes of such biodevices can solve many problems faster and more efficiently than traditional devices, due to the massive parallelism. Biocomputing has numerous applications: treating cancer; creating ,,smart drugs"; creating nano-robots to locate unhealthy organs; prediction etc.). Research of these computing

principles is very relevant in present; it can be compared to the great importance of research of such scientists as Goedel, Kleene, Turing, Post and von Neumann, that led to the creation of modern computers.

In the first direction: the results obtained by the researchers from IMCS in the area of minimal universal Turing machines were included in multiple monographs and manuals. Works [17, 19, 46, 48, 50, 51, 73, 77, 92, 95, 97] are devoted to this topic. Computational models equivalent to the Turing machines are type-0 Chomsky grammars and insertion-deletion systems. The results obtained during research of the latter ones are presented in works [43, 57, 59].

Within the second direction, various natural computational models were studied, such as H systems, membrane systems, and networks of evolutionary processors.

Membrane systems, also called P systems, represent a number of regions, separated by membranes. Each region contains objects in some multiplicities. These objects evolve according to certain rules associated to regions and/or membranes. Depending on the computational model, the rules can be of evolution (replacing objects with other ones), communicative (objects move from a region across a membrane into another region), etc. Works of the researchers from IMCS, which concern this domain, can be divided in 7 groups.

Computational power of evolution-communication P systems, the works [6, 23, 33, 53, 54, 136]. Systems with non-cooperative evolution rules and communication rules acting on at most two objects were studied. For such systems, two membranes are sufficient for the computational completeness, and three membranes are sufficient in the deterministic case. Other particular cases were also thoroughly studied.

Computational power of communicative P systems, the works [29, 31, 34, 36, 37, 42, 65, 70, 105, 135, 137, 138, 140, 141, 142]. As opposed to the previous group, the evolution rules are replaced with unbounded supply of objects in the external region (environment). Practically all results obtained by other authors on this topic were improved. In case of communicative rules with at most two objects and one membrane only finite sets can be generated. Three membranes are sufficient for the

computational completeness, while two membranes are enough modulo one additional object.

Efficiency of P systems with active membranes, the works [5, 28, 41, 52, 55, 56, 61, 69, 153]. This computational model with active membranes is convenient for describing solutions of NP-complete and even PSPACE-complete problems. Cases with two polarizations and without polarizations were thoroughly studied.

Computational power of P systems with active membranes, the works [6, 30, 63, 66, 139]. The computational completeness of systems with one membrane and two polarizations, and for systems without polarizations was proved.

Other variants of P systems, the works [6, 24, 25, 32, 35, 39, 40, 57, 63, 64, 66, 133, 134, 139, 143,152, 156]. The following P-systems were investigated: non-distributed; neural; with partial halting; with rules with target indications; systems with energy assigned to membranes; with rules of object diffusion; with bi-stable catalysts, promoters, inhibitors; models with strings.

H systems – abstraction of splicing reactions (performed in biology by restriction enzymes DNA lygase), the works [2, 3, 18, 20, 22, 26, 49, 57, 60, 118]. The results of other authors were improved, obtaining a series of unexpected results. It was proved that time-varied H systems of degree I are computationally complete devices.

Other biocomputing models, the works [27, 67]. Computing by observing, self-assembly, networks of evolutionary processors.

A possibility of efficient use of molecular computing in non-commutative computational systems is argued in [94]. An algorithm of deciding finiteness of algebra dimension is proposed based on molecular operations [125].

,,Life as computation" paradigm was introduced in hope that experience in computer science and mathematics obtained in result of research in parallel systems will help us to understand the complexity of the processes taking place in living cells, contributing to discovery of regularities of biological systems that are otherwise difficult or even

impossible to observe. The goal of this research is to understand the way of functioning of cells or organisms, conditions of adaptation and survival in dynamic environment, all explained in terms of computations.

References

[1] S. Cojocaru, A.Podoplelov, V. Ufnarovski. Non-Commutative Groebner Bases and Anick's resolution. In: Progress in Mathematics, vol.173, Chapter 7, Birkhäuser Verlag, 1999, p.139-159.

[2] M. Margenstern, Yu. Rogozhin. Time-varying distributed H systems of degree 1 generate all recursively enumerable languages. In: ,,Words, Semigroups, and Transductions. Festschrift in Honor of Gabriel Thierrin" (M. Ito, Gh. Păun, Sh. Yu, Eds.), World Scientific, 2001, pp.329-340.

[3] M. Margenstern, Yu. Rogozhin. Time-varying distributed H systems of degree 2 generate all recursively enumerable languages. In: "Where Mathematics, Computer Science, Linguistics and Biology Meet", Chapter 35, Kluwer Academic publishers, 2001, pp. 399-407.

[4] J.Backelin, S.Cojocaru, V.Ufnarovski. BERGMAN. In: "Computer Algebra Handbook". J.Grabmeier, E.Kaltofen, V.Weispfenning (Eds.). Springer, 2003, pp. 349-352.

[5] A. Alhazov, C. Martín-Vide, L. Pan. Solving Graph Problems by P Systems with Restricted Elementary Active Membranes. In: N. Jonoska, Gh. Păun, G. Rozenberg: Aspects of Molecular Computing - Essays dedicated to Tom Head on the occasion of his 70th birthday, Lecture Notes in Computer Science, vol. 2950, Festschrift, Springer, 2004, pp. 1-22.

[6] A. Alhazov, D. Sburlan. Static Sorting P Systems. In: G. Ciobanu, Gh. Păun, M.J. Pérez-Jiménez: "Applications of Membrane Computing, Natural Computing Series", Springer-Verlag, Berlin, 2005, pp. 215-252.

[7] J. Backelin, S. Cojocaru, V Ufnarovski. Mathematical computations using Bergman. Lund University, Sweden, 2005, ISBN 91-631-7203-8, 206 p.

[8] L. Burţeva, S. Cojocaru, C. Gaindric, G. Magariu, T. Verlan. Digital divide: A glance at the problem in Moldova. In: ,,Information technologies communication and human development:Opportunities and challenges", Idea Group Inc., Hershey, London, Melbourne, Singapore, 2006, pp. 77-115.

