Paper

A study on fractal dimensions and convergence in fuzzy control systems

Alfons Schuster, William Blackburn, and Miguel Segui Prieto

Abstract — This paper addresses a problem in the area of intelligent, knowledge-based systems, namely the generation of knowledge, by presenting a proposal for the automation of this task. The proposed approach is limited however by focusing on fuzzy control systems (FCSs). Results obtained from different experimental investigations indicate the potential of the approach.

Keywords* — *fuzzy control systems, chaos theory, convergence, fractal dimension.

1. Introduction

Knowledge is a central component in any intelligent, knowledge-based system. Problems obtaining or generating knowledge can arise from different sources. They may be due to the complexity of the domain, the accessibility and availability of domain knowledge or domain experts, the number of rules needed for a rule base, or the consistency and maintenance of such a base, for example [5]. Note that due to the focus of the paper the discussions here emphasise issues related to FCSs. Rules are not the only means by which knowledge is captured in a FCS. Fuzzy sets, their shape and arrangement, as well as the mechanisms by which they communicate in a system are also very important [15]. FCS design also very often has a strong trial and error nature in which the system designers very often play a vital role. One of the insights we gained from this underlying trial and error approach is that it is very often possible to generate multiple FCS solutions for the same problem. For example, the only difference between two FCS solutions could be the defuzzification technique they employ, but it also could be the slightly different shape of particular fuzzy sets. Another observation is that many FCSs show similarities in their dynamic behaviour. For example, the dynamic behaviour of a FCS application might look similar to the illustration given in Fig. 1.

This illustration actually is taken from an example application provided with the commercial fuzzy logic tool CubiCalc 2.0 that was used in this study. Note however that the circled line in the figure has been added manually to ease forthcoming discussions. Figure 1 illustrates the trajectories of two objects, A and B, moving from left to right in time. Y and X in the figure define a co-ordinate system. The objective of object B is to approach and finally catch object A. Both objects move with constant, but individual speeds, and so a dot or circle at position (X, Y) in the figure represents the position of an object in time. For simplicity object A moves on a straight line. Object B has to be more flexible due to the definition of its task. Note that although Fig. 1 illustrates two trajectories for object B, at the moment only the trajectory labelled with the number *1* is of interest. Figure 1 indicates that object B, following trajectory *1*, really approaches object A, and therefore provides a solution to the given task. It was mentioned earlier that this or a similar dynamic behaviour could be found in many other situations. Indeed, Fig. 1 could illustrate the movement of a robot arm trying to grasp an object on an assembly line, it could illustrate the control of the temperature in a room, but also the path of a remotely controlled vehicle on a planet approaching an object for probing, for example.

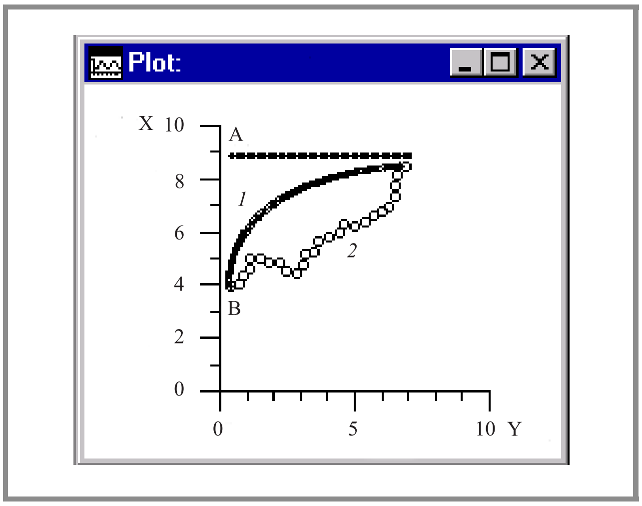

Fig. 1. Dynamic behaviour of an example FCS.

For later discussions it is also important to understand that with most commercial tools it is usually possible to record the values of selected variables (e.g., the X, Y positions of object A and B in Fig. 1) at each step in a time series. A time series therefore, in a sense, contains information about the dynamic behaviour of the system.

