Fifth IFSA World Congress (1993), 869-872

PRINCIPLES FOR A CIPHER SYSTEM BASED ON CHAOTIC AND CHAOTIC FUZZY TOOLS

H.N. Teodorescu*, T. Yamakawa**

* *Polytechnic University of Iasi, Romania* ** *FLSI, Iizuka, Japan*

Abstract. The chaotic fuzzy logic systems behave in a more complex way than crisp chaotic systems, and they can show some advantages in complex applications. Such an application is introduced in this paper, namely in scrambling and ciphering the signals.

1. Introduction

Telephone voice scramblers for secure and confidential telephone communications are spreading recent years. Such scramblers convert normal speech into unintelligible electronic signals that are transmittted over the phone circuit. The scrambling is based on digital techniques and is controlled by user selectable codes (in general, more than 1000 available per system). Codes can be changed from conversation to conversation.

Fuzzy systems can present a chaotic-like regime. Obviously, such a regime is to be avoided in control systems and in any applications where the response has to be well defined (deterministic, i.e. crisp). On the other hand, both crisp and fuzzy chaotic systems can also have some applications. It is the purpose of this paper to introduce scramblers which can be implemented with both crisp and fuzzy chaotic systems.

In **Figure 1**, a chaotic continuous crisp signal and samples of a chaotic fuzzy signal are presented. In fact, the crisp

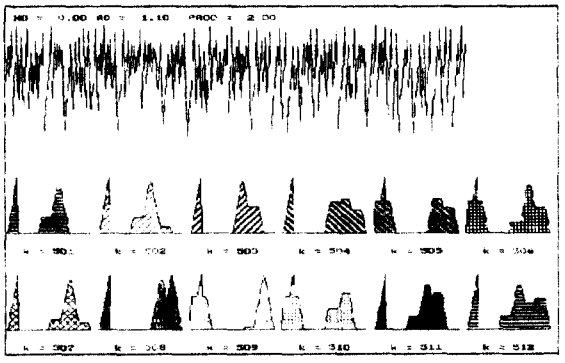

Figure 1: A chaotic crisp signal (up), and samples of a chaotic fuzzy signal (down).

signal is got by defuzzifying the fuzzy one. Note that the shape of the corresponding membership functions are evolving in a chaotic way with respect to time.

Due to the bidimensional character of the fuzzy signals, more definitions of the chaotic fuzzy signals -- all natural -- are possible. The signal can be chaotic with respect to the time variable, but not with respect to the variable of the membership function, or vice versa, or with respect to both variables. Moreover, the fuzzy signal

can be put into correspondence with a crisp signal that can be chaotic with respect to time. Such a correspondence can be generated by choosing a global parameter of the membership function - for example its center of gravity.

For applications, specific definitions got characterizing at any moment the corresponding membership function $\mu(to,x) = \mu(x)$ by a parameter -- e.g. the center of gravity -- are useful. Let us denote by w(t) the value of the chosen parameter of $\mu(t, x)$. One gets a (crisp) function w(t) as a representative function for $\mu(t,x)$.

Definition *The fuzzy signal $\mu(x,t)$ is chaotic iff there exists a parameter, denoted by w, of the membership function $\mu(x)$ such that the generated function w(t) is* ***chaotic***.

The chaotic behaviour for discrete signals and systems is defined by restricting the above continuous functions to discrete maps.

2. A configuration of chaotic fuzzy logic systems

There are many classes of fuzzy logic systems that exhibit chaotic behavior. Only one class of chaotic systems will be used in this paper for exemplification: the class of systems with the block diagram shown in Figure 2. For details, see [1], [2], [3].

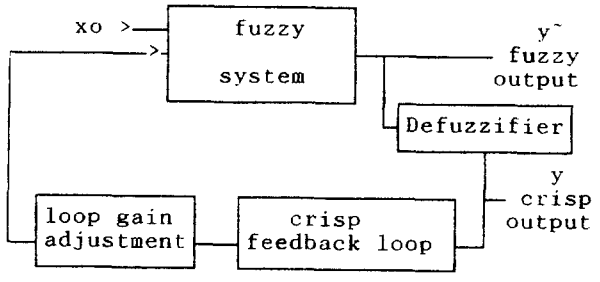

Figure 2: Block diagram of the chaotic fuzzy systems used

Such a configuration can be used to build up both discrete (i.e. discrete time) and continuous (i.e. continuous time) chaotic fuzzy systems.

3. Coding Principles

3.1. Coding based on a chaotic crisp system

When building a text ciphering system, the main idea is that the codes representing every letter in the alphabet have to have an as random as possible distribution (uniform distribution ensures the best ciphering). This can be achieved by using a random generator and by making some correspondence between the text and the random signal. Obviously, it is important that the correspondence is bijective, else the deciphering is impossible.

The main idea behind this ciphering method is to use a chaotic system as a random signal generator. Such a system can exhibit an almost uniform distribution of the output values. The second idea is that the deterministic nature of the chaotic systems allows for the realization of the bijective mapping text <--> output values, as necessary in decoding. This mapping can be established in different ways (see subsequent paragraphs for an example). Finally, the third idea introduced is that of using chaotic fuzzy systems to increase the complexity of information used in ciphering. Below, these ideas are briefly exemplified.

3.2. Coding based on a chaotic fuzzy logic system

The output of the system is addressed by the time variable t, and the initial state xo only, and thus it needs less information than it outputs carries. Only one couple of numerical values is needed to determine the output, and the fuzzy output y~ is a function, i.e. a bi-dimensional signal that contains a rich information, determined by the structure of the system: **{initial state (xo); time lapse after the start (No)} --> output membership function.**

Because of the chaotic behavior of the system, almost every possible continuous function is available at the output of the system in an infinite laps of time.