The chapter is republished in: Information Communication Technologies: Concepts, Methodologies, Tools, and Applications (6 Volumes). Edited by Craig Van Slyke, University of Central Florida, IGI Global, USA, 2008, Chapter 5.21, pp. 2531-2565.

[9] L. Burţeva, S. Cojocaru, C. Gaindric, G. Magariu, T. Verlan. Digital divide: Introduction to the problem. In: ,,Information technologies communication and human

development:Opportunities and challenges", Idea Group Inc., Hershey, London, Melbourne, Singapore, 2006, pp. 57-76.

The chapter is republished in: Information Communication Technologies: Concepts, Methodologies, Tools, and Applications (6 Volumes). Edited by Craig Van Slyke, University of Central Florida, IGI Global, USA, 2008, Chapter 1.6, pp.74-90.

[10] S. Cojocaru. Interfeţe inteligente. Subcapitolul 4.4.4.3 în monografia F.G. Filip ,,Sisteme suport pentru decizii", Editura tehnică, Bucureşti, 2007, pp. 213-215.

[11] M.V.Evstiunin, S.K.Cojocaru, A.N.Terehov, B.A.Ufnarovski. How Pascal and Oberon fall into Samson or the art of compilers construction. Chişinău, "Ştiinţa", 1991, 304 p. (in Russian)

[12] C. Gaindric. Luarea deciziilor. Metode şi tehnologii, Chişinău, "Ştiinţa". 1998, 162 p.

[13] S. Cojocaru, C. Gaindric. Considerente asupra edificării societăţii informaţionale în Moldova, Chişinău, Institutul de Matematică şi Informatică, 2003, 95 p.

[14] Machines, Computations and Universality. M. Margenstern, Yu. Rogozhin (Editors). Lecture Notes in Computer Science, Springer, vol. 2055, Third International Conference, MCU 2001 Chişinău, Moldavia, May 23-27, 2001, Proceedings, 320 p.

[15] Symposium on Intelligent Systems and applications. H.N.Teodorescu, C. Gaindric, E. Sofron (Editors), România, september 19-20, 2003, CD ISBN 973-97737-2-9.

[16] Computational Commutative and Non-Commutative Algebraic Geometry. S. Cojocaru, G. Pfister, V. Ufnarovski (Editors). NATO Science Series. SeriesIII: Computer and Systems Sciences – Vol.196, IOS Press, 2005, 325 p. Cartea este însoţită de un CD cu sisteme de algebră computaţională, care conţine şi sistemul BERGMAN, elaborat cu contribuţia autorului.

[17] M. Kudlek, Yu. Rogozhin. New small universal circular Post machines. Lecture Notes in Computer Science, vol. 2138, Springer, 2001, pp. 217-227.

[18] M. Margenstern, Yu. Rogozhin. About time-varying distributed H systems. Lecture Notes in Computer Science, vol. 2054, Springer, 2001, pp. 53-62.

[19] M. Kudlek, Yu. Rogozhin. A Universal Turing Machine with 3 States and 9 symbols. Lecture Notes in Computer Science, vol. 2295, Springer, 2002, pp. 311-318.

[20] M. Margenstern, Yu. Rogozhin. A universal time-varying distributed H system of degree 1. Lecture Notes in Computer Science, vol. 2340, Springer, 2002, 371-380.

[21] C. Gaindric. Fuzzy evaluation Processing in Decision Support Systems, in Systematic Organization of Information in Fuzzy Systems, NATO Science Series III Computer and Systems Sciences, v.184, IOS Press, Amsterdam, Berlin, Oxford, Tokyo, Washington DC, 2003, pp. 355-358.

[22] M.Margenstern, Yu. Rogozhin, S. Verlan. Time-Varying Distributed H Systems of Degree 2 Can Carry Out Parallel Computations. Lecture Notes in Computer Science, vol. 2568, Springer, 2003, pp. 326-336.

[23] A. Alhazov, M. Cavaliere: Proton Pumping P Systems. In: C. Martín-Vide, G. Mauri, Gh. Păun, G. Rozenberg, A. Salomaa. Membrane Computing, International Workshop, WMC 2003, Tarragona, 2003, Revised Papers, Lecture Notes in Computer Science, vol. 2933, Springer, 2004, pp. 1-18.

[24] V. Rogozhin, E. Boian. Simulation of mobile ambients by P systems. Part 1. Lecture Notes in Computer Science, vol. 2933, Springer, Berlin, 2004, pp. 304–319.

[25] F. Freund, R. Freund, M. Oswald, M. Margenstern, Yu. Rogozhin, S. Verlan. P Systems with Cutting/Recombination Rules Assigned to Membranes. Lecture Notes in Computer Science, vol. 2933, Springer, 2004, pp. 191-202.

[26] M. Margenstern, Yu. Rogozhin, S. Verlan. Time-varying distributed H systems with parallel computations: the problem is solved. Lecture Notes in Computer Science, vol. 2943, Springer, 2004, pp. 48-53.

[27] A. Alhazov, M. Cavaliere: Computing by Observing Bio-Systems: the Case of Sticker Systems. In: C. Ferretti, G. Mauri, C. Zandron: DNA Computing: 10th International Workshop on DNA Computing, DNA10, Milan, Italy, June 7-10, 2004, Revised Selected Papers, Lecture Notes in Computer Science, vol. 3384, Springer, 2005, pp. 1-13.

[28] A. Alhazov, R. Freund. On the Efficiency of P Systems with Active Membranes and Two Polarizations. In: G. Mauri, Gh. Păun, M.J. Pérez-Jiménez, G. Rozenberg, A. Salomaa: Membrane Computing, International Workshop, WMC 2004, Milan, 2004, Revised Selected and Invited Papers, Lecture Notes in Computer Science, vol. 3365, Springer, 2005, pp. 146-160.