Another point that needs mentioning is illustrated by the second (circled) trajectory for object B in Fig. 1. Like trajectory *1*, this trajectory finally approaches object A, and thus, a FCS generating this trajectory could be regarded as a solution to the problem too. The solution finally selected however could be the FCS that produces trajectory *1*, because for many problems FCS designers prefer a system that converges towards a solution with some smoothness. Simply imagine the two trajectories in Fig. 1 as being proposed solutions for the robot arm mentioned before. It is not difficult to select the one more appropriate for the task. The following provides a synopsis of these observations:

- In the field of FCSs it is very often possible to generate multiple solutions for a problem.
- FCSs applications in different problem-solving situations show similarities in their dynamic behaviour.
- Convergence with a certain degree of smoothness can be a requirement in some FCS applications.

These observations form the basis for this work, which in broad terms can be summarised as a study investigating the similarities in FCSs, mentioned before. The means by which we aim to achieve this goal are:

1. The convergence of a proposed FCS solution is examined by a measure of convergence.
2. A fractal dimension algorithm on the other hand, determines the smoothness of a solution.

We investigated a number of FCSs and other models for quality assessment. The results generated in these studies indicate the potential of the approach. The reminder of the paper is organised as follows. Section 2 explains the measures we use in investigation in this study. Section 3 describes the FCSs and models we investigated. Section 4 presents the results from these investigations. Section 5 proposes an integrated system. Section 6 reviews related work, and Section 7 ends the paper with a summary.

2. Two measures used in this study

The measures introduced in this paper relate to some degree to what is sometimes loosely termed chaos theory. This theory has its origins in the study of nonlinear dynamical systems, and hence nonlinear differential equations [17]. It obtained increasing attention within the natural sciences about four decades ago. A key element being the fast progress in computer technology within this period [14]. With computes growing more and more powerful it was possible to investigate more and more complex systems with increasing efficiency. Lorenz, for example, investigated the extent to which weather is predictable [11]. Lorenz's work also produced an interesting by-product, the so-called Lorenz-attractors. The artistic beauty of many attractors led to an increasing awareness and popularity of the theory. Nowadays chaos theory is studied in many domains including medicine, engineering, and computer science, for example [3, 6, 8]. Out of these studies emerged a variety of new concepts and measures.

Before the two measures are explained in more detail we use the forthcoming section to discuss another concept from chaos theory that is important in the context of this paper, namely that of an attractor.

2.1. Attractors

Attractors, also often referred to as strange attractors, or fractals, are a very important concept in chaos theory. In chaos theory an attractor is more or less the state development of a dynamic system over time. The temporal development of these systems is often illustrated in so-called phase-state plots or phase diagrams. Very often these mathematically generated illustrations bear a striking similarity with structures we can find in nature. The shapes and forms of trees, lungs, shells, and clouds are typical examples [12]. To understand and connect mathematically generated attractors and fractals with these observations in nature is a strong motivation for the study of chaos theory, and so it is needless to say that a lot of work has been done in this area already. Little work however has been done on a particular view on attractors. In this particular view we suggest that the principle of an attractor appears quite frequently, often under different names, in our everyday life. The different expressions we use for the term attractor in many of these situations include the terms **goal**, **aim**, or **target**, for instance. The following two examples help illustrating this relationship. The goal of a person planning a holiday can be to be at a specific location over time. Or, the aim of an autonomous agent over time can be to avoid a number of obstacles. It is important to understand that the main objects (person, autonomous agent) in the two example systems move towards an **attractor**, or **goal**, or **aim** over time.

We humans are able to discuss natural structures, goals, and aims more or less elegantly via the use of our natural language. On the other hand, the language of chaos theory is mathematics. Although the mathematics of attractors and fractals can be relatively simple in some cases, it remains a fact that the mapping and interpretation of mathematical statements into the real world often can be very difficult, if not impossible. Let us therefore say:

- there is some sort of a gap between the mathematical world and the natural (problem-solving) world.

This observation makes this study in the area of FCSs interesting and promising, because by its very definition fuzzy logic provides a means for acting as a communicator between the mathematical world and the natural (linguistic) problem-solving world. Note also, that although the discussion here concentrates on attractors there are other concepts from chaos theory that are also very relevant in this context (e.g. self-similarity, and self-organisation) [16].