Following, the fuzzy chaotic system is the almost ideal coder because it is able

to output a large amount of information when addressed with a few information.

Consider for example the class of chaotic fuzzy systems described by the block diagram pictured in Fig. 2, with input and output triangular membership functions. Two different ways of using the output information are possible, according to the application in hand. If one needs a **wave-form memory / generator**, the output is already available as the membership function of $y^{\sim}$ (either discrete or continuous case).

In text ciphering, one needs strings of finite length words in the form:

$$w(t_k) = (a_{k1}, a_{k2}, \ldots, a_{kn})$$

where $w(t_k)$ represents the word generated (transmitted) at time moment t_k, and a_{kj} are letters (symbols) from the used alphabet.

To get an output string of finite words instead of an output bidimensional signal (function), the output function $\mu(t,x)$ has to be sampled with respect to both the time (t) and the x variables. (Sampling the output is not necessary if the fuzzy system is a discrete one). The number of samples with respect to the x variable determines the amount of output information obtained with just one set of address numbers: it determines the length of the word (the dimension of the output vector).

An important practical problem is as follows. Let be an alphabet A of n characters. Let be the set A_p of all words of p characters (this set includes or equals the vocabulary). Let the samples of the output membership functions be mapped into the set of characters. Being given a chaotic fuzzy system, is it able to generate, by appropriate sampling of the output membership function and by appropriate mapping of the samples into the set A, all the words from A_p ? Let the characters of this alphabet be mapped into a finite interval I. Then the problem is equivalent to: is the output of the system a compact interval in R ? If not, but still the output is a union of compact intervals, a mapping from the set of outputs to a compact interval is still possible. If this in not true, one may have the case: the output is a dense coverage of an interval (or of a set of intervals).

The following result can be proved, based on the fact that a chaotic fuzzy systems generalizes the crisp ones.

Proposition: *Any vocabulary (set Ap) based on words of finite length (finite number of characters), and based on a finite alphabet can be covered by at least one chaotic fuzzy system using the above algorithm.*

Let us note that the use of a single parameter of the membership function of the output -- e.g., the center of gravity, i.e. the defuzzified output -- instead of a set of samples of the membership function dramatically reduces the benefits of using a fuzzy chaotic system. Indeed, the same results can be got using just a crisp chaotic system.

3.3. A simple ciphering algorithm

Any ciphering algorithm can be used in conjunction with a chaotic fuzzy system as above, eg. simple or multiple substitutions algorithms. A very simple method, using just the correspondence:
letter <--> current number in the alphabet,
is presented below.

The initial state x_0 of the chaotic system, and the current number **No** of the first time moment took into account (see below) are used as keys, together with the system parameters.

Every letter in the text is coded as an output value of a chaotic fuzzy system (defuzzified output); the first letter is the value of the output at the **(No + k)**-th time moment, where **No** is part of the key, as above, **k** is the number order of that letter in the alphabet; the seconder letter is coded as the **[(No + k) +j]**, where **j** is the number of the second letter in the alphabet etc. The way of using more complex coding algorithms is obvious.

The block diagram of a communication system for ciphered data using chaotic

crisp or fuzzy systems is illustrated in Fig. 3. This diagram is still valid if analog implementations of the chaotic systems are used (e.g. chaos chips, either crisp or fuzzy, as designed by Yamakawa Laboratory).

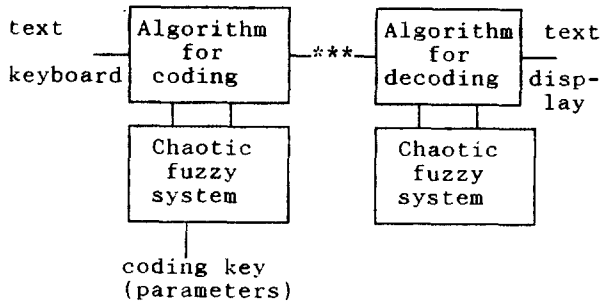

Figure 3. Block diagram of the ciphering system

The encrypted data transmission can be performed using two computers to simulate chaotic fuzzy systems, and also to perform the coding/decoding task. The transmitting computer ciphers data by means of a fuzzy-chaotic algorithm. Resulting information can be transmitted along telephonic lines using any modulation method, e.g. FM using PLL circuits, or by FSK modulation. At reception point, another computer deciphers and displays initial data.

4. Discussion and conclusions

A possible application of both crisp and fuzzy chaotic systems in communications was exemplified.

In contrast to deterministic chaotic systems, chaotic fuzzy systems allow an almost infinite data compression rate. This feature could contribute to improving the communications efficiency and to increase the number of transmissions per channel. It also offers a technically sounding mean for **'artificial intelligence - based broadcasting'**, as introduced at the theoretical level in [5].

According to the above described principles, a software was developed for text ciphering (by the present author in cooperation with S. Pavel), and a hardwired configuration is under development (by F. Grigoras, in cooperation with the first author).

Acknowledgement: Figure 1 is drawn with a program due to S. Pavel.

References

1. H.N. Teodorescu: Fuzzy oscillators. BUSEFAL, No .44/1990, pp.161 - 165
2. H.N. Teodorescu: Chaotic fuzzy systems. Research Report nr. 2-7135/1991, Iasi, 1991
3. H.N. Teodorescu: Fuzzy systems with feedback. Res. rep., 1-7135/1991, Iasi 1991
4. H.N. Teodorescu: Analysis of a class of chaotic fuzzy systems. In: H.N. Teodorescu and M. Ciobanu (Editors): Fuzzy Systems Engineering, Tiraspol, R. Moldova, 1991
5. H.N. Teodorescu et al.: AI-based broadcasting. IBC Conference. Brighton, UK, 1987.