[29] A. Alhazov, R. Freund, M. Oswald: Tissue P Systems with Antiport Rules and Small Numbers of Symbols and Cells. In: C. De Felice, A. Restivo: Developments in Language Theory: 9th International Conference, DLT 2005, Palermo, Italy, 2005, Proceedings, Lecture Notes in Computer Science, vol. 3572, Springer, 2005, pp. 100-111.

[30] A. Alhazov, R. Freund, Gh. Păun. Computational Completeness of P Systems with Active Membranes and Two Polarizations. In: M. Margenstern: Machines, Computations, and Universality, International Conference, MCU 2004, Saint Petersburg, 2004, Revised Selected Papers, Lecture Notes in Computer Science, vol. 3354, Springer, 2005, pp. 82-92.

[31] A. Alhazov, M. Margenstern, V. Rogozhin, Yu. Rogozhin, S. Verlan. Communicative P Systems with Minimal Cooperation. In: G. Mauri, Gh. Păun, M.J. Pérez-Jiménez, G. Rozenberg, A. Salomaa: Membrane Computing, International Workshop, WMC 2004, Milan, 2004, Revised Selected and Invited Papers, Lecture Notes in Computer Science, vol. 3365, Springer, 2005, pp. 161 - 177.

[32] A. Alhazov, D. Sburlan. Ultimately Confluent Rewriting Systems. Parallel Multiset-Rewriting with Permitting or Forbidding Contexts. In: G. Mauri, Gh. Păun, M.J. Pérez-Jiménez, G. Rozenberg, A. Salomaa: Membrane Computing, International Workshop, WMC 2004, Milan, 2004, Revised Selected and Invited Papers, Lecture Notes in Computer Science, vol. 3365, Springer, 2005, pp. 178-189.

[33] A. Alhazov. Number of Protons/Bi-stable Catalysts and Membranes in P Systems. Time-Freeness. In: R. Freund, Gh. Păun, G. Rozenberg, A. Salomaa: Membrane Computing, International Workshop, WMC 2005, Vienna, 2005, Revised Selected and Invited Papers, Lecture Notes in Computer Science, vol. 3850, Springer, 2006, pp. 79-95.

[34] A. Alhazov, R. Freund, M. Oswald. Symbol / Membrane Complexity of P Systems with Symport / Antiport Rules. In: R. Freund, Gh. Păun, G. Rozenberg, A. Salomaa: Membrane Computing, International Workshop, WMC 2005, Vienna, 2005, Revised Selected and Invited Papers, Lecture Notes in Computer Science, vol. 3850, Springer, 2006, pp. 96-113.

[35] A. Alhazov, R. Freund, M. Oswald, M. Slavkovik. Extended Spiking Neural P Systems. In: H.J. Hoogeboom, Gh. Păun, G. Rozenberg, A. Salomaa: Membrane Computing, International Workshop, WMC 2006, Leiden, The Netherlands, 2006, Revised Selected and Invited Papers, Lecture Notes in Computer Science, vol. 4361, Springer, 2006, pp. 123-134.

[36] A. Alhazov, R. Freund, Yu. Rogozhin. Computational Power of Symport/Antiport: History, Advances, and Open Problems. In: R. Freund, Gh. Păun, G. Rozenberg, A. Salomaa: Membrane Computing, International Workshop, WMC 2005, Vienna, 2005, Revised Selected and Invited Papers, Lecture Notes in Computer Science, vol. 3850, Springer, 2006, pp. 1-30.

[37] A. Alhazov, Yu. Rogozhin. Towards a Characterization of P Systems with Minimal Symport/Antiport and Two membranes. In: H.J. Hoogeboom, Gh. Păun, G. Rozenberg, A. Salomaa: Membrane Computing, International Workshop, WMC 2006, Leiden, The Netherlands, 2006, Revised Selected and Invited Papers, Lecture Notes in Computer Science, vol. 4361, Springer, 2006, pp. 135 - 153.

[38] S. Cojocaru, A. Colesnicov, L. Malahov. Providing Modern Software Environments to Computer Algebra Systems. Computer Algebra in Scientific Computing. Lecture Notes in Computer Science, vol. 4194, Springer, 2006, pp. 129-140.

[39] Yu. Rogozhin, S. Verlan. On the Rule Complexity of Universal Tissue P Systems. Lecture Notes in Computer Science, vol. 3850, Springer, 2006, pp. 356-362.

[40] A. Alhazov, R. Freund, M. Oswald, S. Verlan. Partial Halting in P Systems Using Membrane Rules with Permitting Contexts. In: J. Durand-Lose, M. Margenstern (Eds.), Machines, Computations, and Universality, 5th International Conference, MCU 2007, Orléans, Lecture Notes in Computer Science, vol. 4664, Springer, 2007, pp. 110-121.

[41] A. Alhazov, M.J. Pérez-Jiménez. Uniform Solution of QSAT Using Polarizationless Active Membranes. In: J. Durand-Lose, M. Margenstern (Eds.), Machines, Computations, and Universality, 5th International Conference, MCU 2007, Orléans, Lecture Notes in Computer Science, vol. 4664, Springer, 2007, pp. 122-133.

[42] A. Alhazov, Yu. Rogozhin. Skin Output in P Systems with Minimal Symport/Antiport and Two Membranes. In: G. Eleftherakis, P. Kefalas, Gh. Păun, G. Rozenberg, A. Salomaa (Eds.), Membrane Computing, 8th International Workshop, WMC 2007 Thessaloniki, 2007, Revised Selected and Invited Papers. Lecture Notes in Computer Science, vol. 4860, Springer, 2007, pp. 97-112.

[43] A. Matveevici, Yu. Rogozhin, S. Verlan. Insertion-Deletion Systems with One-Sided Contexts. Lecture Notes in Computer Science, vol. 4664, Springer, 2007, pp. 205-217.

[44] E. Boian, A.Danilchenco, L.Topal. Automation of Word-forming Process in the Romanian Language. Studies in Informatics and Control, vol.3, no.1, March,1994, pp. 43 - 52.