2.2. A measure of convergence

The previous section revealed that convergence could be an important feature in FCSs. For example, the task for object B in Fig. 1 was to approach and finally catch object A. The trajectories of the two objects consequently need to converge towards each other. The measure used in this study for distinguishing the convergence of different systems aims to reflect this behaviour. For example, let Fig. 2 illustrates the trajectories of two objects A and B.

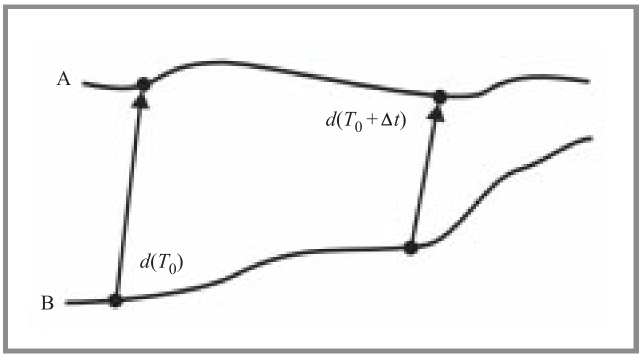

Fig. 2. Trajectories of two objects A and B.

In Fig. 2 $d(T_0)$ shall be the distance d between objects A and B at time T_0, and $d(T_0+\Delta t)$ the distance at time $t = T_0+\Delta t$. In order to use the distance development between the two objects over time we here define a measure of convergence (*MOC*) as follows:

$$MOC = \frac{1}{N-1}\sum_{n=1}^{N-1} \lg\left|\frac{d_{n+1}}{d_n}\right| . \qquad (1)$$

Note that the variable N in the equation stands for the number of data points in the time series. The features of this measure that could be useful in this study are:

- $MOC < 0$, may be an indicator for a system that produces convergent trajectories.
- $MOC = 0$, might indicate a system that is in some sort of steady state mode, for example, objects A and B moving on two parallel lines.
- $MOC > 0$, very likely an indicator for a system that produces non-convergent trajectories.

Here it could be interesting to refer to the so-called Lyapunov Exponent λ found in chaos theory. The Lyapunov Exponent is a measure to assist in the distinguishing between different types of orbits or trajectories of dynamic systems [10]. It is based on the mean exponential rate of divergence of two initially close trajectories, and describes the dynamic of a system qualitatively as:

- $\lambda < 0$, the orbit is attracted to a stable fixed point or a stable periodic orbit.
- $\lambda = 0$, the orbit is a neutral fixed point. The system is in some sort of steady state mode, like a satellite in a stable orbit, for example.
- $\lambda > 0$, the orbit is unstable and chaotic. Nearby points, no matter how close diverge to any arbitrary separation.

Although it is not the intention here to use the *MOC* for determining whether a system is chaotic or not it is interesting here to identify the similarity it bears with the Lyapunov Exponent.

2.3. A measure of smoothness

Section 1 suggested that convergence alone is very frequently not the only criterion when developing FCSs. Very often a solution should have certain smoothness too. The basic assumption is that, given different FCS solutions, a smoother trajectory is more likely to be selected than a trajectory that is rather jagged or irregular.

The study of so-called fractals may provide a possibility for quantifying the shape of a trajectory in terms of its smoothness, or jaggedness, respectively. Very generally, fractals are patterns or structures which, when being dealt with mathematically, produce results or properties that are difficult to be interpreted, or conflicting with predictions of traditional mathematics. An example would be the Koch-snowflake curve, a geometric object with finite area, but infinite circumference. Outstanding mathematicians attempted to come to grips with these objects. Mandelbrot for example, associates these pathological structures with forms that can be found in nature [12]. Hausdorff and Besicovitch on the other hand came forward with a general definition for the calculation of a (fractal) dimension for such objects. Their definition of a fractal dimension is based on an investigation of how geometric figures fill the space in which they are represented [4]. It is important here to mention that there exist many definitions for measurements on fractals. This paper, for instance, uses a method proposed by Gough for the calculation of a fractal dimension [7]. Also, remember that the geometric objects investigated here are time series representing the development of the distance between the trajectories of two objects. However, let the time series illustrated in Fig. 3 represents the distance development of an example system.

