Regularity and chaos in recurrent fuzzy systems[☆]

Jürgen Adamy*, Roland Kempf

Institut für Automatisierungstechnik, Technische Universität Darmstadt, Landgraf-Georg-Str. 4, Darmstadt 642 83, Germany

Abstract

In this paper, we shall present a mathematical definition of recurrent fuzzy systems and begin to systematically investigate the underlying theory involved. Unlike static fuzzy systems, recurrent fuzzy systems are equipped with time-delayed feedback of their output and allow representing knowledge-based dynamic processes that may be stated in the form of "if ..., then ..." rules. We study their relationship to automata and show that they have an automaton-like behavior when appropriately designed. In other cases, recurrent fuzzy system may exhibit chaotic behavior. We present sufficient conditions for the occurrence of chaos in recurrent fuzzy systems that can easily be checked solely on the basis of the qualitative, linguistically formulated models. We also discuss the extent to which state graphs may be used for describing the behaviors of recurrent fuzzy systems.

Keywords: Recurrent fuzzy systems; Automata; Recurrent neural nets; Dynamic fuzzy systems; Chaos; Linguistic dynamic

1. Introduction

In general, classical fuzzy systems describe static interdependencies of their input and output variables. In describing dynamic processes, one may call upon the methods of Takagi and Sugeno [43], where the dynamics involved are described by either difference equations or differential equations, weighted by fuzzy rules, as required by the premises on which those rules are based. These rules have the following form:

$$\text{if } \ldots, \text{ then } \mathbf{y}(k+1) = \mathbf{f}(\mathbf{y}(k), \mathbf{u}(k)). \tag{1}$$
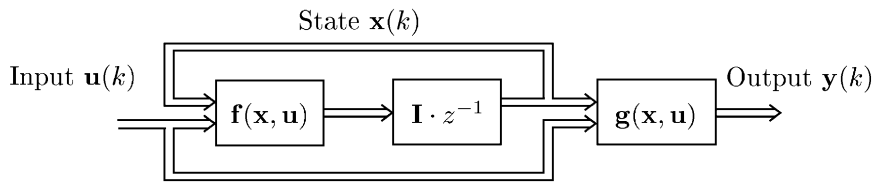
Fig. 1. Block schematic of a general, discrete-time, dynamic system, $\mathbf{I}$ denotes the unit matrix.

The fuzzy part of this hybrid expression is used to weight various dynamic systems, whose dynamics are fully specified by the equation appearing in the conclusion part of the rules involved.

However, there are ways of directly describing the dynamics of systems by means of rules. In order to provide an overview of the subject matter, we shall start off by considering a general, discrete-time, dynamic system (cf. Fig. 1) described by the function $\mathbf{f}$, the output function $\mathbf{g}$, and the equations

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)), \tag{2}$$

$$\mathbf{y}(k) = \mathbf{g}(\mathbf{x}(k), \mathbf{u}(k)). \tag{3}$$

Since the output function has no effect on the system dynamics and may not even exist in many instances, it will not be considered in detail here. This approach yields various types of dynamic systems whose characteristics will depend upon the ranges X and U for which the state variable, $\mathbf{x}(k)$, and the input variable, $\mathbf{u}(k)$, are defined and the types of functions chosen for $\mathbf{f}$ and $\mathbf{g}$.

Under the first case considered, real vector spaces, $X = \mathbb{R}^n$, $U = \mathbb{R}^m$, are chosen as their domain. Employing linear functions for $\mathbf{f}$ and $\mathbf{g}$ then yields linear, discrete-time systems. Correspondingly, employing nonlinear functions yields nonlinear, discrete-time systems. The nonlinear functions involved may be categorized by their origins and characteristics, and may be either analytically defined or available in the form of lookup tables. One example hereof is employment of neural nets for $\mathbf{f}$ and $\mathbf{g}$, which yields recurrent neural nets.

Under the second case considered, X and U are discrete, finite sets. If we employ discrete functions for $\mathbf{f}$ and $\mathbf{g}$, the resultant dynamic systems will form finite automata. The mappings of their discrete states, $\mathbf{x}$, on new states may, in all cases, be regarded as having been assembled from logical switching functions based on a dual-valued or multi-valued logic incorporating a finite number of logical values.

Under the third case considered, the defining sets involved may contain entirely general types of objects. These will usually be taken from a metric space in order to allow considering continuous mappings. The types of objects involved include sets, intervals, and functions.

Fuzzy principles may be employed in any of these three cases. Employing fuzzy numbers or fuzzy membership functions for the intervals or functions on X and U involved in the third case yields either dynamic fuzzy systems [41,28] or iterated fuzzy sets [24,12]. Extending the range of logical values allowed to the interval $[0, 1]$ in the second case yields fuzzy automata [48,30], also called fuzzy finite-state machines [21,45,35], which are similar to stochastic automata, except that their transition probabilities are replaced by transition correlations that may be interpreted as transition possibilities.

Finally, in the first case, the functions employed, which will usually be nonlinear functions, may be assembled from a complete fuzzy system consisting of fuzzification, inference, and defuzzification. Such systems form the subject of this article, and shall be referred to as "recurrent fuzzy systems". The concept of recurrent fuzzy systems was contemporaneously and independently introduced in [1,2,20]. In [1,2], they were motivated by their automaton-like characteristics and used for a monitoring system in industrial applications [1,2,4]. In [20], they were motivated by their relationship to recurrent neural nets. In both cases cited, the idea of this kind of dynamical system was presented, however, no formal mathematical definition and thorough examination of their dynamic behavior and characteristics has been done, yet.

The following section presents a formal definition of recurrent fuzzy systems. Of course, in addition to the discrete-time systems mentioned above, time-continuous recurrent fuzzy systems equipped with integrators on their feedback loops are also possible, but will not be treated here. We shall illustrate the extent to which discrete-time recurrent fuzzy systems are related to automata and under which conditions recurrent fuzzy systems behave similarly to automata, which yields both a clear picture of the dynamics of recurrent fuzzy systems and a methodology for designing them. Conversely, these considerations also demonstrate that recurrent fuzzy systems are capable of yielding qualitatively new types of behaviors, e.g. chaos, that can be described by means of nonlinear discrete-time dynamic systems. After a short introduction to chaos, we will then derive sufficient conditions that allow to detect or exclude chaotic behavior of recurrent fuzzy systems.

