

# Review of a Coding Scheme in Information Storage Techniques

R. Chemlal

Badji-Mokhtar University, Faculty of Sciences  
Laboratory of Applied Mathematics, BP 12, Annaba, Algeria

I. Djellit

Badji-Mokhtar University, Faculty of Sciences  
Laboratory of Applied Mathematics, BP 12, Annaba, Algeria

## Abstract

This paper deals with coding scheme, first step in storage of information. Information storage techniques use one and two dimensional piecewise maps, this method consists in associating stable and unstable cycle of the map with each information to be stored, in this process the coding scheme used is of capital importance..

**Mathematics Subject Classification:** 37E05, 11H71, 37N30, 37N40

**Keywords:** Piecewise maps, Coding scheme, Dynamical Systems in optimization.

## 1 Introduction

One-dimensional maps are simple models with complicated nonlinear dynamics. They are used in studying attractors, in investigating of chaotic dynamics, and in analyzing information processing. Dynamics appears in iteration of the maps, and the information carriers are dynamical attractors: stable or unstable limit cycles, or strange attractors. In these systems the recognition problem can be solved, in the sense of the search for the reference most close to the presented one.

Mathematical models based on piecewise linear maps are used as a storage method, these maps can be one-dimensional or two dimensional, continuous or piecewise linear.

The precursor works are those of Dmitriev [5-7], Andreyev [1-3]. They aimed at the problem of information process in dynamical systems. This technology of the use of dynamical systems for storage implies that the privileged regime to code information are fixed point attractors, a piece of a trajectory, a part of the phase space, this information is stored as equilibrium states which play the role of the information carriers. The importance of these applications has served to focus the coding theory community on the complexity of coding techniques and then coding theorists have attempted to construct structured codes.

Many techniques are proposed in order to deal with the problem of repeated symbols, the "  $q$ -level" storage consists in associating a cycle to a sequence of  $q$  symbols which permits to deal with information blocks that do not contain a series of  $q$  symbols. The orthogonalization of data is used as a complement to previous methods; this consists in considering a repeated sequence as a new symbol.

Another solution is the use of two-dimensional maps to lighten the constraints on repetition and for high information capacity. Indeed since their appearance (techniques) in 1991, the methods were developed, their capabilities were essentially extended by means of using multi-dimensional maps as the information store-houses.

A growth of interest in memorizing, storing and recognizing information saw day this last decade. Information theory, coding and cryptography are the three load-bearing pillars of modern digital communication systems. All the three topics are vast, and there is a vast literature that deals with these topics individually. Effective use of coding demands appropriate algorithms for processing large amounts of data. These reasons prompt to focus our attention on interrelations of nonlinear dynamic systems and information processing. One of the reasons is stipulated by the fundamental results of the development of dynamic system theory, which are formulated as if they dealt with objects related somehow with information. For example, in [5,6] Dmitriev speaks about the existence of finite set of cycles with commonly fixed structure in dynamic systems, e.g. one-dimensional maps. In this paper, an attempt has been made to discuss two important concepts of coding process according to the papers [8-10] of Rouabhi : optimization-criterion for the choice of the coding parameter, and the repetition problem. An innovative algorithm is described and this is intended as a simple paper on the subject of simulation.

The information to be stored is represented in the form of discrete sequences composed of the elements of a finite-length alphabet. **In the original version** [8, 9], one-dimensional maps have the following form:  $x_{n+1} = f(x_n)$ , where  $x_{n+1}$  is called the image of  $x_n$  and  $x_n \in [0, 1]$ .

In the case of one-dimensional map the components must be all different. If the block of information contains coinciding terms, the process of storage will be impossible, the cycle will overlap and the retrieval of the stored information will not occur.

The plan of the paper is as follows. The next section deals with the concept of information coding and its efficient representation. The third section describes the coding method (filter method) proposed by [9] which permits, according to the author, to associate with each information block a period- $k$  cycle which points are all different, even if the information to be stored contain coinciding symbols and supposed to overcome simply the problem of repetition of a symbol in information blocks encountered in previous publications, we show how to solve the repetition problem, and the results obtained in term of the encoding capacity are summarized. In section 4, our concept of coding is introduced and a criterion of optimization is discussed. This paper concludes with some numerical simulations.