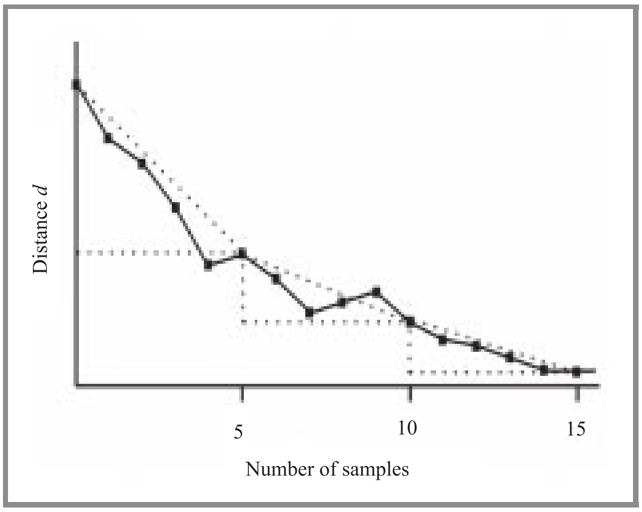

Fig. 3. Distance development of two trajectories, and length estimation using a ruler length of five.

Figure 3 illustrates that individual distance measurements are connected to a continuous line. Gough's method is used to calculate a fractal dimension from such a line. Initially the method determines different estimates of the length L of the line by measuring it with different so-called rulers of length r. The line in Fig. 3 for instance is measured with a ruler of length five. A length measurement for a particular ruler is determined by the following equation:

$$MOC = \frac{1}{N-1}\sum_{i=1}^{N-1}\left[\left\{r^2 + (x_{ir} - x_{(i+1)r})^2\right\}^{\frac{1}{2}} + \right.$$
$$\left. + \left\{(N - rk - 1)^2 + (x_{rk} - x_{N-1})^2\right\}^{\frac{1}{2}}\right].$$

In this equation $k = \mathrm{Trunc}\left[\frac{\mathrm{N-1}}{\mathrm{r}}\right]$, r represents the ruler length, and N the number of distance measurements in the time series. In simple terms a single length estimate (L_r) is a summation of hypotenuses. In order to extract a fractal dimension from such a diagram the method then plots the logarithm of the length estimates ($\lg L_r$) against the logarithm of the ruler length ($\lg r$). Figure 4 illustrates an example of such a graph.

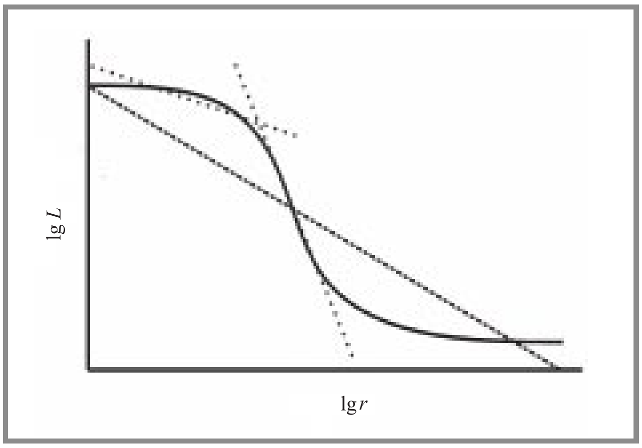

Fig. 4. Extraction of a fractal dimension.

The establishment of a fractal dimension from such a diagram is not that simple however. The traditional definition by Hausdorff and Besicovitch leads towards using the slope of a regression line (dashed line in Fig. 4) through the data points as an approximation for a fractal dimension. Other researchers came up with other interpretations. Kaye for example generates regression lines and fractal dimensions for separate regions in a plot (the two dotted lines in Fig. 4 for example), and compares these fractal dimensions with the features of “structure” and “texture” in fine-particle science [9]. This paper follows Kaye’s view, and so it could be said that a measurement with longer rulers identifies the global behaviour (structure) of the distance development between two trajectories. On the other hand, measurement with smaller rulers provides information about the behaviour of the distance function at smaller scales (texture).