2. Definition of "recurrent fuzzy systems"

A recurrent fuzzy system is a discrete-time, dynamic system, as depicted in Fig. 1 and described by Eqs. (2) and (3). Its state vector, $\mathbf{x}$, input vector, $\mathbf{u}$, and output vector, $\mathbf{y}$, are real-valued vectors. The functions $\mathbf{f}$ and $\mathbf{g}$ represent complete fuzzy systems consisting of fuzzification, inference, and defuzzification.

In the following, we shall start off by considering the function $\mathbf{f}$. For every component, x_i, of the state vector, $\mathbf{x}$, we define a series of linguistic values, $L_{j_i}^{x_i}$ (cf. Fig. 2), where $j_i \in \{1, 2, \ldots\}$. In the same manner, we obtain linguistic values, $L_{q_p}^{u_p}$, for every component, u_p, of the input vector, $\mathbf{u}$.

The rule base governing $\mathbf{f}$ should contain rules having the following form only:

If

$$x_1(k) = L_{j_1}^{x_1} \text{ and } \ldots \text{ and } x_n(k) = L_{j_n}^{x_n}$$

and

$$u_1(k) = L_{q_1}^{u_1} \text{ and } \ldots \text{ and } u_m(k) = L_{q_m}^{u_m}, \tag{4}$$

then

$$x_1(k+1) = L_{w_1}^{x_1} \text{ and } \ldots \text{ and } x_n(k+1) = L_{w_n}^{x_n}.$$

This rule base should be both free of contradictions and complete, i.e., for every combination of linguistic values, $L_{q_p}^{u_p}$ and $L_{j_i}^{x_i}$, there should be a single rule determining every component of the next state value. That means that each rule defines a mapping of a certain combination of index vectors,
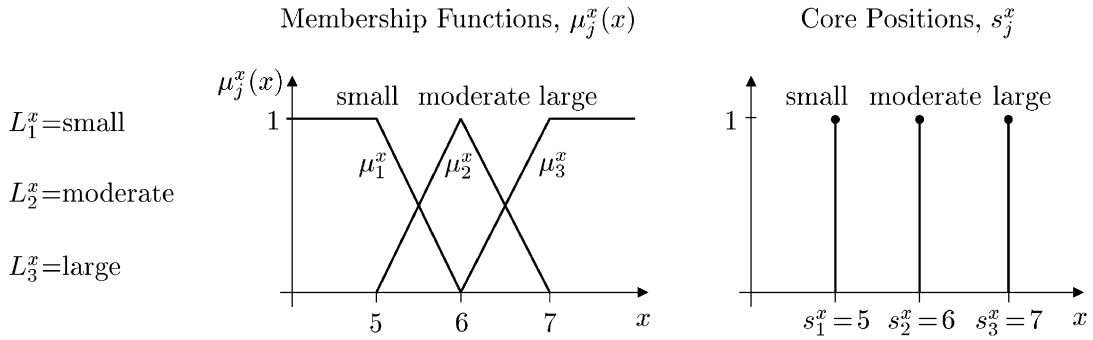
Fig. 2. Example of linguistic values, L_j^x, membership functions, $\mu_j^x(x)$, and core positions, s_j^x, for a single state variable x. As no further state variables are considered the index i in x_i, $L_{j_i}^{x_i}$, $\mu_{j_i}^{x_i}$, and $s_{j_i}^{x_i}$ is omitted.

$\mathbf{j}=(j_1,\dots,j_n)$ and $\mathbf{q}=(q_1,\dots,q_m)$, onto components, w_l, of an index vector, $\mathbf{w}=(w_1,\dots,w_n)$. In order to indicate this relationship we sometimes use $L_{w_l(\mathbf{j},\mathbf{q})}^{x_l}$ to denote the linguistic values in the conclusion of a certain rule. Note that the conclusion of the rules may only include linguistic values, $L_{w_l}^{x_l}$, that also appear in the premise of rules in the rule base.

Fuzzification involves assigning a membership function, $\mu_{j_i}^{x_i}(x_i)$, to every linguistic value, $L_{j_i}^{x_i}$. A value, $s_{j_i}^{x_i}$, termed a "core position", shall be defined for every core for which the membership function, $\mu_{j_i}^{x_i}(x_i)$, assumes its maximum value. These core positions, $s_{w_l}^{x_l}$, are chosen as singletons for the membership functions of the linguistic variables, $L_{w_l}^{x_l}$, in the conclusion of the rules. The membership functions, $\mu_{j_i}^{x_i}(x_i)$ employed for fuzzification should satisfy the following set of conditions:

- (C1) Delimitation: $\mu_{j_i}^{x_i}(x_i)\in[0,1]$ for all $x_i\in X_i\subseteq\mathbb{R}$.
- (C2) Convexity: $\begin{cases}\mu_{j_i}^{x_i}(x_i)\text{ monotonically increases for all }x_i<s_{j_i}^{x_i},\\\mu_{j_i}^{x_i}(x_i)\text{ monotonically decreases for all }x_i>s_{j_i}^{x_i}.\end{cases}$
- (C3) Partition: $\sum_{j_i}\mu_{j_i}^{x_i}(x_i)>0$ for all $x_i\in X_i$.
- (C4) Feedback correspondence: $\mu_{j_i}^{x_i}(s_{j_i}^{x_i})>0$ and $\mu_{j_i}^{x_i}(s_{l_i}^{x_i})=0$ for $j_i\neq l_i$.

The name for Condition (C4) originates from the following property: Feeding back a linguistic value, $L_{j_i}^{x_i}$, involves defuzzifying its membership function, which is a singleton at the core position, $s_{j_i}^{x_i}$. Due to Condition (C4), subsequent fuzzification of the resultant value, $s_{j_i}^{x_i}$, yields nonzero membership values for the membership function, $\mu_{j_i}^{x_i}(x_i)$, of the same linguistic value, $L_{j_i}^{x_i}$, only. The linguistic value thus is preserved under defuzzification and subsequent fuzzification.