## 2 Information coding

Coding an information block of length  $k$  consists in associating to this information block a vector  $X \in [0, 1]^k$ , the components of  $X$  must be all different. This coding process is a key part which will make the memorization possible or not.

Coding an information block even containing some identical symbols must give a vector where all elements are different, in particular this implies that :

- Two identical symbols of a same block must correspond to different codes, because a cycle of period  $k$  contains  $k$  different points.

- Two identical symbols of different blocks must correspond to different codes because two cycles can't have a common point.

Consider an alphabet composed of  $N$  symbols  $(\sigma_0, \dots, \sigma_{N-1})$  and an information block of length  $k$  to be stored, the  $(i + 1)^{th}$  symbol of the alphabet is associated with the point  $x_i = \frac{i}{N-1}$  ;  $i = 0, \dots, N - 1$ . The information block so formed defines a vector  $X = [x_0, \dots, x_{k-1}]^T$  the components of which belong to the interval  $I = [0, 1]$ . Each point  $x_j$  is associated with the symbol  $\sigma_j$ .

An interesting coding scheme was proposed in [8,9], based on a multiplication of the information block by a particular matrix and the possibility to choose a parameter  $a$  such that all the components of the coded vector are different. Let  $X$  be an information block of length  $k$ , we define a new coded vector  $X$

by

$$X' = A^{-1}(k) .X$$

where  $A[k] = [A_{ij}]$  is a square matrix of rank  $k$ , It is chosen in the following form :

$$A_{ii} = a, \quad A_{j,j-1} = (1 - a), \quad A_{1k} = (1 - a), i = 1..k, j = 2..k$$

The inverse matrix of  $A$  is given by

$$A^{-1}(k) = \begin{pmatrix} \frac{a^{(k-1)}}{a^k - (a-1)^k} & \frac{(a-1)^{(k-1)}}{a^k - (a-1)^k} & \frac{(a-1)^{(k-2)} a}{a^k - (a-1)^k} & \cdots & \frac{(a-1) a^{(k-2)}}{a^k - (a-1)^k} \\ \frac{(a-1) a^{(k-2)}}{a^k - (a-1)^k} & \frac{a^{(k-1)}}{a^k - (a-1)^k} & \frac{(a-1)^{(k-1)}}{a^k - (a-1)^k} & \cdots & \frac{(a-1)^2 a^{(k-3)}}{a^k - (a-1)^k} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{(a-1)^{k-2} a}{a^k - (a-1)^k} & \frac{(a-1)^{k-3} a^2}{a^k - (a-1)^k} & \frac{(a-1)^{k-4} a^3}{a^k - (a-1)^k} & \cdots & \frac{(a-1)^{k-1}}{a^k - (a-1)^k} \\ \frac{(a-1)^{k-1}}{a^k - (a-1)^k} & \frac{(a-1)^{k-2} a}{a^k - (a-1)^k} & \frac{(a-1)^{k-3} a^2}{a^k - (a-1)^k} & \cdots & \frac{a^{(k-1)}}{a^k - (a-1)^k} \\ \frac{a^{(k-1)}}{a^k - (a-1)^k} & \frac{a^{(k-1)}}{a^k - (a-1)^k} & \frac{a^{(k-1)}}{a^k - (a-1)^k} & \cdots & \frac{a^{(k-1)}}{a^k - (a-1)^k} \end{pmatrix}$$

With  $a$  a real positive parameter chosen such that the coding scheme gives a vector without repeated symbols. Let us consider the family of vectors of the form:  $XP := [XP_i]_{i=1..k}$  that satisfies

$$\exists q / k : \forall i \quad XP_{i+q} = XP_i$$

All these vectors possess a same sequence of  $q$  symbols repeated  $n = k/q$  times, we will call them periodic information blocks.

In a recent article [4] we have shown that periodic information blocks can't be coded with popular coding schemes, in particular for any value of  $a$  the result of multiplying the matrix  $A$  by a periodic information block will give a vector with at least two coinciding terms.

We have shown inversely that the only information blocks for which the coding scheme described above fails are periodic ones.

### 3 Breaking the periodicity.

To improve the filter method and deal with periodic information blocks , we will try to break the periodicity form of these vectors.

Let be  $(x_i)_1^n$  a periodic word i.e. it exists  $q$  such that  $x_{i+q} = x_i$ .