3. Investigated systems

Figure 5 illustrates some of the systems we investigated. The systems will be referred to as System 1, 2, 3, and so forth. The first three systems are FCS applications taken from an example library that is included in the software tool that has been used in the study. System 1 has already been introduced in Section 1 and therefore a description of it is omitted here. System 2 is a FCS that controls the movement of a truck (B) that tries to enter a parking slot. Figure 5 illustrates three parking attempts. The starting position of the truck is always randomly selected. The parking slot (A) remains at position 50.0 on the x-axis. The three scenarios in Fig. 5 show that the trajectories produced by the truck always converge towards the parking slot. FCS System 3 faces the problem of trying to suspend a metal object (B) in air at a stable position midway between an electromagnet at height 10.0 and the ground (height 0.0). Figure 5 illustrates two attempts. For example, take the attempt where the initial position of the metal object is at height 7.0 between the ground and the electromagnet. The FCS controls the magnetic field generated by the electromagnet according to the position of the metal object between the magnet and the ground. The field is continually changed until object (B) is suspended midway (height 5.0) between the electromagnet and the ground. This position is labelled (A) in Fig. 5. The x-axis in the figure represents the number of iterations the FCS goes through over time. Figure 5 illustrates that the trajectories produced in both attempts represent a solution to the problem.

To make the study more comprehensive we investigated various other systems. Some of these systems, Systems 4 to System 12, are illustrated in Fig. 5. Each illustration in Fig. 5 contains two trajectories (y_1 and y_2), corresponding to the movement of two imaginary objects. For example, the first trajectory for System 4 is defined by the exponential function $y_1 = 100\,\mathrm{e}^{-0.02x}$, and the second trajectory by the function $y_2 = 0$. Note that apart from System 6 and System 9 the second trajectory is always defined as $y_2 = 0$. Note also that the range for the values along the x-axis is the same for these system, namely [0, 200]. Further, System 4, 5, and 6 illustrate systems exhibiting convergent behaviour, whereas System 7, 8, 9, 10, 11, and 12 are used to represent non-convergent behaviour. The non-convergent systems can be further divided. System 10 and 11 indicate objects moving along in parallel (System 10) or oscillating parallel (System 11). Trajectory y_1 of System 12 finally was generated randomly.

4. Results

Table 1 illustrates *MOCs* and fractal dimensions extracted from these systems using the techniques described before.

Column 1 in Table 1 indicates the system, and column 2 the number of data points in a time series. Column 3 holds the *MOC* for each system. Column 4 and 5 finally contain fractal dimensions. The two columns differ in using different sets of rulers for the measurement of a fractal dimension of a time series. For example, taking a system with 200 data points, Ruler 1 to 10 means that the time series has been measured with rulers of length 1, 2, …, 10. For the same