The same conditions, (C1)–(C4), should hold for the membership function, $\mu_{q_p}^{u_p}(u_p)$, and its core position, $s_{q_p}^{u_p}$, for each linguistic value, $L_{q_p}^{u_p}$. The upper index, u_p, is needed to distinguish between the membership functions for the input variables, u_p, and the state variables, x_i. In this case, Condition (C4) does not have the meaning of a feedback correspondence, since u_p is not fed back.

Conditions (C2) and (C4) particularly single out state values, x_i, and input values, u_p, that coincide with core positions, $s_{j_i}^{x_i}$ or $s_{q_p}^{u_p}$. State vectors, $\mathbf{x}$, input vectors, $\mathbf{u}$, and combinations thereof, $(\mathbf{x},\mathbf{u})$, that have core positions, $s_{j_i}^{x_i}$ and $s_{q_p}^{u_p}$, only as entries in their components, also represent special vectors, termed "core position vectors", $\mathbf{s_j^x}\in X$, $\mathbf{s_q^u}\in U$ and $(\mathbf{s_j^x},\mathbf{s_q^u})\in X\times U$, where all of their indices, j_i and q_p, have been formed into vectors, $\mathbf{j}$ and $\mathbf{q}$. These core position vectors pervade the state space, X,
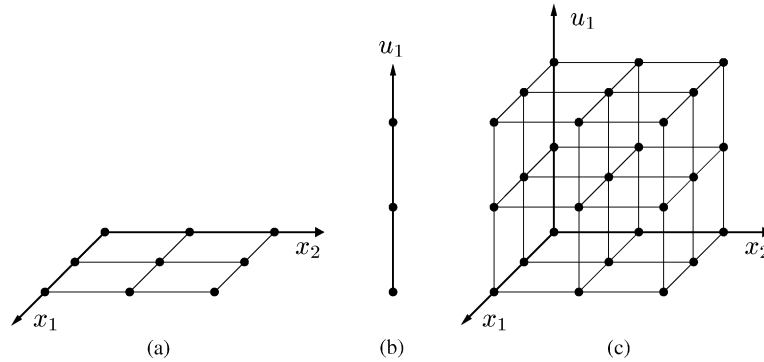
Fig. 3. The core position vectors form a lattice in the spaces (a) X, (b) U, and (c) $X \times U$.

the input space, U, or the space $X \times U$, like a lattice (cf. Fig. 3). Linguistic values may also be formed into linguistic vectors, $\mathbf{L_j^x}$, $\mathbf{L_q^u}$, and $(\mathbf{L_j^x}, \mathbf{L_q^u})$, employing procedures similar to those employed for forming core position vectors.

Algebraic multiplication is employed as both the aggregation operator and the implication operator, and summation is employed as the accumulation operator [46]. Defuzzification is based on the center of singletons defuzzification (CoS) [46]. Note that the theorems in this article also hold for the choice of the minimum operator as both the aggregation operator and the implication operator, and maximum as the accumulation operator. However, this may not be the case for the choice of arbitrary t-norms and t-co-norms or when not using CoS as defuzzification method.

If the membership functions, $\mu_{j_i}^{x_i}(x_i)$ and $\mu_{q_p}^{u_p}(u_p)$, satisfy in addition

• (C5) Continuity: $\mu_{j_i}^{x_i}(x_i) \in [0, 1]$ is continuous in $x_i \in X_i$,

the functions $\mathbf{f}$ and $\mathbf{g}$ will also be continuous with respect to the variables $\mathbf{x}$ and $\mathbf{u}$, thereby yielding continuous recurrent fuzzy systems.

At this point, all parts of $\mathbf{f}$ are defined. The function $\mathbf{g}$ is similarly configured, and, together with the foregoing, uniquely defines a recurrent fuzzy system.

In order to illustrate the definitions, we will provide an example accompanying us throughout this article. We consider a simplified model of an insect population [49,39] whose size is largely determined by the available food supply and its size during the preceding year. Larvae are presumed to hatch in the spring and develop into reproductive insects that lay eggs in the fall, provided that they have managed to survive until then, and then die. The resultant generations follow one another and have no temporal overlap, i.e., there is no intermixing of populations. The time interval involved is one calendar year.

Since the example involves a single state variable, x, and a single input variable, u, the indices i and p of x_i and u_p, respectively, can be omitted. The current population size, $x(k)$, is described by one of three allowed state values, $L_1^x = \text{small}$, $L_2^x = \text{moderate}$ or $L_3^x = \text{large}$. The food supply forms the input parameter, $u(k)$, which may have any one of three allowed linguistic values, $L_1^u = \text{small}$, $L_2^u = \text{moderate}$, or $L_3^u = \text{large}$. The evolution of the insect population may be described in terms of a few simple rules: The population will be small if the food supply is small. If the food supply is moderate or large, a small or moderate population will grow and become either moderate or large. However, large populations will decline and revert to being either moderate or small due to

Current Population Size L_j^x

		small	moderate	large
Food	small	small	small	small
Supply	moderate	moderate	moderate	small
L_q^u	large	moderate	large	moderate

Fig. 4. Rule base for a simplified model of an insect population.

intraspecific competition [7], since large populations consume the available food supply at rapid rates, which rapidly reduces the food supply. These rules may be assembled into the rule base shown in Fig. 4. The resultant new state values, L_w^x, are the rule base's output values and describe the resultant population size, $x(k+1)$, in the year $k+1$.

