On Correcting of the Full Burst Errors for Reed-Solomon Codes

V.P. Semerenko, Ph.D., Member, IEEE
Departament of computer technique,
National Technical University,
Vinnytsia, Ukraine
vpsemerenko@mail.ru

***Abstract*—The representation of Reed-Solomon (RS) codes based on the mathematical theory of the linear finite-state machines (LFSM) is considered. The multilevel graphical and automatical models of the LFSR are offered. The algorihtm of full error burst correction of arbitrary length based LFSM models is suggested.**

***Keywords** - Reed-Solomon codes; error bursts; linear finite-state machines; graphical model.*

I. The Mathematical Models of Reed- Solomon Codes

Reed-Solomon (RS) code of length $n = q - 1$ over $GF(q)$ (,), which can be corrected τ random errors usually are presented by the generator polynomial

$$g(x) = \alpha^0 + \alpha^1 x + \alpha^2 x^2 + \ldots + \alpha^{2\tau-1} x^{2\tau-1} + x^{2\tau}. \quad (1)$$

For simplification of coding and decoding procedures it is more convenient to represent RS code by means of the theory of special class of finite automata – Linear Finite-State Machine (LFSM). According to [2,3], a LFSM over $GF(q)$ is defined by a state (transition) function

$$S(t+1) = A \times S(t) + B \times U(t), \quad GF(q) \quad (2)$$

and an output function

$$Y(t) = C \times S(t) + D \times U(t), \quad GF(q)$$

where t is a index of discrete time, $S(t)$, $U(t)$ and $Y(t)$ are the state, the input and the output vectors respectively; A, B, C, D are the LFSM characteristic matrices.

For the majority of the tasks the transition function is used only, therefore is enough only presence of matrices A and B which can be presented as follows:

$$A = \begin{vmatrix} 0 & 0 & \cdots & 0 & \alpha^0 \\ \alpha^0 & 0 & \cdots & 0 & \alpha^1 \\ 0 & \alpha^0 & \cdots & 0 & \alpha^2 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \alpha^0 & \alpha^{2\tau-1} \end{vmatrix}, \quad B = \begin{vmatrix} \alpha^0 \\ 0 \\ 0 \\ \cdots \\ 0 \end{vmatrix}.$$

(3)

The entries of the last column of the matrix A from (3) are the constant coefficients of the generator polynomial (1).

As RS codes can be described by means of finite automata, therefore is natural the transitions diagram (TD) of finite automata can be chosen as graphical model of such codes.

For r -dimensional LFSM over $GF(q)$ the graph of finite automaton is the directed graph $G_{FA}(V_{FA}, E_{FA})$, in which vertices from the set V_{FA} correspond to the internal states of the automaton, and the edges from the set E_{FA} show the directions of the transitions between internal states.

In general case the zero edge e^{out}_{null} corresponding of zero element over $GF(q)$ and n edges e^{out}_0 , … , e^{out}_{n-1} (them we shall call nonzero), which correspond to degrees , … , α^{n-1} of a primitive element α over $GF(q)$, can go out from the vertex v_j . The zero edge e^{in}_{null} corresponding of zero element over $GF(q)$ and n edges e^{in}_0 , … , e^{in}_{n-1} (them we shall call nonzero too), which correspond to degrees , … , α^{n-1} of a primitive element α over $GF(q)$, can put into to the vertex v_j ($v_j \in V_{FA}$, $e^{in}_{null}, e^{in}_i, e^{out}_{null}, e^{out}_i \in E_{FA}$, $i = 0, \ldots, n-1$, $j = 1, \ldots, q^r$).

As PC codes belong to a class of cyclic codes, therefore the vertices of the graph G_{FA} form numerous cycles. It is possible to perform the ordering of these ZC's on the basis of zero edges.

Always there is one vertex v^0_{null} for which entering e^{in}_{null} and leaving e^{out}_{null} zero edges are united and form a zero cycle (ZC). According to terminology from [4],[5] which we shall adhere further, we believe, that the vertex v^0_{null} with the edges e^{in}_{null} and e^{out}_{null} form a trivial ZC (TZC).

Other vertices of the graph G_{FA} by means of zero edges form cycles of length no more n which it is possible to arrange on levels as follows.

At the first level n basic ZC (BZC) of length n are placed, and i -th BZC is connected with TZC by means of pairs oppositely nonzero edges accordingly e^i_{in} и e^i_{out} ($i = 0, \ldots, n-1$). All remaining ZC's, that according to [4], [5] we will also named as peripheral ZC's (PZC's), are allocated on the levels in the following method.

At the second level that PZC is placed which is connected to one of BZC's by means of pairs opposite nonzero edges accordingly e^i_{in} и e^i_{out}. At the ($\tau+1$)-th level each PZC has the nonzero edges with the ZCs at the τ -th level and there are no the nonzero edges with the ZCs at the ($\tau-1$) and less levels $(\tau = 2,3,\ldots)$.