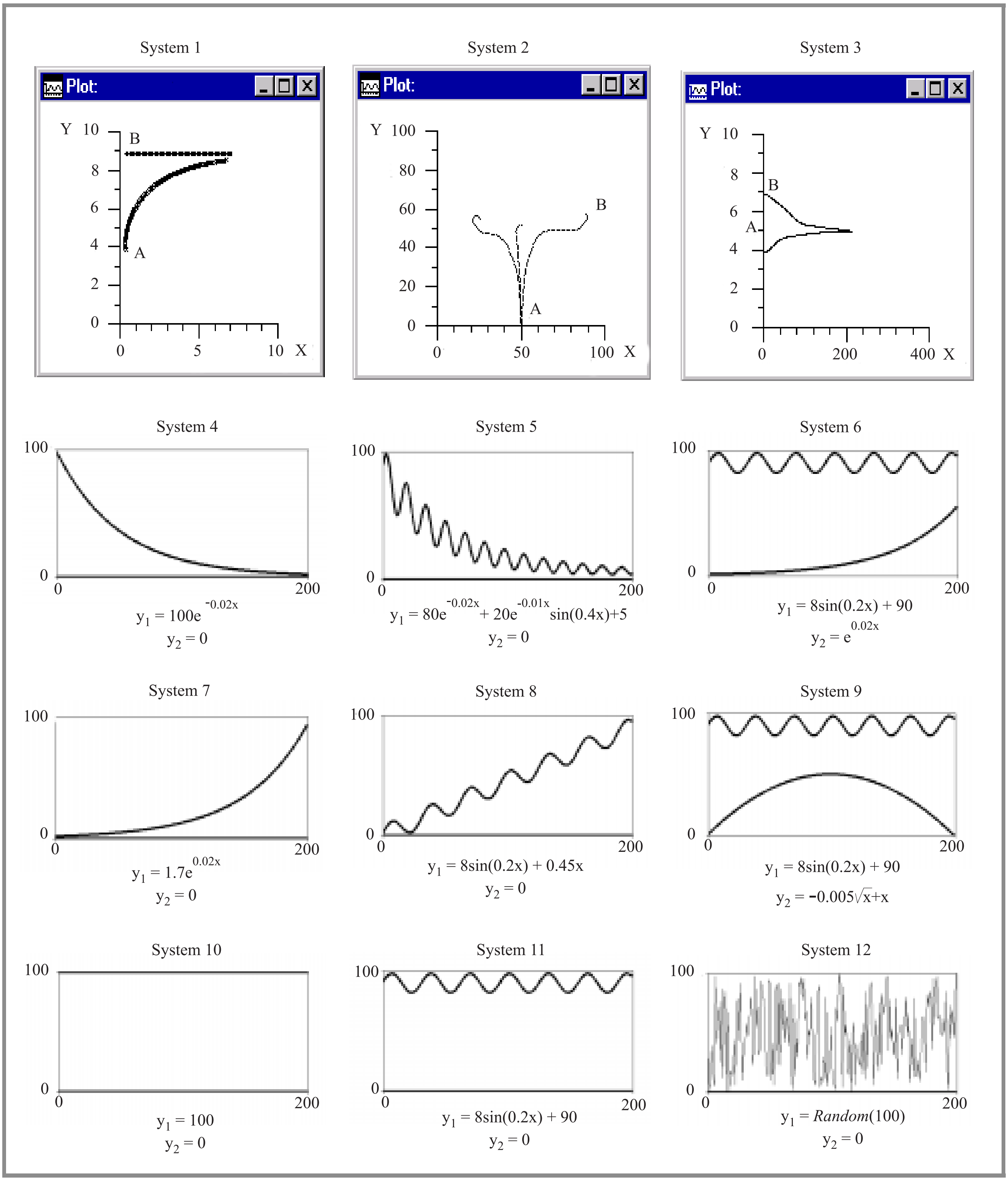

Fig. 5. Example systems studied in this work.

Table 1
MOCs and fractal dimensions of some of the example systems investigated in this study

System	Data points	*MOC*	Ruler (1 to 10)	Ruler (10 to 40)
System 1	75	-0.0280	1.0000	1.0000
System 2	61	-0.0342	1.0000	1.0010
System 3	200	-0.0070	1.0000	1.0000
System 4	200	-0.0086	1.0000	1.0021
System 5	200	-0.0069	1.0003	1.0022
System 6	200	-0.0017	1.0355	1.2250
System 7	200	0.0086	1.0001	1.0019
System 8	200	0.0084	1.0318	1.1776
System 9	200	0.0001	1.0355	1.2250
System 10	200	0.0000	1.0000	1.0000
System 11	200	0.0001	1.0355	1.2250
System 12	200	-0.0052	1.9754	1.6497
	200	0.0010	1.9510	1.6183

200 data points, Ruler 10 to 40 stands for a measurement with rulers of length 10, 11, ..., 40.

4.1. Discussion of results

Table 1 illustrates that the *MOCs* of the six convergent systems (System 1, 2, 3, 4, 5, and 6 in Fig. 5) are all negative. The *MOCs* of the non-convergent systems (System 7, 8, and 9) are all positive. The *MOC* of System 10 (parallel) is zero, and that of System 11 (oscillating parallel) is very close to zero. These results are encouraging, because the *MOC* so far separates convergent from non-convergent systems. They are also interesting when being compared with the qualitative interpretation of a Lyapunov Exponent in Section 2.1, where a negative exponent indicated stable systems, a positive exponent unstable systems, and one of zero systems that are in some sort of steady state. Table 1 however also reveals that it is possible to obtain positive as well as negative *MOCs* for different random systems (System 12). Initially this seems to be problematic, but the fractal dimension values in column 4 and 5 indicate a possible solution to this problem. Remember that the “preferred” solutions are less jagged and irregular, and so should have a smaller fractal dimension. The fractal dimension values for the two random examples clearly reflect this assumption. It is also interesting to see that the three FCSs, as well as System 4, 7, and 10 all have very low fractal dimensions (close to 1.000), which is corresponding to the smoothness they illustrate. The remaining systems, apart from System 5, all have higher fractal dimensions. Note that although this discussion refers to the values in column 4 in Table 1, an interpretation of column 5 leads to similar observation.