Obtaining numerical values for its population size, $x(k)$, requires that the states, L_j^x, be assigned numerical values such as $L_1^x = \text{small} \to 10^5$, $L_2^x = \text{moderate} \to 10^6$, and $L_3^x = \text{large} \to 10^7$, or, on a logarithmic scale, that will be used in the following, the values 5, 6, and 7, respectively. However, this approach will not allow treating any population sizes intermediate between these three values. To this end, membership functions, $\mu_1^x(x)$, $\mu_2^x(x)$, and $\mu_3^x(x)$ have to be defined like those shown in Fig. 2. In a similar manner, membership functions, $\mu_1^u(u)$, $\mu_2^u(u)$, and $\mu_3^u(u)$, have to be defined for the input variable, u. As mentioned above, the product is used both for the aggregation operator and the implication operator. The summation is employed as the accumulation operator. Defuzzification is based on the center of singletons defuzzification (CoS). The singletons for the defuzzification were chosen at positions that coincide with core positions, $s_1^x = 5$, $s_2^x = 6$, and $s_3^x = 7$ (cf. Fig. 2). As the membership functions, $\mu_j^x(x)$, satisfy conditions (C1) through (C5) all parts of this recurrent fuzzy system are given.

3. The form of the transition function

The transition function, $\mathbf{f}$, may be stated in compact, analytic, form. Once the computation of the membership values, $\mu_{j_i}^{x_i}(x_i)$ and $\mu_{q_p}^{u_p}(u_p)$, involved in fuzzification has been concluded, aggregation and implication yields the logical values for each associated rule, which will be given by

$$\prod_i \mu_{j_i}^{x_i}(x_i) \prod_p \mu_{q_p}^{u_p}(u_p). \tag{5}$$

Accumulation and defuzzification then yields the components, f_l, of $\mathbf{f}$, which will be given by

$$f_l(\mathbf{x}, \mathbf{u}) = \frac{\sum_{\mathbf{j},\mathbf{q}} s_{w_l(\mathbf{j},\mathbf{q})}^{x_l} \prod_i \mu_{j_i}^{x_i}(x_i) \prod_p \mu_{q_p}^{u_p}(u_p)}{\sum_{\mathbf{j},\mathbf{q}} \prod_i \mu_{j_i}^{x_i}(x_i) \prod_p \mu_{q_p}^{u_p}(u_p)}. \tag{6}$$

In most cases the membership functions satisfy even a stringent formulation of condition (C3), namely:
• (C3′) Normalized partition: $\sum_{j_i} \mu_{j_i}^{x_i}(x_i) = 1$ for all $x_i \in X_i$.

Then, the denominator of the right-hand side of Eq. (6), which can be rewritten as

$$\sum_{\mathbf{j,q}} \prod_i \mu_{j_i}^{x_i}(x_i) \prod_p \mu_{q_p}^{u_p}(u_p) = \prod_i \sum_{j_i} \mu_{j_i}^{x_i}(x_i) \cdot \prod_p \sum_{q_p} \mu_{q_p}^{u_p}(u_p), \tag{7}$$

will invariably have a value of 1 and thus may be simply left out. Defuzzification will then be simplified, and Eq. (6) reduces to

$$f_l(\mathbf{x,u}) = \sum_{\mathbf{j,q}} s_{w_l(\mathbf{j,q})}^{x_l} \prod_i \mu_{j_i}^{x_i}(x_i) \prod_p \mu_{q_p}^{u_p}(u_p). \tag{8}$$

Condition (C3$'$) can always be satisfied, as we can substitute any membership functions, $\mu_{j_i}^{x_i}(x_i)$, satisfying Conditions (C1)–(C4), by the membership functions

$$\tilde{\mu}_{j_i}^{x_i}(x_i) = \frac{\mu_{j_i}^{x_i}(x_i)}{\sum_{j_i} \mu_{j_i}^{x_i}(x_i)}, \tag{9}$$

satisfying not only Conditions (C1)–(C4) but also Condition (C3$'$). The transition function, $\mathbf{f}$, is uneffected by this substitution, as can be seen by comparing Eq. (8) for $\tilde{\mu}_{j_i}^{x_i}(x_i)$ and Eq. (6) for $\mu_{j_i}^{x_i}(x_i)$.

In the example of the insect population, stated above, the membership functions chosen already satisfy Condition (C3$'$), as can be checked in Fig. 2. Since the system has a single input variable, u_1, and a single state variable, x_1, only and we obtain

$$f_1(\mathbf{x,u}) = \sum_{j_1,q_1} s_{w_1(j_1,q_1)}^{x_1} \cdot \mu_{j_1}^{x_1}(x_1) \cdot \mu_{q_1}^{u_1}(u_1). \tag{10}$$

After eliminating the, in this case, unnecessary indices, Eq. (10) reduces to

$$f(x,u) = \sum_{j=1}^{3} \sum_{q=1}^{3} s_{w(j,q)}^{x} \cdot \mu_j^x(x) \cdot \mu_q^u(u). \tag{11}$$

The values of $s_{w(j,q)}^x$ may be read off the rule base given in Fig. 4 by substituting the linguistic values by their corresponding core positions.

In the case where the food supply is large and exclusively large, i.e., for the case where $\mu_1^u(u) = 0$, $\mu_2^u(u) = 0$, and $\mu_3^u(u) = 1$, we then find that f is given by

$$x(k+1) = f(x(k), u(k)) = 6 \cdot \mu_1^x(x(k)) + 7 \cdot \mu_2^x(x(k)) + 6 \cdot \mu_3^x(x(k)). \tag{12}$$

A plot of this function is presented in Fig. 5.

4. Linguistic automata and recurrent fuzzy systems

Now that we have defined what is meant by "recurrent fuzzy systems", we shall, in this section, take up the matter of their interrelationships to automata. This investigation was motivated by the fact that automata describe dynamic behavior in terms of transitions that may be represented by logical switching functions. Recurrent fuzzy systems also describe dynamic behavior, based on a transition
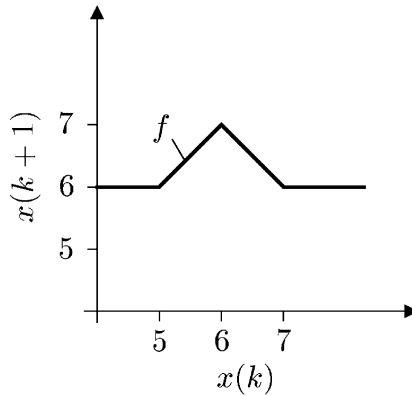
Fig. 5. Transition function for a simplified model of an insect population.

function composed of fuzzy logic functions. Their employment of fuzzy logic as an extension of Boolean logic suggests that recurrent fuzzy systems represent an extension of automata. However, the question remaining to be answered is: How are recurrent fuzzy systems related to automata and under which conditions will recurrent fuzzy systems exhibit some of the characteristics of automata?