[45] S. Cojocaru, M.Evstiunin, V. Ufnarovski. Romanian Spelling Checker. Studies in Informatics and Control, vol.3, no.1, 1994, Bucharest, pp. 53-58.

[46] Yu. Rogozhin. Small Universal Turing Machines. Theoretical Computer Science, 168-2, Elsevier, 1996, pp. 215-240.

[47] S. Cojocaru, V. Ufnarovski. BERGMAN under MS DOS and Anick's resolution. Discrete Mathematics and Theoretical Computer Science, vol.1, Special Issue: Lie Computations, 1997, pp. 139-147.

[48] Yu. Rogozhin. A universal Turing machine with 22 states and 2 symbols. Romanian Journal of Information Science and Technology, vol. 1, no.3, Romanian Academy, 1998, 259-265.

[49] M. Margenstern, Yu. Rogozhin. A universal time-varying distributed H-system of degree 2. Biosystems, 52, Elsevier, 1999, pp. 73-80.

[50] M. Kudlek, Yu. Rogozhin. Small universal Turing and circular Post machines. PU.M.A. Pure Mathematics and Applications, Budapest University of Economics, University of Siena, HU ISSN 1218-4586, 13, no.1-2, 2002, pp. 197-210.

[51] M. Margenstern, Yu. Rogozhin. Self-describing Turing machines. Fundamenta Informaticae, 50, no.3-4, IOS Press, 2002, pp. 285-303.

[52] A. Alhazov, C. Martín-Vide, L. Pan. Solving a PSPACE-Complete Problem by P Systems with Restricted Active Membranes. Fundamenta Informaticae, volume 58, number 2, IOS Press, 2003, pp. 67-77.

[53] A. Alhazov. Minimizing Evolution-Communication P Systems and Automata. New Generation Computing, volume 22, number 4, Ohmsha Ltd., Springer, 2004, pp. 299-310.

[54] A. Alhazov. On Determinism of Evolution-Communication P Systems. Journal of Universal Computer Science, volume 10, number 5, Graz University of Technology, 2004, pp. 502-508.

[55] A. Alhazov, L. Pan. Polarizationless P Systems with Active Membranes. Grammars, volume 7, 2004, pp. 141-159.

[56] A. Alhazov, L. Pan, Gh. Păun. Trading Polarizations for Labels in P Systems with Active Membranes. Acta Informatica, volume 41, numbers 2-3, Springer, 2004, pp. 111-144.

[57] M. Margenstern, G.h. Păun, Yu. Rogozhin. On the power of the (molecular) crowd: set-conditional string processing. Publicationes Mathematicae, Debrecen (Hungary), vol.65, 2004, pp. 575 – 590.

[58] E. Boian, C.Ciubotaru, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahov. Lexical Resources for Romanian. Academia Română. Memoriile secţiilor ştiinţifice. Seria IV, tomul XXVI, 2003, Editura Academiei Române, Bucureşti, 2005, pp. 267-278.

[59] M. Margenstern, Gh. Păun, Yu. Rogozhin, S. Verlan. Context-free insertion-deletion systems. Theoretical Computer Science, vol.330, issue 2, Elsevier, 2005, pp. 339-348.

[60] M. Margenstern, S. Verlan, Yu. Rogozhin. Time-Varying Distributed H Systems: An Overview. Fundamenta Informaticae, vol. 64, IOS Press, 2005, pp. 291-306.

[61] L. Pan, A. Alhazov, T.-O. Ishdorj. Further Remarks on P Systems with Active Membranes, Separation, Merging, and Release Rules. Soft Computing. A Fusion of Foundations, Methodologies and Applications, volume 9, number 9, Springer, 2005, pp. 686-690.

[62] E. Rîbac, S. Cojocaru,C. Gaindric, S.Puiu, V.Turcanu. The user interface design for a decision support system in sonographic investigation, Revista de inventica, N48, v.I AN V-2005 pp. 7-12.

[63] A. Alhazov. P Systems without Multiplicities of Symbol-Objects. Information Processing Letters, volume 100, number 3, Elsevier, 2006, pp. 124-129.

[64] A. Alhazov, R. Freund, A. Leporati, M. Oswald, C. Zandron. (Tissue) P Systems with Unit Rules and Energy Assigned to Membranes. Fundamenta Informaticae, volume 74, number 4, IOS Press, 2006, pp. 391-408.

[65] A. Alhazov, R. Freund, M. Oswald. Cell / Symbol Complexity of Tissue P Systems with Symport / Antiport Rules. International Journal of Foundations of Computer Science, volume 17, number 1, World Scientific, 2006, pp. 3-26.

[66] A. Alhazov, R. Freund, A. Riscos-Núñez. Membrane Division, Restricted Membrane Creation and Object Complexity in P Systems. International Journal of Computer Mathematics, volume 83, number 7, Taylor & Francis, 2006, pp. 529-548.

[67] A. Alhazov, C. Martín-Vide, Yu. Rogozhin. On the Number of Nodes in Universal Networks of Evolutionary Processors. Acta Informatica, volume 43, number 5, Springer, 2006. pp. 331-339.

[68] E. Boian, C.Ciubotaru, S. Cojocaru, A. Colesnicov, L. Malahov. Lexical resources for Romanian: creation, word inflection, checking. Revista de Inventica. Romanian Journal for Creativity in Engineering and Technology – Research and Education Reports. Nr.53,vol. X., an XVI-2006, pp. 27-32.

[69] L. Pan, A. Alhazov. Solving HPP and SAT by P Systems with Active Membranes and Separation Rules. Acta Informatica, volume 43, number 2, Springer, 2006, pp. 131-145.

[70] A. Alhazov, Yu. Rogozhin, S. Verlan: Minimal Cooperation in Symport/Antiport Tissue P Systems. International Journal of Foundations of Computer Science, volume 18, number 1, World Scientific, 2007, pp. 163-180.

[71] Gh. Duca, C. Gaindric. A vision upon Science in a Knowledge Society, Studies in informatics and Control, v.16, N 4, pp. 445-452, Bucharest, 2007.