It was mentioned earlier that the systems in Fig. 5 are representative instances of a larger group of systems we investigated. For example, the *y*-axis for System 4 to System 12 in Fig. 5 is scaled from 0 to 100 in this paper, but we also have evaluated systems showing similar trajectories at different scales. The results established by these other systems did allow an interpretation similar to the interpretation given before. From the viewpoint of the motivation behind this paper the results established in this study therefore can be interpreted as quite positive and encouraging to undertake further research in this direction.

5. Proposal for an implementation

Figure 6 at the end of this section illustrates our vision of a system that could be capable to automatically generate components for the knowledge base of a FCS application.

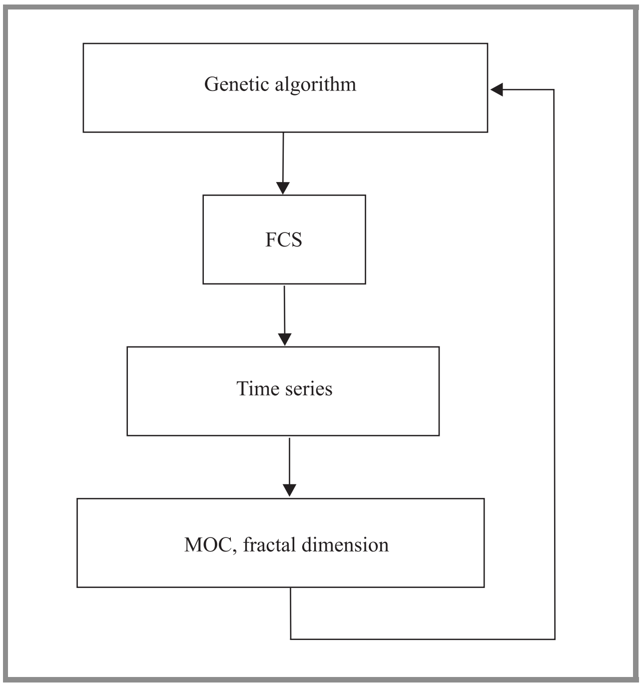

Fig. 6. Proposal for an implementation of the techniques presented before into a full system.

Figure 6 basically illustrates the integration of the methods presented in this paper with a genetic algorithm (GA). For example, the GA initially generates a pre-defined number of FCSs. Each of these FCSs is tested, and each of them produces a time series when tested. On the basis of this time series it is possible to estimate the potential of a FCS according to the *MOC* and the fractal dimension it produces. Solutions that indicate as being better than others are selected and modified by the GA to achieve further improvement. This process runs until a pre-defined threshold is reached. A system developer would evaluate the final proposal of the system.

Certainly, this process can be implemented at different levels of complexity. The GA could be used for the generation of a rule base only. Additionally it could be used for the generation of fuzzy sets, and the selection of different inference mechanisms. Our intention therefore is to begin with the testing of less complex systems. This strategy is also supported by the fact that very often the description of a problem and its solution could be very simple in a FCS. For example, FCS System 1 uses only five rules, eight fuzzy sets, one input variable, and one output variable.

6. Related work

Control systems, including FCSs have been studied extensively, with different interests, in the past [1, 15]. This section mentions some of the work that motivated us in our research.