In order to clarify this matter, we shall start off by considering the rule bases involved and their constituent rules, as defined by Eq. (4), which represent the major factors determining system dynamic behavior. Referring to the definition of linguistic vectors, we may express those rules in the following form:

$$\text{if } \mathbf{x}(k) = \mathbf{L}_j^x \text{ and } \mathbf{u}(k) = \mathbf{L}_q^u \quad \text{then } \mathbf{x}(k+1) = \mathbf{L}_w^x. \tag{13}$$

These rules map the linguistic vector, $\mathbf{L}_j^x$, of the state at time k onto a new linguistic vector, $\mathbf{L}_w^x$, at time $k+1$, where the result obtained will depend upon the linguistic input vector, $\mathbf{L}_q^u$, involved. Thus, this mapping obeys Eq. (2), as well, whereas it operates on a finite set of linguistic vectors, $\mathbf{L}_j^x$. The rules of Eq. (13), along with feedback, thus form a finite automaton, which shall be referred to here as a "linguistic automaton." For further details on automata the reader is referred to [11,15].

The behavior of such a linguistic automaton may be illustrated in the form of its state graph, which is immediately derivable from the rule base involved. The nodes of this graph are formed by the linguistic vectors, $\mathbf{L}_j^x$, of the state variables, $\mathbf{x}$, involved. The arrows indicating transitions state the conditions under which the system may undergo a transition from one state to another, expressed in terms of the linguistic vectors, $\mathbf{L}_q^u$, of the input variable, $\mathbf{u}$. A graphical representation of such a state graph, corresponding to the model of the insect population and its rule base in Fig. 4 , is depicted in Fig. 6. The wavy shape of the nodes in this state graph represents the fuzzy nature of the single state variable, x, which can take on the linguistic values small, moderate or large. The single input variable, u, can also take on the linguistic values small, moderate or large. In general, such a state graph may be generated for any linguistic automaton. Conversely, all of the linguistic values involved and the entire rule base involved may be extracted from this type of representation. Moreover, system characteristics, such as accessibility and controllability of the linguistic automaton, may also be read off the state graph.
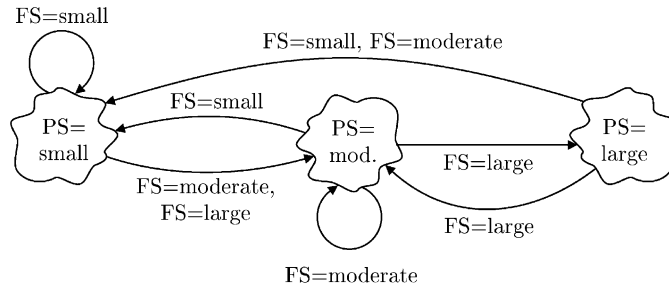
Fig. 6. State graph for a simplified model of an insect population based solely on population size, PS, and food supply, FS.

If one intends to process both linguistic values and numerical values, x_i as state variables and u_p as input variables, one must correlate them to the linguistic values involved. To this end, the numerical values to be involved are assembled into (fuzzy) sets, each of which corresponds to a linguistic value. Following further steps, one obtains a recurrent fuzzy system. If one chooses to employ discontinuous membership functions that can take on the logical values 0 or 1 only, then the systems involved will continue to form automata, since, in all cases, only a single rule at a time is employed for computing their subsequent state.

Even general recurrent fuzzy systems are still related to their linguistic automata, which are finite automata as mentioned above, as stated in the following lemma:

Lemma 1. *A recurrent fuzzy system operates on its core position vectors in the same manner as its associated linguistic automaton operates on the corresponding linguistic vectors.*

Proof. If core position vectors, $\mathbf{s}_j^x$ and $\mathbf{s}_q^u$, are chosen for $\mathbf{x}(k)$ and $\mathbf{u}(k)$, then, according to the feedback correspondence condition, Condition (C4), the premises for computing the next state vector, $\mathbf{x}(k+1)$, or output vector, $\mathbf{y}(k)$, will be satisfied for a single rule only given by Eq. (13). In this case, Eq. (6) implies that $\mathbf{x}(k+1) = \mathbf{f}(\mathbf{s}_j^x, \mathbf{s}_q^u) = \mathbf{s}_{w(j,q)}^x$ equals the singleton position vector corresponding to the linguistic vector, $\mathbf{L}_{w(j,q)}^x$, appearing in the conclusion part of that particular single rule. $\quad\square$

As a recurrent fuzzy system that is limited to operating on core position vectors is isomorphic to the associated linguistic automaton, one may describe the dynamics of a recurrent fuzzy system subject to that sort of restriction in terms of its associated linguistic automaton, and vice versa. This close interrelationship between the linguistic vectors $\mathbf{L}_j^x$ and $\mathbf{L}_q^u$ and the core position vectors $\mathbf{s}_j^x$ and $\mathbf{s}_q^u$, distinguishes core positions from other input values and state values.

However, recurrent fuzzy systems are not defined at their core position vectors only. Their associated membership functions span the full extents of their input spaces and their state spaces. In the case of continuous membership functions, there will always be ranges where neighboring membership functions overlap and both of the membership functions involved are nonzero. More than a single rule will thus be needed for computing their subsequent state, which will have to be weighted by the logical value of its associated premises. As a next step, we need to consider the extent to which state graphs may be reasonably employed, since several transition arrows will now be simultaneously active.