[72] C. Ciubotaru. Practical approach to solution of the cyclic recurrence problem for attribute grammars. Bulletin of the Academy of Sciences of Moldova. The series of Physical-Technical and Mathematical sciences. No.2, 1991, pp. 32-39. (in Russian)

[73] Yu. Rogozhin. Universil Turing machine with 10 states and 3 symbols. Bulletin of the Academy of Sciences of Moldova. Mathematics. Chisinau, 1992, No. 4 (10), pp. 80-82. (in Russian)

[74] S. Cojocaru, M. Evstiunin, V. Ufnarovski. Detecting and correcting spelling errors for Romanian language. Computer Science Journal of Moldova, vol.1, no.1, 1993, pp. 3-22.

[75] C. Gaindric, V.Ungureanu, D.Zaporojan. A decision support system for Resources planning in scientific and technical programs, Computer Science Journal of Moldova, vol.1, no.2, 1993, pp. 105-109.

[76] G. Magariu, V. Madan, L. Burţeva. What does the simulation system SOL/PC do? Computer Science Journal of Moldova. vol.1, no.2, 1993, pp. 3-13.

[77] Yu. Rogozhin, About Shannon's problem for Turing machines. Computer Science Journal of Moldova, vol.1, no.3(3), 1993, pp. 108-111.

[78] N. Andronaty, A. Golovnea, Yu. Rogozhin, The structural robustness of multiprocessor computing system. Computer Science Journal of Moldova, vol.3, no.3, 1995, pp. 212-237.

[79] S. Cojocaru, V. Ufnarovski. Noncommutative Groebner basis, Hilbert series, Anick's resolution and BERGMAN under MS DOS. Computer Science Journal of Moldova, vol.3, no.1, 1995, pp. 24-39.

[80] A. Colesnicov. Message composition and its application to event-driven system construction. Computer Science Journal of Moldova, vol. 3, no. 2 (8), 1995, pp. 123-133.

[81] A. Colesnicov. The Roumanian spelling checker ROMSP: the project overview. Computer Science Journal of Moldova, vol.3, no. 1 (7), 1995, pp. 40-54.

[82] N. Shvets, C. Ciubotaru. Meta-generation of syntax-oriented editors. Computer Science Journal of Moldova, vol.3, no.1, 1995, pp. 3-9.

[83] E. Boian, S. Cojocaru, L. Malahov. Some tools to implement linguistics applications oriented to the Romanian language. Computer Science Journal of Moldova, vol.4, no.2, 1996, pp. 204-223.

[84] S. Cojocaru, E. Boian. The inflexion regularities for the Romanian language. Computer Science Journal of Moldova, vol.4, no.1, 1996, pp. 40-58.

[85] A. Colesnicov. Implementation and usage of the Bergman package shell. Computer Science Journal of Moldova, vol. 4, no. 2 (11), 1996, pp. 260-276.

[86] V. Demidova, T. Verlan. An approach to the word division into syllables for Romanian language. Computer Science Journal of Moldova, vol.4, no.1, 1996, pp. 59-68.

[87] L.Burţeva. On the Architecture of Problem-Oriented Simulation Systems. Computer Science Journal of Moldova, vol.5, no.3(15), 1997, pp. 273-284.

[88] C. Ciubotaru. Grammar flow analysis reduction to scheduling problem. Computer Science Journal of Moldova, vol.5, no.3(15), 1997, pp. 353-365.

[89] C. Gaindric. Decision support systems. Acta Academia, 1998 Intern. Informatization Academy, Chişinău, Evrica, 1998, pp. 236-245. (In Russian)

[90] G. Magariu. A technique of creation of simulation system with experiments design and analysis controlled by user queries, Computer Science Journal of Moldova, vol.6, no.3(18), 1998, pp. 286-293.

[91] G. Magariu, I. Verlan, L. Burţeva. The simulation system of credit portfolio evaluation. Computer Science Journal of Moldova. vol.6, no.3(18), 1998, p.294-305.

[92] M. Margenstern, Yu. Rogozhin. Some small self-describing Turing machines. Computer Science Journal of Moldova, ISSN 1561-4042, vol.6, no.1, 1998, pp. 57-82.

[93] C. Gaindric. Decision support system for vehicle dispatching, Computer Science Journal of Moldova, ISSN 1561-4042, vol.7, no. 1(19), 1999, pp. 86-104.

[94] S. Cojocaru, V. Ufnarovski. Non-commutative computer algebra and molecular computing. Computer Science Journal of Moldova, ISSN 1561-4042, vol.9, no.3, 2001, pp. 369-377.

[95] M. Kudlek, Yu. Rogozhin. Small Universal Circular Post Machines. Computer Science Journal of Moldova, ISSN 1561-4042, vol. 9, no.1, 2001, pp. 34-52.

[96] A. Tkachenko, A. Alhazov. The Multiobjective Bottleneck Transportation Problem. Computer Science Journal of Moldova, ISSN 1561-4042, vol. 9, no. 3(27), 2001, pp. 321-335.

[97] A. Alhazov, M. Kudlek, Yu. Rogozhin. Nine Universal Circular Post Machines. Computer Science Journal of Moldova, ISSN 1561-4042, vol. 10, no. 3(30), 2002, pp. 247-262.

[98] I. Attali, C. Ciubotaru, N. Meergus. Experimental functional realization of attribute grammar systems. Computer Science Journal of Moldova, ISSN 1561-4042, vol.10, no. 2(29), 2002, pp. 190–203.

[99] S. Cojocaru, A. Colesnicov, L. Malahov. Network version of the computer algebra system Bergman. Computer Science Journal of Moldova, ISSN 1561-4042, vol.10, no.2, 2002, pp. 216-222.

[100] C. Gaindric. The digital technologies as the chance for sustainable development of Moldova, Computer Science Journal of Moldova, ISSN 1561-4042, vol. 10, no. 1(28), 2002, pp. 53-58.

[101] E. Boian, S. Cojocaru, V. Demidova. Resurse lexico-gramaticale şi instrumentare pentru aplicaţii de limbaj natural. Economica, Nr.2 (42), 2003, Editura ASEM, pp. 99-103.