Similar cyclic structure of TD can be received if to consider interrelation of the vertices of the graph G_{FA} by means of any of nonzero edges e^{in}_i ($i = 1 \div n-1$). Always there is one vertex v^i_{null} for which entering e^{in}_{null} and leaving e^{out}_{null} zero edges are united and form a cycle, that is j -th trivial nonzero cycle (TNC j). At the first level n basic nonzero cycles ZC (BNC) of length n are placed, and j -th BNC is connected with corresponding TNC by means of pairs oppositely nonzero edges accordingly e^i_{in} and e^i_{out} ($i = 0 \div n-1$).

Further peripheral nonzero cycles (PNC's), are allocated on the levels in the following levels thus: at the ($\tau+1$)-th level j -th PNC j is connected with nonzero cycles at the τ -th level by means of pairs oppositely nonzero edges accordingly e^i_{in} и e^i_{out} and there are no such nonzero edges with the nonzero cycles at the ($\tau-1$) and less levels $(\tau = 2,3,\ldots)$. Always nonzero edges inside of nonzero cycles differ from nonzero edges which connect nonzero cycles of different levels.

Thus, from same vertices of the graph G_{FA} can be obtained n variants of multilevel graphical model of codes RS.

II. Algorithm of correction of full error bursts for RC codes

Definition 1: The sparse error burst Λ^{τ}_{sp} of length τ is a burst where the first error is in position ν and its last error is in position $(\nu+\tau-1) \bmod n$, but inside of the burst there can be correct symbols ($\nu = 1 \div n$).

Definition 2: The full error burst Λ^{τ}_{fl} of length τ is a burst where all erroneous symbols in the burst are located successively and differ from correct on an identical constant.

In coding theory the sparse error burst are considered only. Full error bursts Λ^{τ}_{fl} can be considered as well as the special case of sparse error bursts Λ^{τ}_{sp}, however the full error bursts are selected as a separate class because there are special methods of error correction for its. Further the full error burst will be considered only.

Let's consider $C(x)$ as a codeword without errors, and $C^{b,\nu}_{sol}(x)$ as the codeword containing a full error burst of length b with the beginning in a component ν. Then the full error burst can be presented by means of a error vector

$$E^{b,\nu}_{sol}(x) = C(x) + C^{b,\nu}_{sol}(x) .$$

Let's consider interpretation of the errors from the point of view of graphical models described above. Under influence of the full error burst there will be a transition from TZC to the vertex v_{err} of some ZC which we shall name as the error ZC. The transition path to error ZC which we shall name as code path consists of set of zero and nonzero adges between the neighbouring vertices. Under influence of the error burst Λ^b_{sol} in the beginning the transition on continuous set of nonzero adges is carried out inside of the corresponding j -th BNC and then the transition on continuous set of zero adges is carried out inside the error ZC.

The vectors $E^{b,\nu}_{sol}(x)$ and $C^{b,\nu}_{sol}(x)$ correspond to two various code paths from the vertex v^0_{null} (i.e. TZC) to the vertex v_{err} in the error ZC. So far as for these paths the initial and final vertices coincide, therefore they are equivalent. The code path corresponding a vector $E^{b,\nu}_{sol}(x)$ is more convenient for the analysis as its first part (τ nonzero adges) is located inside of j -nonzero cycle PNC j , and other part ($n-\tau$ zero adges) is located in that ZC which has the common vertex from PNC j . This ZC is the error ZC.

Let's consider interpretation of the error burst from the point of view of automatical model of LFSM. Under the influence both the vector $C^{b,\nu}_{sol}(x)$ and the vector $E^{b,\nu}_{sol}(x)$ the LFSM will pass from initial state $S(0)$ over to the state $S^{b,\nu}_{sol}(n)$, which we will name as the error syndrome of the full error burst. It is not difficult to show, that the vertex v^0_{null} in the graph G_{FA} corresponds to initial state $S(0)$ and vertex v_{err} – to the state $S^{b,\nu}_{sol}(n)$ of LFSM.

The problem of correction of the full error burst in RS code consists in the obtaining of the error burst vector $E_{sol}^{b,\nu}(x)$ that is equivalent to search of a code path in the graph G_{FA} from the vertex v_{err} to the vertex v_{null}^{0}. So far as there is a interdependent conformity between the vertices of the graph G_{FA} and the LFSM states, therefore this problem can be solved in terms of the automatical model of LFSM with the help of the formula (2). It is possible to suggest the following algorithm of correction of a single full error bursts.

Algorithm:

Input: – codeword $C_{sol}^{b,\nu}(x)$ with full error burst;

Output: corrected codeword $C(x)$.