5. Automaton-like behavior of recurrent fuzzy systems

The necessary condition for reasonable employment of a state graph for an arbitrary initial state/input vector, $(\mathbf{x}, \mathbf{u})$ is as follows: there must exist a core position vector, $(\mathbf{s_j^x}, \mathbf{s_q^u})$, that is somehow correlated to the vector $(\mathbf{x}, \mathbf{u})$ and will exhibit approximately the same dynamic behavior as the original vector, $(\mathbf{x}, \mathbf{u})$. In this case, the associated state graph will also approximately describe the behavior of the iterated vector, $(\mathbf{x}, \mathbf{u})$. The recurrent fuzzy system involved will then behave like an automaton for the latter vector, $(\mathbf{x}, \mathbf{u})$. In order to formalize this idea, we shall define the correlation between its state/input vector, $(\mathbf{x}, \mathbf{u})$, and core position vectors, $(\mathbf{s_j^x}, \mathbf{s_q^u})$, as follows:

Definition 1 (Correlation of core position vectors and other vectors). A core position vector, $(\mathbf{s_j^x}, \mathbf{s_q^u})$, and another vector, $(\mathbf{x}, \mathbf{u})$, are said to be correlated iff there is only one, single, core position, i.e. $s_{j_i}^{x_i}$ or $s_{q_p}^{u_p}$, resp., that falls within every closed interval bounded by the values of the components of the vectors $(\mathbf{s_j^x}, \mathbf{s_q^u})$ and $(\mathbf{x}, \mathbf{u})$.

A core position vector will be correlated to all vectors falling within the largest, open, paraxial, hypersquare that contains the core position vector involved alone as its sole core position vector. Conversely, any vector will, in general, be correlated to several core position vectors. In the special case where the vector involved is a core position vector, it will be correlated to itself only. Fig. 7 presents examples of the correlations involved. "Automaton-like" may now be defined as follows:

Definition 2 (Definition of "automaton-like"). A recurrent fuzzy system is said to be automaton-like in a set, $M \subseteq X \times U$, iff for every vector, $(\mathbf{x}, \mathbf{u}) \in M$, there exists a core position vector, $(\mathbf{s_j^x}, \mathbf{s_q^u})$, correlated to it such that the core position vector, $\mathbf{f}(\mathbf{s_j^x}, \mathbf{s_q^u})$, is also correlated to $\mathbf{f}(\mathbf{x}, \mathbf{u})$.

This "automaton-like" property may be immediately transferred to the unions of such sets, M, and will be satisfied by arbitrary subsets in such sets, M.

Automaton-like recurrent fuzzy systems allow approximating the dynamics in such sets, M, by the dynamics of their core position vectors. Since their state graphs provide full descriptions of the dynamics of their core position vectors, they may also be utilized for deriving approximate descriptions of the dynamics of recurrent fuzzy systems taken from such sets, M, since starting points in the neighborhood of a certain core position vector will remain close to the image of that core position vector subsequent to mapping. Although this property will be maintained by any

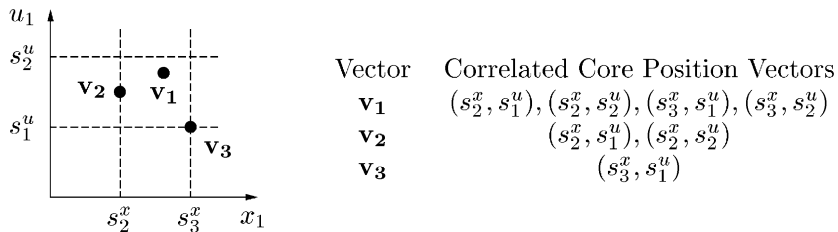| Vector	Correlated Core Position Vectors
$\mathbf{v_1}$	$(s_2^x, s_1^u), (s_2^x, s_2^u), (s_3^x, s_1^u), (s_3^x, s_2^u)$
$\mathbf{v_2}$	$(s_2^x, s_1^u), (s_2^x, s_2^u)$
$\mathbf{v_3}$	(s_3^x, s_1^u)

Fig. 7. Several typical fuzzy states and their correlated core position vectors within $X \times U$.

continuous mapping, the automaton-like property imposes limitations on the potential growth of the displacements involved. Especially, the state vector of an automaton-like recurrent fuzzy system cannot take on a core position vector if the latter does not explicitly appear in the rules needed for its computation of the state vector. Applying the above definition of "automaton-like", Lemma 1 may be restated in the form:

Lemma 2. *Every recurrent fuzzy system is automaton-like in the set of its core position vectors.*

Proof. Holds by definition. □

It would be interesting to formulate a criterion that would allow determining whether a given recurrent fuzzy system is automaton-like within a subspace, M, of its state/input space, $X \times U$, or everywhere within the space $X \times U$. Definitions 3–5, below, will be of assistance in formulating such a criterion:

Definition 3 (Definition of "adjacent"). Two core position vectors are said to be adjacent iff only one of their components differ and no further core positions fall within the range bounded by the two values of the differing component involved.

Two linguistic vectors, $\mathbf{L}_j^x$ (or $\mathbf{L}_q^u$) are said to be adjacent iff their core position vectors are adjacent.

Two rules are said to be adjacent iff they are defined as in Eq. (13) and their linguistic state/input vectors, $(\mathbf{L}_j^x, \mathbf{L}_q^u)$, are adjacent with respect to their premises.

Definition 4 (Definition of "rule continuity"). A rule base, defined as in Eq. (13), is said to be rule-continuous on a set of rules iff for any two adjacent rules contained in the latter set the linguistic vectors contained in its conclusion part are either equal or adjacent.

An entire rule base is said to be rule-continuous iff the rule base involved is rule-continuous on the set of all rules.

If a rule base is rule-continuous, then its conclusion part will not skip any linguistic values, i.e., will be, so to say, "linguistically" continuous. The conditions involved may be readily verified, based on the matrix representation of the rule base involved. Vertically or horizontally adjacent matrix elements must contain equal or adjacent linguistic values. It is easy to check that the rule base of the population dynamic of the insect population shown in Fig. 4 is rule-continuous.

Definition 5 (Definition of "elementary hypersquare"). The set of core position vectors correlated to a vector, $(\mathbf{x}, \mathbf{u})$, form the corners of a hypersquare. This hypersquare is said to be the elementary hypersquare of the vector $(\mathbf{x}, \mathbf{u})$.