Breaking the periodicity can be viewed as a transformation of one of the repeated subsequences  $x_1, \dots, x_q$ , any transformation that changes the order of the appearing terms will break the periodicity of the information blocks, if this transformation is invertible it becomes possible to recover the original information block.

A criterion choice for a periodicity breaking is to take two elements say  $x_i, x_{i'}$  from  $x_1, \dots, x_q$  and to permute them this leads to a new non periodic vector, which is a good solution that prove successful in simulations.

In order to implement the permutation efficiently, we performed the permutation as a multiplication of the coded vector by a permutation matrix as defined below.

Let be the matrices  $E_{ij}$  the identity matrix  $I_n$  for which lines  $i$  and  $j$  have been permuted, the matrices  $E_{ij}$  are called elementary Perlis operators, they are regular any elementary transformations (permuting lines or columns) can be achieved by a multiplication by one of these matrices.

## 4 A criterion choice for the parameter $a$

In this section we will study the criterion that will enable us to choose the optimal values of the parameter  $a$ , Let  $E = \{z_i, i = 1..q\}$  the set of the elements of some coded vectors, the memorization will occur iff  $z_i \neq z_j, \forall i \neq j$ .

We can define the function *MinDist* as the sum of the quadratic distances between two elements of  $E$  without repetition

$$MinDist(a) = \sum_{i \neq j, i > j} (z_i - z_j)^2$$

The optimal value of  $a$  correspond to maximal value of the function *DistMin*, the function *MinDist* is a fractional function and we have the identity

$$\forall i, j : A_{ij}^{-1} \left(\frac{1}{2}\right) = \frac{\left(\frac{1}{2}\right)^{k-1}}{\left(\frac{1}{2}\right)^k - \left(-\frac{1}{2}\right)^k}$$

Any element  $z_i \in E$  can be written in the form  $z_j = \sum A_{ij}^{-1} X_i$  where  $X_i$  are the **elements** of the initial (encoded) vector  $X$ , so the function *MinDist* ( $a$ ) is a rational fraction and its nominator is a combination of elements of the form  $\left(A_{ij}^{-1} - A_{i'j'}^{-1}\right)^2 x$

DistMin is a derivable function and there is two cases possible either  $\frac{1}{2}$  is a zero of  $DistMin'$  or  $DistMin'$  has an infinite limit at the point  $\frac{1}{2}$ , in the two cases, this correspond to a maximal value.

However  $\frac{1}{2}$  is only a theoretical point of view as for the coding to occur we need  $a > 1$  to have a distribution of points in the interval  $[0, 1]$ .

## 5 Numerical simulation.

In this section we will show some numerical results obtained by the modifications we have made,

Original method	
Information block	Coded Vector
[7, 5, 7, 5, 7, 5, 7, 5]	[.5873, .5873, .5874, .5874, .7460, .7460, .7460, .7460]
[7, 4, 9, 8, 7, 4, 9, 8]	[.5031, .5032, .7970, .7970, .8936, .8936, .9170, .9172]
Modified method	
Information block	Coded Vector
[7, 5, 7, 5, 7, 5, 7, 5]	[.5609, .5873, .5874, .5916, .7417, .7460, .7467, .7717]
[7, 4, 9, 8, 7, 4, 9, 8]	[.5033, .5192, .7347, .7981, .8936, .9000, .9172, .9556]

In our simulation we have used a maple based code, where we define the permutation as a multiplication by a Perlis matrix, here is a sample of the maple code who permits to simulate the technique proposed in this article.

