An introduction to coding theory of flow equations in ecological models

F. Vives¹, Y. Villacampa¹, M. Cortés¹ & J.L. Usó²
¹Departamento de A.M & Matemática Aplicada, Universidad de Alicante, Spain
Email: villacampa@ua.es
²Departamento de Matematicas, Campus Riu Sec., Universidad Jaime I, Castellón, Spain

Abstract

The modelling of dynamic systems depends on the solution of a differential equation system. Some problems appear because we do not know the mathematical expressions of the said equations. This problem is common to the modelling of the socio-economic structural systems. Enough numerical data of the system variables are known. The authors have used a methodology and have developed a language which was explained in other papers (Usó et all [4], Villacampa et all [6]).

The authors, going on with their research-line, think that it is very important to establish a code between the different languages to let them codify and decodify information. Coding permits us to reduce the study of some objects to others. The mathematical expressions, which model some system variables, are complex, so it is advisable to define a code alphabet, which can establish the correspondence between the equations and alphabet words, what will lead us to best manipulation of the said equations.

In this paper the authors begin with the introduction to the coding and decoding of complex structural system modelling. In the first place the authors start with a code between two languages to carry them through in order to obtain the simple expression of a mathematical equation with models a system variable that can be decodified from a further manipulation.

1 Introduction

Some mathematical equations, which analyse and study some process, are obtained in the risk system modelling. Due to the importance to simulate several situations in this system, it is advisable to dispose of a tool which allows to compare the models. Therefore, the convenient equations to be retrieved and to
be manipulated will need be stored and saved. In this respect, it is necessary to codify all language words used in the mathematical modelling L(C) and developed in the paper "A computational algorithm of language L(C), in order to construct mathematical models" [5]. It is impossible to store all the equations that take part in the selected intermediate process because the process complexity and equation number may be excessive.

2 Recognition code of flow equations

Let a risk system and a variable "y" be by representing a certain process to study. The flow equation which it defines is:

\[ y = F(x_1, \ldots, x_n) \]  (1)

In this paper, the authors define the equation modelling by its mathematical expression.

The mathematical modelling of these equation depends on the used methodology. The equation (1) will be defined mathematically in a language, independently of the used method. The authors have described the alphabet of the said language generated from elemental functions:

\[ \{f_i\} \]  (2)

in the paper "A computational algorithm of language L(C), in order to construct mathematical models" [5]. The flow equation (1) is given by lineal combinations of transformed functions generated from (2). When the elemental functions are chosen the flow equation can be modelled as it is indicated on the next example:

\[ y = a_0 + a_1f_1(x_1) + \ldots + a_n(f^2f_2)(x_n) \]  (3)

The flow equation modelling is complex in risk systems, thus it is necessary to express it with a symbol \( \Psi \) and a code that allows us to obtain the corresponding mathematical expression of immediate way.

Then, a symbol source alphabet \( \Psi_a \) to represent the flow equations will be defined, a code alphabet made up of elemental functions including the identity function and some codification rules.

3 Coding theory of flow equations

3.1 Definitions

Let an alphabet \( \mathcal{U} \) consisting of a finite numbers of letters be:

\[ \mathcal{U} = \{ \varphi_a, \varphi_b, \ldots, \varphi_l \} \]
Each symbol \( \varphi_i \) has a subscript, \( i \), formed by a string of \( m \) numerical characters (\( m \) is the upper order of the used transformed equations by the structural complex model). We call alphabet \( U \) the “transformed equations alphabet”.

We call a finite string of symbols

\[
\Psi_{i1i2...in} = \varphi_{i1} \varphi_{i2} \ldots \varphi_{in}
\]

da word in \( U \), and the value \( n \) its length (to be denoted by \( l(\Psi_{i1i2...in}) \)). Let \( S = S(U) \) be the set of all non-zero words in \( U \), and \( S' \) a subset of \( S \). \( S' \) is the set of words chosen by \( L(C) \).

The object generating words from \( S' \) is called a message source, and the words from \( S' \) messages. The words \( \Psi_{i1i2...in} \) are flow equations.

Suppose that an alphabet \( B \)

\[
B = \{ f_0, f_1, f_2, \ldots, f_q \}
\]
is given. \( f_0 \) is the identity function, that is to say \( f_0(x) = x \) and \( f_j \), \( j = 1, 2, \ldots, q \) elementary functions.

Denote by \( B \) a word in \( B \), and by \( S(B) \) the set of all non-zero words in \( B \).

Let a mapping \( F \) associating the word

\[
\Psi_{i1i2...in} \in S'(U)
\]

with each word \( \Psi_{i1i2...in} \in S'(U) \) be given.

We call \( B \) the message code, and the transition from the message \( \Psi_{i1i2...in} \) to its code incoding.

In coding theory, mappings \( F \) are given by an algorithm.

### 3.2 Alphabet coding

Consider the correspondence between the letters of the alphabet

\[
U = \{ \varphi_a, \varphi_b, \ldots, \varphi_r \}
\]

and certain words in the alphabet

\[
B = \{ f_0, f_1, f_2, \ldots, f_q \}
\]

viz.,

\[
\varphi \rightarrow f_0 \ f_0 \ldots \ f_0 \ f_i
\]
where $f_a f_b \ldots f_k$ means the composition of the functions, which is $f_a \circ f_b \circ \ldots \circ f_k$.