Elementary hypersquares may have fewer dimensions than the space $X \times U$. For example, every core position vector is an elementary hypersquare of dimension 0. A vector may be simultaneously contained within several elementary hypersquares of different vectors, e.g., if it is located on their surfaces or their common edges or corners.

Rules given in the compact form appearing in Eq. (13) will be designated "the rules of the associated hypersquare" if their premises, $\mathbf{L}_j^x$ and $\mathbf{L}_q^u$, correspond to core position vectors that are corners of an elementary hypersquare. The rules of the elementary hypersquare of $(\mathbf{x}(k), \mathbf{u}(k))$ alone will be sufficient for computing the subsequent state, $\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k))$.

That concludes the preliminaries for the following theorem:

Theorem 1 (The similarity theorem). (a) *A recurrent fuzzy system that has a single state variable will be automaton-like in an elementary hypersquare, $M \subseteq X \times U$, if the rules applicable to that hypersquare are rule-continuous.*

(b) *A continuous recurrent fuzzy system that has a single state variable will be automaton-like in an elementary hypersquare, $M \subseteq X \times U$, if, and only if, the rules applicable to that hypersquare are rule-continuous.*

As stated in the paragraph following the definition of "automaton-like", a recurrent fuzzy system that is automaton-like within several elementary hypersquares will also be automaton-like within the union of all sets of elementary hypersquares. In particular, a recurrent fuzzy system that has a single state variable will be automaton-like throughout the full extent of the space $X \times U$ if its rule base is everywhere rule-continuous. This is the case for the example of the insect population.

Proof of the similarity theorem. Since the recurrent fuzzy system involved has a single state variable only, its core positions, s_j^x in X, will be identical to its core position vectors, $\mathbf{s}_j^x$ in X.

(a) Assume that the rules applicable to a recurrent fuzzy system having a single state variable, x, only are rule-continuous within an elementary hypersquare, M. Consider a state/input vector, $(x, \mathbf{u})$, in M and its elementary hypersquare, $H \subseteq M$. The vector involved, $(x, \mathbf{u})$, will then be correlated to each corner of that hypersquare only. The corners of that hypersquare, H, will be mapped onto a set, R, of core positions taken from X by the transition function, f. Due to the rule-continuity condition, the set R will contain all core positions contained within X falling within the range $r_{\min} = \min\{R\}$ to $r_{\max} = \max\{R\}$. To see how this follows, consider that rule of the elementary hypersquare, H, that leads to the core position $r_{\min}$ and alter the linguistic values appearing in the premises on which the rules involved are based step by step until that rule for the elementary hypersquare that leads to core position $r_{\max}$ results. Due to the rule-continuity condition, each such step will alter the core positions appearing in the conclusion part of the rules involved by a single unit at most, i.e., all core positions falling within the range $r_{\min}$ to $r_{\max}$ will have been transited.

All that remains to be done in order to prove that the recurrent fuzzy system involved is automaton-like is to show that a core position correlated to the transition vector, $f(x, u)$, falls within the interval $[r_{\min}, r_{\max}]$. Since the subsequent state, $f(x, u)$, will be either a convex combination in accordance with Eq. (8) or a generalized mean of core positions taken from R, as defined by Eq. (6), we have that $r_{\min} \leqslant f(x, \mathbf{u}) \leqslant r_{\max}$. All core positions correlated to $f(x, \mathbf{u})$ will also fall within $[r_{\min}, r_{\max}]$, since otherwise either $r_{\min}$ or $r_{\max}$ would fall within the range defined by $f(x, \mathbf{u})$ and its correlated core positions. The system involved must thus necessarily be automaton-like.

(b) In view of the conclusions reached under (a), above, it will be sufficient to prove that a continuous recurrent fuzzy system cannot be automaton-like within an elementary hypersquare, M, if the rule base is not rule-continuous on the rules of the elementary hypersquare, M.
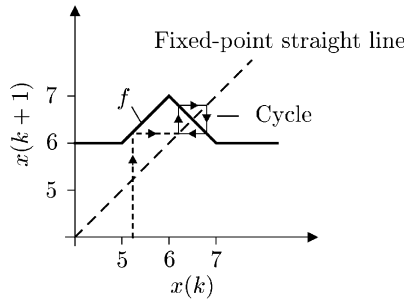
Fig. 8. Transition function for a simplified model of an insect population.

Assume that the rules applicable to a recurrent fuzzy system having a single state variable, x, are not rule-continuous within the elementary hypersquare, M. Further assume that there exist two adjacent linguistic states, L_a^x and L_b^x, that do not lead to adjacent subsequent states for a fixed linguistic input, $\mathbf{L_q^u}$. Since f is continuous in the variables x, there exists some state value, x_3, falling in between the core positions $x_1 = s_a^x$ and $x_2 = s_b^x$ such that $f(x_3, \mathbf{u})$ takes on the value of a core position falling within the range $f(x_1, \mathbf{u})$ to $f(x_2, \mathbf{u})$. However, if that were the case, then neither $(x_1, \mathbf{u})$ nor $(x_2, \mathbf{u})$, the two core positions only correlated to $(x_3, \mathbf{u})$, would be mapped onto $f(x_3, \mathbf{u})$, i.e., onto the sole core position correlated to $f(x_3, \mathbf{u})$, which would violate the necessary condition for the automaton-like property. The other case involved, that where identical linguistic values lead to non-adjacent subsequent states for adjacent linguistic inputs, may be treated in an entirely similar manner. □

The range of fields were linguistic models may be used to describe a dynamic system is wide: engineering, biology, ecology, economics and social sciences [27,10,5]. If the rule base governing a one-dimensional linguistic model is rule-continuous, the similarity theorem guarantees that the extension of that linguistic model to a recurrent fuzzy system will be automaton-like, and thus the dynamic of the resultant mathematical model can be approximately described by the linguistic model. As automaton-like recurrent fuzzy systems are capable of describing mathematical differentiation and integration linguistically [5] they are expected to appear in many linguistic models of dynamic processes.