Chen and Hwang for example, indicate that it is nearly always possible to describe FCS applications in completely different domains with a relatively small number (about five to eight) of very often similar fuzzy sets [2]. An early paper by Miller supports Chen and Hwang's work by identifying the number seven plus/minus two as a benchmark in many complex situations [13]. For example, instead of a lengthy explanation chess player often only mention a small number of key features of a game. These examples so far parallel the observations mentioned earlier here in terms of the simplicity and the similarity of many FCS applications. The simplicity aspect in particular can be advantageous for the system we bear in mind. For example, the discussion so far suggest the generation of a relatively small number of fuzzy sets for a FCS by the GA in Fig. 6, and this would keep the complexity of the full system low. Further relevant material can be found in a paper by Schuster [16]. Schuster discusses the relationship between self-similarity in chaos theory and so-called adaptive fuzzy sets in the context of intelligent systems. Schuster argues that a set of fuzzy sets used for the description of a system variable often can be used for the same variable at different scales, but also very often for a completely different variable. Finally, in the field of FCSs researchers nearly always emphasise the trial and error nature of the development process and the importance of the system developer. The integration of the techniques presented in this paper with a search strategy such as a genetic algorithm therefore seems to be very promising for the problem at hand.

7. Summary

This paper presented a proposal for the task of automated knowledge generation. The proposal includes ideas and concepts from chaos theory. The results we obtained from different experimental investigations are encouraging in our opinion. Although our study concentrated on a particular type of knowledge based systems, namely FCSs, we belief that the presented approach may have the potential to be useful for a wider range of problems. Our current efforts revolve around an implementation of the presented proposal in a system similar to the system described in Section 5.

Acknowledgements

We want to thank Nortel Networks and IRTU for their support in this work.

References

[1] G. A. F. Aly, M. N. Aly, A. A. Shoukry, and A. A. Daby, "Different techniques for tracking non-linear control systems.", in *CDROM 6th Int. Conf. Inform. Syst., Anal. Synth. ISAS '2000*, Orlando, USA, 2000.

[2] S. J. Chen and C. L. Hwang, *Fuzzy Multiple Attribute Decision Making, Methods and Applications*. Berlin, Heidelberg: Springer Verlag, 1992.

[3] N. Crook and C. Dobbny, "Identifying trajectories in dynamic systems", in *Proc. Int. ICSC Symp. Eng. Intel. Syst.*, Tenerife, Spain, 1998, pp. 469–475.

[4] G. Elert, "The chaos hypertextbook", http://www.hypertextbook.com/chaos

[5] J. Finlay and A. Dix, *An Introduction to Artificial Intelligence*. UCL Press, 1996.

[6] A. L. Goldberger, "Fractal mechanisms in the electrophysiology of the heart", *IEEE Eng. Med. Biol.*, pp. 47–51, 1992.

[7] N. A. J. Gough, "Fractal analysis of foetal heart rate variability", *Phys. Meas.*, vol. 14, pp. 309–315, 1993.

[8] P. Grim, "Self-reference and chaos in fuzzy logic", *IEEE Trans. Fuzzy Syst.*, vol. 1, no. 4, pp. 237–253, 1993.

[9] B. H. Kaye, *A Random Walk Through Fractal Dimensions*. Ed. Weinheim, Cambridge, New York, 1989.

[10] H. J. Korsch and H. J. Jodl, *A Program Collection for the PC*. Berlin, Heidelberg: Springer Verlag, 1994.

[11] E. N. Lorenz, "Deterministic non-periodic flow", *J. Atmos. Sc.*, vol. 20, p. 130, 1963.

[12] B. B. Mandelbrot, *The Fractal Geometry of Nature*. New York: Freeman & Company, 1977.

[13] G. A. Miller, "The magic number seven, plus or minus two", *Appl. Phys. B*, vol. 63, pp. 81–97, 1965.

[14] T. S. Parker and L. O. Chua, "Chaos: a tutorial for engineers", *Appl. Phys. B*, vol. 75, no. 8, pp. 982–1008, 1987.

[15] T. J. Ross, *Fuzzy Logic with Engineering Applications*. New York: McGraw-Hill, 1995.

[16] A. Schuster, K. Adamson, and D. A. Bell, "Problem-solving in a self-similar world and adaptive fuzzy sets", in *Proc. IASTED Int. Conf. Artif. Intel. Soft Comput.*, Honolulu, Hawaii, USA, 1999, pp. 193–196.

[17] I. Stewart, *Does God Play Dice? The New Mathematics of Chaos*. London: Penguin, 1997.

Alfons Schuster
e-mail: a.schuster@ulst.ac.uk
Faculty of Informatics
School of Information and Software Engineering
University of Ulster
Shore Road, Newtownabbey, Co. Antrim, Northern Ireland