[102] L. Burţeva. The comparative analysis of implementation practices of e-government services, Computer Science Journal of Moldova, ISSN 1561-4042, vol.12, no. 3(36), 2004, pp. 457-466.

[103] L. Burţeva, S. Cojocaru, C. Gaindric, G. Magariu, T. Verlan. Services in Public Administration (e-government); Privacy and Freedom of Information (review of study mode for situation in Moldova), Computer Science Journal of Moldova, ISSN 1561-4042, vol.12 no. 3 (36), 2004, pp. 467-496.

[104] S. Cojocaru, A. Colesnicov, L. Malahov. Interfaces to symbolic computation systems: reconsidering experience of Bergman. Computer Science Journal of Moldova, ISSN 1561-4042, vol. 13, no.2, 2005, pp. 232-244.

[105] A. Alhazov, Yu. Rogozhin. Generating Languages by P Systems with Minimal Symport/Antiport. Computer Science Journal of Moldova, ISSN 1561-4042, vol. 14, no. 3(42), 2006, pp. 299-323.

[106] S. Cojocaru. The assessment of the inflexion models for Romanian. Computer Science Journal of Moldova, ISSN 1561-4042, vol.14, no.1(40), 2006, pp. 103-112.

[107] S. Cojocaru, A. Colesnicov, L. Malahov. Integrity and correctness checking of a lexical database. Computer Science Journal of Moldova, ISSN 1561-4042, vol.14, no. 1(40), 2006, pp. 138-151.

[108] L. Burţeva, S. Cojocaru, C. Gaindric, E.Jantuan, SONARES-A decision support system in ultrasound investigation. Computer Science Journal of Moldova ISSN 1561-4042, vol.15, no.2 (44), 2007, pp. 153-177.

[109] I. Coşuleanu, C. Gaindric. Distance voting (e-voting): the ways of its applicability in Moldova. Computer Science Journal of Moldova, v ISSN 1561-4042, vol.15, no.3 (45), 2007, pp. 354-380.

[110] C. Gaindric, V.Ungureanu, D.Zaporojan. A decision support system for Resources planning in scientific and technical programs, Advances in fuzzy Sets and applications, Ed.Univ. Iaşi, 1990.

[111] G. Magariu, L. Burţeva. Analysis of experiment results in simulation system SOL/PC, Proceedings International AMSE Conference on Applied Modeling and Simulation, Lviv, Ukrain, pp.51-62, 1993.

[112] T. Verlan. Many words, but little place (about electronical dictionary of sinonims). Proceedings of technical-scientific conference ,,Informatics and computing engineering", Chisinau, 1993, pp.40-43. (in Russian)

[113] F. Dlikman, F. Frishberg, V. Sedyakin, G. Magariu, T. Tofan, L. Bureva. Evaluation of characteristics and choice of working regimes of geliogeophysical informational-calculational system according to simulation results, In: Advances in Modeling &Analysis, B, AMSE Press, Vol.31, 4, 1994, pp. 23-29.

[114] G. Magariu. Model running visualization in simulation system SOL/PC, Proceedings of European simulation meeting on simulation tools and applications, Gyor, Hungary, 1995, pp. 217-223.

[115] S. Cojocaru. Lexicon român: instrumentar, implementare, utilizare. Limbaj i tehnologie, Academia Română, Bucureşti, 1996, p.37-40.

[116] S. Cojocaru. Romanian Lexicon: Tools, Implementation,Usage. Recent Advances in Romanian Language Technology. Editors: D. Tufiş, P. Andersen. Editura Academiei Române, Bucureşti, 1997, p.107-114.

[117] C. Gaindric. Decision support systems: concepts evolution and some perspectives. Proceedings of International Symposium ,,Computers in Europe – past, present and future", Kiev, November 5-7, 1998, pp. 216-220. (in Russian)

[118] . L. Priese, Yu. Rogozhin, M. Margenstern. Finite H-systems with 3 Test Tubes are not Predictable. In Proceedings of Pacific Simposium on Biocomputing, 3, Kapalua, Maui, January 1998, Hawaii, USA (R.Altman, A.Dunker, L.Hanter, T.Klein, eds.), World Sci.Publ., Singapore, 1998, pp.545-556.

[119] E. Boian, S. Cojocaru, L. Malahov. Instrumentar pentru aplicaţii lingvistice. Terminologia în România şi Republica Moldova. "Clusium", România, 2000, pp.38-41.

[120] E. Boian, S. Cojocaru, L. Malahov. Instruments pour applications linguistiques. La terminologie en Roumanie et en Republique de Moldova, Hors serie, N4, 2000, p.42-44.

[121] I. Attali, C. Ciubotaru. Functional realization of attribute grammar systems. Proceedings of the 3-rd International Conference "Microelectronics and Computer Science", Technical University of Moldova, Chişinău, September 26-28, 2002, vol. II, p.87-91.

[122] C. Gaindric. Impact of transition to Informational Society in Moldova, Proceedings of the 3rd International Conference on ,,Microelctronics and Computer Science" vol.2, Chişinău, 2002, pp. 16-20.

[123] J. Backelin, S. Cojocaru, V. Ufnarovski. The Computer Algebra Package Bergman: Current State. Commutative Algebra, Singularities and Computer Algebra.NATO Science Series, II Mathematics, Physics and Chemistry, Vol. 115, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2003, pp. 75-100.

[124] E. Boian, C.Ciubotaru, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahov. Lexical resources for Romanian - a project overview. Symposium on Intelligent Systems and Applications, September 19-20, 2003, Iaşi, Romania. Eds.: H.N.Teodorescu, G.Gaindric, E.Sofron. Publisher: Tehnici şi Tehnologii, Iaşi. ISBN 973-97737-2-9 (CD).

[125] S. Cojocaru, V. Ufnarovski. Non-Commutaive Computer Algebra and Molecular Computing. 2nd Annual Meeting of Project MolCoNet IST-2001-32008. November 27-29, 2003, Wien. Vienna University of Technology, 4 p.