1. Compute the syndrome $S_{err}^{b,v}(n)$:

   1.1 For i from 1 to n perform the following:

   $S(i) = A \times S(i-1) + B \times U[i]$,

   where $U[i]$ – i -th bit position of the codeword $C_{sol}^{b,\nu}(x)$;

   1.2. Assign $S_{err}^{b,v}(n) = S(n)$.

2. Assign $S(0) = S_{err}^{(b)}(n)$, $Z(0) = B$.

3. For i from 0 to $n-1$ perform the following:

   3.1 Compute the vector $Z(i+1)$:

   $Z(i+1) = A \times Z(i) + B$, $GF(q)$.

   3.2 For h from 0 to $n-1$ perform the following:

   3.1.1 For j from 0 to $n-1$ perform the following:

   3.1.1.1 Compute the vector $S(j+1)$:

   $S(j+1) = A \times S(j)$, $GF(q)$.

   3.1.1.2 If $S(j+1) = Z(i+1)$, then go to p. 5.

   3.1.2 Form the new vector $Z(i)$, in which the bit position z_w compute the following:

   $z_w = (z_w + 1) \bmod n$, .

   $z_w \in Z(i)$, $w = 1 \div n - k$.

4. Codeword $C_{fl}^{\tau,\nu}(x)$ contains noncorrected configuration of the error burst. Go to step 8.

5. Form the first error vector $E_{sol}^{b,\nu,1}(x)$ which corresponds to full error burst of length $\tau_1 = i+1$ with its beginning in position $\nu_1 = (j-i+1) \bmod n$.

6. Form the second error vector $E_{sol}^{b,\nu,2}(x)$, which corresponds to full error burst of length $\tau_2 = (n-\tau_1) \bmod n$ with its beginning in position $v_2 = (\tau_1 + v_1) \bmod n$.

   Nonzero categories are equal both error vectors to a degree α^h of the primitive element α over $GF(q)$.

7. Correct the codeword $C_{sol}^{\tau,\nu}(x)$ in accordance to the first error vector $E_{sol}^{b,\nu,1}(x)$ or to second error vector $E_{sol}^{b,\nu,2}(x)$:

   $C(x) = E_{sol}^{b,\nu,1}(x) + C_{sol}^{b,\nu}(x)$, $GF(q)$

   or

   $C(x) = E_{sol}^{b,\nu,1}(x) + C_{sol}^{b,\nu}(x)$, $GF(q)$

   ,

8. End.

As well as for a case of full error bursts over $GF(2)$ [5] in nonbinary Galois field we shall always deal with two indistinguishable full error bursts which give an identical error syndrome. In practice as more probable the error burst of the shortest length is chosen.

III. Conclusions

In works [6],[7] methods are shown for correcting full vector symbol error bursts of length $n-k-1$ or less with special restrictions (linear independent of the error vectors, some properties of parity check matrix). Known methods require many badly formalized operations with greater data.

The suggested algorithm allow to correction of a single full error bursts of length $(b = 1 \div n-1)$ and does not require the operations of interleaving and deinterleaving. This algorithm does not using a any type of full error bursts too but it has complexity $O(n \times b)$ concerning the operations of computing of the LFSM states.

The offered multilevel graphical and automatical models of LFSR defines the correcting capabilities of RS codes and allows to provide uniform approach to correction of errors of various types (random errors, error bursts and erasures). Based on LFSM theory the procedures of errors search of various types allow to perform it in parallel. Thanks to use only one operation of recursive calculation of the next LFSR state the suggested algorithm easily is interpreted to an architecture of the high-efficiency computing systems which use principles of matrix-conveyor data processing.

References

[1] B. Sklar, Digital Communications. Fundamentals and Applications. Second Edition, Prentice Hall, Los Angeles, 2001.

[2] T. L. Booth, Sequential Mashines and Automata Theory. New York: Wiley, 1968.

[3] A. Gill, Linear Sequential Circuits (Линейные последовательностные машины). Moscow, USSR: Nauka, 1974. (Russian translation).

[4] V. P. Semerenko, “Parallel Decoding of Bose-Chaudhuri-Hocquenghem Codes,” Engineering Simulation, vol. 16, pp.87-100, January 1998.

[5] V.P. Semerenko Burst-Error Correction for Cyclic Codes. Proceeding of International IEEE Conference EUROCON2009, S.Petersburg, Russia, pp.1646-1651.

[6] G. Chen and P. Owsley, “A Burst-error Algorithm for Reed-Solomon Codes,” IEEE Trans. Inform. Theory, vol. 38, NO. 6, Nov, 1992, pp.1807-1812.

[7] J. J. Metzner, “On Correcting Bursts (and Random Errors) in Vector Symbol (n, k) Cyclic Codes,” IEEE Trans. Inform. Theory, vol. 54, NO. 4, April, 2008, pp.1795-1807.