This correspondence is called scheme, and denoted by $\Sigma$.

It determines alphabet coding as follow: each word $\Psi_{i_1i_2 \ldots i_n} = \varphi_i \varphi_{i_2} \ldots \varphi_{i_n}$ from $S'(U)$ is associated with the word $B = B_1 B_2 \ldots B_m$, called the code for $\Psi_{i_1i_2 \ldots i_n}$, being each $B_i$ elementary codes of the scheme.

### 3.3 Example

**variables:** $x_1$, $x_2$, $x_3$.

**elementary functions:**

- $f_1(x) = \sin(x)$
- $f_2(x) = \log(x)$
- $f_3(x) = \exp(1/x)$
- $f_4(x) = x^2$

**upper order of the transformed equations:** 3

**Solution:**

a) **Alphabet source**:

$$U = \{ \varphi_{001}, \varphi_{002}, \varphi_{003}, \varphi_{004}, \varphi_{011}, \varphi_{012}, \varphi_{013}, \varphi_{014}, \varphi_{021}, \varphi_{022}, \varphi_{023}, \varphi_{024}, \varphi_{031}, \varphi_{032}, \varphi_{033}, \varphi_{034}, \varphi_{041}, \varphi_{042}, \varphi_{043}, \varphi_{044}, \varphi_{111}, \varphi_{112}, \varphi_{113}, \varphi_{114}, \varphi_{121}, \varphi_{122}, \varphi_{123}, \varphi_{124}, \varphi_{131}, \varphi_{132}, \varphi_{133}, \varphi_{134}, \varphi_{141}, \varphi_{142}, \varphi_{143}, \varphi_{144}, \varphi_{211}, \varphi_{212}, \varphi_{213}, \varphi_{214}, \varphi_{221}, \varphi_{222}, \varphi_{223}, \varphi_{224}, \varphi_{231}, \varphi_{232}, \varphi_{233}, \varphi_{234}, \varphi_{241}, \varphi_{242}, \varphi_{243}, \varphi_{244}, \varphi_{311}, \varphi_{312}, \varphi_{313}, \varphi_{314}, \varphi_{321}, \varphi_{322}, \varphi_{323}, \varphi_{324}, \varphi_{331}, \varphi_{332}, \varphi_{333}, \varphi_{334}, \varphi_{341}, \varphi_{342}, \varphi_{343}, \varphi_{344}, \varphi_{411}, \varphi_{412}, \varphi_{413}, \varphi_{414}, \varphi_{421}, \varphi_{422}, \varphi_{423}, \varphi_{424}, \varphi_{431}, \varphi_{432}, \varphi_{433}, \varphi_{444} \}$$

b) **Alphabet code**:

$$B = \{ f_0, f_1, f_2, f_3, f_4 \}$$

c) **Scheme**:
Example: the word $\Psi_{01003114} = \Phi_{011} \Phi_{003} \Phi_{114} \rightarrow f_0 f_1 f_0 f_0 f_3 f_1 f_4 =$

$= a_1 (f_1 o f_1)(x_1) + a_2 f_3(x_2) + a_3 (f_1 o f_1 o f_4)(x_3) + b =$

$= a_1 \sin(\sin(x_1)) + a_2 \exp(1/x_2) + a_3 (\sin(\sin((x_3)^2))) + b$

\[ \sum = \begin{cases} 
\Phi_{001} \rightarrow f_0 f_0 f_1 \\
\Phi_{004} \rightarrow f_4 f_4 f_4 
\end{cases} \]

4 Test for unique decipherability

Here we consider alphabet coding for two alphabets $U$ and $B$, specified by the scheme

\[ \Phi_{0010} \rightarrow f f f \ldots f f f \]

\[ \Phi_{0001} \rightarrow f f f \ldots f f f \]

\[ \Phi_{abc} \rightarrow f f f \ldots f f f \]

It is obvious that alphabet coding generates a mapping of the set $S(U)$ into the set $S(B)$. We denote by $S_\Sigma(B)$ the image of $S(U)$ under this mapping.

If the mapping of $S(U)$ onto $S_\Sigma(B)$ is one-to-one, then decoding is possible, we will say that the alphabet is one-to-one. i.e., it is possible to uniquely reconstruct from a code $B$ the original message with code $B$.

The decoding procedure is as follows:

Suppose that a word

\[ a_1 \log(\sin(x_1)) + a_2 (\exp(1/x_2))^2 + a_3 (\sin(\sin(x_3))) + b \]

is given.

Divide the word into elementary codes and replace each one by its correspondent letter in $\sum$:

\[ a_1 \log(\sin(x_1)) + a_2 (\exp(1/x_2))^2 + a_3 (\sin(\sin(x_3))) + b = \]
Then we observe that our alphabet coding is one-to-one and the decoding is possible.

5 Conclusions

The defined code application in the flow equation modelling allows a simplification in the storage process of the said equations. Thus, it will be simple to compare the flow equations obtained either in several modelings or simulations of the same model. This code has reduced the storage of flow equations wanted by the modeller. Its decodifying is possible because the unique decipherability test has been demonstrated. The results obtained in this paper will allow us to dispose of a good tool to obtain the best mathematical models.

References

1. Abramson, N. Teoria de la Codificacion y la Informacion. Ed. Paraninfo