[126] G. Magariu, L. Burţeva. Simulation in bank management process. Trends in the development of the information and communication technologies in education and management. International conference. March 20-21, 2003. Chişinău, ASEM, 2003, pp. 182-185.

[127] L.Burţeva, T.Verlan, C.Gaindric, S.Cojocaru, G.Magariu, C.Ciubotaru. Informational analitical system ,,Scientific potential of Moldova", BIT+, III International Conference ,,Informational Technologies – 2003", April 7-11, 2003, Chisinau, Moldova, V.3, pp.174-179.

[128] B. Rybakin, G. Magariu, L. Burţeva, T. Verlan. Experimental data as initial and boundary conditions for mathematical model of a solid waste landfill. Mathematical modeling in education, science and production. Proceedings of III International scientific-practical conference, September 17-20, 2003, pp.52-53. (in Russian)

[129] L. Burţeva, T. Verlan, C. Gaindric, S. Cojocaru, G. Magariu. The comparative analysis of different approaches to the e-government implementation: e-government services. Abstracts on BIT+ 2004 "IV International Conference on Information Technologies 2004", 3-7 May, 2004, Chisinau, pp. 66-68. (in Russian)

[130] S. Cojocaru, C. Gaindric. Moldova – căi de integrare digitală, BIT+ IV International Conference on Informational Technologies 2004, 3-7 mai, 2004, pp. 23-28.

[131] J. Maffia, B.P. Rybakin, G..A. Magariu, L.V. Burţeva, T.B.Verlan. SoWaDec: Computer modeling of biogas generation processes. Proceedings of the Second Conference of the Mathematical Society of the Republic of Moldova, Chişinău, August 17-19, 2004, pp. 274-278.

[132] O.Popcova, S. Cojocaru, C. Gaindric. Image processing in ultrasound diagnostic system. 8th International Symposium on Automatic Control and Computer Science, SACCS 2004. Iaşi, România. ISBN 973-621-086-3 (CD).

[133] V. Rogojin, E. Boian. Simulation of Mobile Ambients by tissue P systems with a dynamic network of membranes. In: I. Dzitac, T. Maghiar, C. Popescu (Eds.), Proceedings of the International Conference on Computers and Communications - ICCC 2004, pp. 377-382, Editura Universităţii din Oradea, 2004.

[134] A. Alhazov. Maximally Parallel Multiset-Rewriting Systems: Browsing the Configurations. In: M.A. Gutiérrez-Naranjo, A. Riscos-Núñez, F.J. Romero-Campero, D. Sburlan. RGNC Report, ISBN: 84-609-67719-9, University of Seville, Third Brainstorming Week on Membrane Computing, Fénix Editora, Sevilla, 2005, pp. 1-10.

[135] A. Alhazov. Solving SAT by Symport/Antiport P Systems with Membrane Division. In: M.A. Gutiérrez-Naranjo, Gh. Păun, M.J. Pérez-Jiménez: Cellular Computing (Complexity Aspects), ESF PESC Exploratory Workshop, ISBN: 84-609-5338-6, Fénix Editora, Sevilla, 2005, pp. 1-6.

[136] A. Alhazov, M. Cavaliere. Evolution-Communication P Systems: Time-freeness. In: M.A. Gutiérrez-Naranjo, A. Riscos-Núñez, F.J. Romero-Campero, D. Sburlan. RGNC Report, ISBN: 84-609-67719-9, University of Seville, Third Brainstorming Week on Membrane Computing, Fénix Editora, Sevilla, 2005, pp. 11-18.

[137] A. Alhazov, R. Freund. P Systems with One Membrane and Symport/ Antiport Rules of Five Symbols are Computationally Complete. In: M.A. Gutiérrez-Naranjo, A. Riscos-Núñez, F.J. Romero-Campero, D. Sburlan: RGNC Report, ISBN: 84-609-67719-9, University of Seville, Third Brainstorming Week on Membrane Computing, Fénix Editora, Sevilla, 2005, pp. 19-28.

[138] A. Alhazov, R. Freund, M. Oswald. Tissue P Systems with Antiport Rules and Small Number of Symbols and Cells. In: M.A. Gutiérrez-Naranjo, Gh. Păun, M.J. Pérez-Jiménez: Cellular Computing (Complexity Aspects), ESF PESC Exploratory Workshop, ISBN: 84-609-5338-6, Fénix Editora, Sevilla, 2005, pp. 7-22.

[139] A. Alhazov, R. Freund, A. Riscos-Núñez. One and Two Polarizations, Membrane Creation and Objects Complexity in P Systems. Seventh International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC'05), IEEE Computer Society, 2005, pp. 385-394.

[140] A. Alhazov, R. Freund, Yu. Rogozhin. Some optimal results on symport/antiport P systems with minimal cooperation. Cellular Computing (Complexity aspects). ESF PESC Exploratory Workshop, ISBN: 84-609-5338-6, Fénix Editora, Sevilla, Spain, January 31 – February 2, 2005, pp. 23 – 36.

[141] A. Alhazov, Yu. Rogozhin. Minimal Cooperation in Symport/Antiport P Systems with one membrane. In: M.A. Gutiérrez-Naranjo, A. Riscos-Núñez, F.J. Romero-Campero, D. Sburlan: RGNC Report, ISBN: 84-609-67719-9, University of Seville, Third Brainstorming Week on Membrane Computing, Fénix Editora, Sevilla, 2005, pp. 29 - 34.

[142] A. Alhazov, Yu. Rogozhin, S. Verlan. Symport/antiport tissue P systems with minimal cooperation. Cellular Computing (Complexity aspects). ESF PESC Exploratory Workshop, ISBN: 84-609-5338-6, Fénix Editora, Sevilla, Spain, January 31 – February 2, 2005, 37 – 52.

[143] A. Alhazov, S. Verlan. Sevilla Carpets of Deterministic Non-cooperative P Systems. In: M.A. Gutiérrez-Naranjo, Gh. Păun, M.J. Pérez-Jiménez: Cellular Computing (Complexity Aspects), ESF PESC Exploratory Workshop, ISBN: 84-609-5338-6, Fénix Editora, Sevilla, 2005, pp. 53-60.