Appendix : a Maple Code

```
>restart:with(linalg):with(LinearAlgebra):Digits:=4:k:=8:
> X:=vector(k):XX:=vector(k):X1:=vector(k):
> for i from 1 to k do
> X1[i]:=readstat('give_elements_of_the_vector_to_code'):
> X[i]:=evalf(X1[i]/9):od:
> Perlis:=matrix(k,k):
> for i from 1 to k do for j from 1 to k do Perlis[i,j]:=0:od:od:
> for i from 1 to k do Perlis[i,i]:=1:od:
```

```

> Perlis[1,1]:=0:Perlis[2,2]:=0:Perlis[1,2]:=1:Perlis[2,1]:=1:

> X:=multiply(Perlis,X):

> alpha:=1.4: A:=matrix(k,k):

> for i from 1 to k do for j from 1 to k do

> if (j<=i) then

>
A[i,j]:= (((alpha-1)**(i-j))*(alpha**(k-i+j-1)))/((alpha**k)-((alpha-1)**k)):

> else

>
A[i,j]:= ((alpha-1)**(k+i-j))*(alpha**(j-i-1))/((alpha**k)-((alpha-1)**k)):

> fi:od:od:

> XP:=multiply(A,X):
XX:=XP:XP:=sort(convert(XP,'list')):XC:=array[1..k-1]:

> for i from 1 to k-1 do

> XC[i]:= (XP[i]+XP[i+1])/2:od:

> P:=readstat('give the slope'):

> Cycle:=piecewise(x<
=XC[1],XP[2]+P*(x-XP[1]),x>XC[1] and x<=
XC[2],XP[3]+P*(x-XP[2]),x>XC[2] and x<=
XC[3],XP[4]+P*(x-XP[3]),x>XC[3] and x<=
XC[4],XP[5]+P*(x-XP[4]),x>XC[4] and x<= XC[5],XP[6]+P*(x-XP[5]),
x>XC[5] and x<= XC[6],XP[7]+P*(x-XP[6]),x>XC[6] and x<=
XC[7],XP[8]+P*(x-XP[7]), x>
XC[7],XP[1]+P*(x-XP[8])):Cycle:=simplify(Cycle):

> X0:=readstat("Initial condition ?"):

> n:=readstat("number of iterations ?"):

> for i from 1 to n do
X0:=eval(Cycle,x=X0):Xtemp[i]:=X0:od:

> Xtemp:=convert(Xtemp,array):

> for i from 1 to n do

```

```

> R[i]:=Transient\_Regime:for j from 1 to k do if (
Xtemp[i]=XX[j]) then R[i]:=X1[j]:

> fi:

> od:

> od:

> R:=convert(R,list);

```

## References

- [1] Andreyev Yu. V., Dmitriev A. S., Chua L. O., Wu C. W., "Associative And Random Access Memory Using 1-D Maps", *Int. Journal of Bifurcation and Chaos*, (1992) vol.3, N2, p. 483-504.
- [2] Andreyev Yu. V., Belkij Yu. L., Dmitriev A. S., "Storage of information using stable periodic points of two dimensional and multi dimensional maps". *Radiotekhnika i Elektronika*, (1994) vol. 39, n1, p. 114-123 (In Russian).
- [3] Andreyev Yu. V., Dmitriev A. S., Kuminov D. A., Pavlov V. V., "Information processing in 1-D and 2-D maps : recurrent and cellular neural networks implementation" *Proceedings of CNNA, Fourth International Workshop on Cellular Neural Networks and their Applications*, (1996) p. 297-301.
- [4] Djellit I., Chemlal R., "Coding Information and Problems of storage in Dynamical Systems", *FACTA UNIVERSITATIS, SER ELEC ENERG*, Vol 17, December 2004, 355-363.
- [5] Dmitriev A. S., "Chaos and Information Processing in Dynamic Systems" (In Russian). *Radiotekhnika i Elektronika* , (1993) vol. 38, n1, pp 1-24.
- [6] Dmitriev A. S., "Storing and Recognizing Information with One-Dimensional Dynamic Systems" (In Russian). *Radiotekhnika i Elektronika* (1991), vol. 36, n1, pp 101-108.
- [7] Dmitriev A. S., Kuminov D. A., Pavlov V. V., Panas A. I. "Storing and Processing texts in 1-D dynamic systems" preprint n3(585), Institute of Radioengineering and Electronics of RAS, Moscow, 1993 (in Russian)



- [8] Mira C., Millérioux G., Rouabhi S., "Two applications of noninvertible maps in communication and information storage", *European Journal of Operational Research* 139, 2002, p 461-468.
- [9] Rouabhi S., "Storage of Information in One Dimensional Piecewise Continuous Maps" *International Journal of Bifurcation and Chaos.* (2000), vol 10(5), pp 1127-1137
- [10] Rouabhi S., "Mémorisation d'information dans des récurrences unidimensionnelles et multidimensionnelles", *Thèse de Doctorat de l'INSA*, 2000.

**Received: February 24, 2007**