[144] E. Boian, C.Ciubotaru, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahov. Technologization of Romanian: linguistic resources, applications, tools. Proceedings of the 4rd International Conference on Microelectronics and Computer Science. Vol.II, 2005, pp. 519-522.

[145] E. Boian, S. Cojocaru, A. Colesnicov, L. Malahov, T.Baltaga. Development of Tools to inflect compound words in Romanian. Proceedings of the 4rd International Conference on Microelectronics and Computer Science. Vol.II, 2005, pp. 465-469.

[146] L. Burţeva, S. Cojocaru, C. Gaindric, G. Magariu, T. Verlan. Digital divide: a glance at the problem in Moldova Proceedings of the 4rd International Conference on Microelectronics and Computer Science. Vol.II, 2005, pp. 23-29.

[147] S. Cojocaru, A. Colesnicov, L. Malahov, V. Ufnarovski. Problems in interaction with the Computer Algebra System Bergman. Computational Commutative and Non-Commutative Algebraic Geometry. NATO Science Series. SeriesIII: Computer and Systems Sciences – Vol.196, Eds.: S. Cojocaru, G. Pfister, V. Ufnarovski. IOS Press, 2005, pp. 185-198.

[148] E. Rîbac, S. Cojocaru, C. Gaindric, S. Puiu, V. Turcanu. The examination support in sonographic investigations Proceedings of the 4rd International Conference on Microelectronics and Computer Science. Technical University of Moldova, Chişinău Vol.II, 2005, pp. 271-274.

[149] E. Rîbac, S. Cojocaru, C. Gaindric, S. Puiu, V. Turcanu. The process of designing and implementing the examination support in sonogfaphic investigations.The International conference ,,Advanced information and telemedicine technologies for health", Nov.8-10, 2005, Minsk, Belarus.Proceedings, vol.2, pp. 44-47.

[150] B. Rybakin, G..Maffia, G. Magariu, L. Burţeva, T. Verlan. Computer modelling of physical and biochemical processes of organics decomposition at solid waste landfills. Abstracts on 4th International Congress on Waste Management WasteTech-2005", May 31-June 3, 2005, Moscow, pp. 246-247. (in Russian)

[151] B. Rybakin, G. Magariu, L. Burţeva, T.Verlan, G. Maffia. The comparative analysis of some models of biogas generation processes Proceedings of the 4th International Conference on "Microelectronics and Computer Science" (Volume II), Technical University of Moldova, Chişinău, September 15-17, 2005, pp. 44-48.

[152] A. Alhazov, C. Bonchiş, G. Ciobanu, C. Isbaşa: Encodings and Arithmetic Operations in P Systems. In: M.A. Gutiérrez-Naranjo, Gh. Păun, A. Riscos-Núñez, F.J. Romero-Campero: RGNC Report, ISBN: 84-611-0681-4, University of Seville, Fourth Brainstorming Week on Membrane Computing, volume 1, Fénix Editora, Sevilla, 2006, pp. 1-28.

[153] A. Alhazov, M.J. Pérez-Jiménez. Uniform Solution to QSAT Using Polarizationless Active Membranes. In: M.A. Gutiérrez-Naranjo, Gh. Păn, A. Riscos-Núñez, F.J. Romero-Campero. RGNC Report, ISBN: 84-611-0681-4, University of Seville, Fourth Brainstorming Week on Membrane Computing, volume 1, Fénix Editora, Sevilla, 2006, pp. 29-40.

[154] C. Ciubotaru, E. Boian, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahov, O. Burlaca.Resurse lingvistice reutilizabile. Lucrările atelierului ,,Resurse lingvistice şi instrumente pentru prelucrarea limbii române", Iaşi, Editura Universităţii ,,A.I.Cuza", 2006, pp. 75-79.

[155] C. Ciubotaru, E. Boian, S. Cojocaru, G. Magariu, T. Verlan, Iu. Rogojin. Sistem de instruire asistată de calculator pentru morfologia limbii române. Lucrările atelierului ,,Resurse lingvistice şi instrumente pentru prelucrarea limbii române", Iaşi, Editura Universităţii ,,A.I.Cuza", 2006, pp. 135-139.

[156] A. Alhazov, R. Freund, M. Oswald, S. Verlan: Partial Versus Total Halting in P Systems. In: M.A. Gutiérrez-Naranjo, Gh. Păun, A. Romero-Jiménez, A. Riscos-Núñez: RGNC Report, ISBN: 978-84-611-6776-0, University of Seville, Fifth Brainstorming Week on Membrane Computing, Fénix Editora, Sevilla, 2007, pp. 1-20.

[157] C.Ciubotaru, E. Boian, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahov, G. Magariu, T. Verlan. Tehnologii pentru generarea sistemelor de instruire, dicţionarelor electronice specializate şi ghidurilor lingvistice. Proceedings of the 5th International Conference on Microelectronics and Computer Science ICMCS-2007, vol.I, Chişinău, Technical University of Moldova, 2007, pp. 20-23.

[158] S. Cojocaru, C. Gaindric. Decision support system in ultrasound investigations, Proceedings XIII-th International conference KDS-2007, vol.1, ITHEA, Sofia, 2007, pp. 241-246.

Artiom Alhazov, Elena Boian,
Liudmila Burtseva, Constantin Ciubotaru,
Svetlana Cojocaru, Alexandru Colesnicov,
Constantin Gaindric, Galina Magariu,
Iurie Rogojin, Tatiana Verlan

Received June 21, 2008

Institute of Mathematics and Computer Science,
Academy of Sciences of Moldova
5, Academiei str., Chisinau,
MD 2028, Moldova
E–mail: *artiom@math.md*, *lena@math.md*, *burtsevamath.md*,
sveta@math.md, *kae@math.md*, *gaindric@math.md*,
gmagariu@math.md, *rogozhin@math.md*, *tverlan@math.md*