As an example, we return to the model of the insect population. The system is automaton-like as the similarity theorem is applicable due to the rule-continuity of the systems rule base. In the case of a large food supply, the transition function in plotted in Fig. 8. The fixed-point straight line serves as a guide in determining the population's evolution, which is depicted in Fig. 9. Running its population through the recursion relation n times, where

$$x(k+n) = f^n(x(k)) = f(f(f \ldots (x(k)))), \tag{14}$$

yields cycles of length 2. In general, a numerical sequence, $a, f(a), \ldots, f^{n-1}(a), f^n(a) = a$, and $f^i(a) \neq a$ for $i = 1, \ldots, n-1$, is termed a cycle of length n. The amplitude of these cycles will depend upon the initial values involved and, in this particular case, will fall within the interval $[6,7]$. This behavior is similar to that of the associated linguistic automaton as the state graph of this automaton also exhibits a cycle of length 2 in the case of a large food supply (cf. Fig. 6).
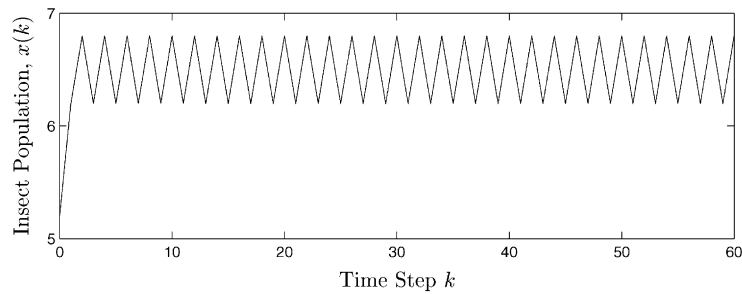
Fig. 9. Time series for the population size under the simplified model of an insect population for an initial population of $x(0) = 5.2$.

If the recurrent fuzzy system is automaton-like it has got a predictable regularity as described in the similarity theorem. However, if just one single rule fails to satisfy the rule-continuity condition, that failure may lead to a totally new type of behavior of the dynamic system involved and chaos may occur, which will now be illustrated through an example.

6. An example of a chaotic recurrent fuzzy system

We now change a single rule, as shown in Fig. 10. If the food supply is large and the population is also large, intraspecific competition will become even stronger and there will be such strong competition for food that the population will decline and be small the following year. Then the population's dynamic behavior will drastically change and chaos will occur in the case of a large food supply, as has been indicated by the jagged arrows appearing on the state graph shown in Fig. 11.

From Eq. (10), the transition function, f, for the case of a large food supply becomes

$$x(k+1) = f(x(k), u(k)) = 6 \cdot \mu_1^x(x(k)) + 7 \cdot \mu_2^x(x(k)) + 5 \cdot \mu_3^x(x(k)). \qquad (15)$$

This function is plotted in Fig. 12. The simulation for two nearly identical initial populations shown in Figs. 13 and 14 illustrates the chaotic dynamics involved. The chaos that occurs represents chaos in the sense of Li and Yorke, which is defined as follows [32,26,6]:

Definition 6 (Chaos in the sense of Li and Yorke[1]).   Given a continuous function, $f : I \rightarrow I$, where I is a compact interval, i.e. a bounded and closed interval, this function will be termed "chaotic" in the sense of Li and Yorke if there exists an uncountably infinite set, $S \subseteq I$, such that the conditions $\limsup_{n \rightarrow \infty} |f^n(x_1) - f^n(x_2)| > 0$ and $\liminf_{n \rightarrow \infty} |f^n(x_1) - f^n(x_2)| = 0$ will be satisfied for every $x_1, x_2 \in S$, where $x_1 \neq x_2$.

---

[1] Under the original definition stated by Li and Yorke [32], a further condition for chaos is that cycles having periods of every arbitrary length must exist. In the literature, however, most authors refer to chaos, in the sense of Li and Yorke, without requiring that this condition be satisfied [31,6].

Current Population Size L_j^x

		small	moderate	large
Food Supply L_q^u	small	small	small	small
	moderate	moderate	moderate	small
	large	moderate	large	small	← formerly "moderate"

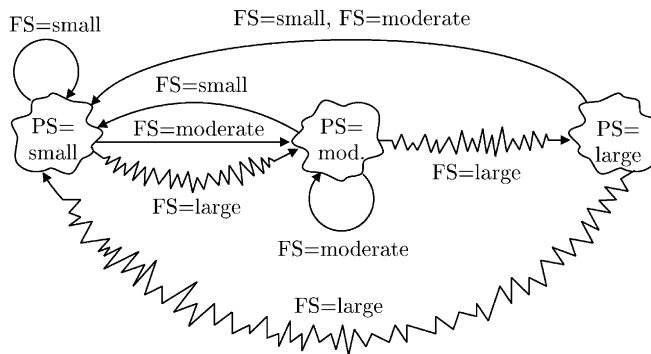Fig. 10. Rule base for a chaotic model of an insect population.

Fig. 11. State graph for a model of an insect populution that is chaotic under some circumstances and based solely on population size, PS, and food supply, FS.
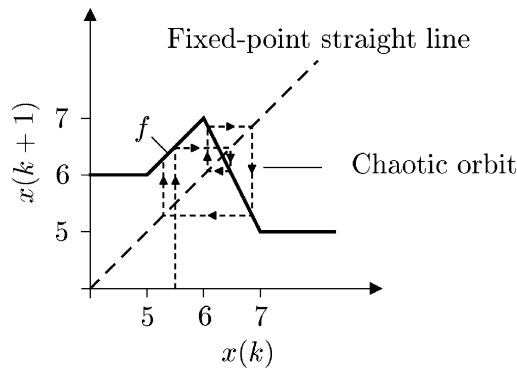

Fig. 12. Transition function for a chaotic model of an insect population.

In other words, there exist state trajectories that, with the passage of time, repeatedly approach one another arbitrarily closely and then separate, and nearly identical initial values may lead to widely differing dynamic behaviors. This property may be seen from Figs. 13 and 14. It should be noted that these two trajectories approach one another once again after 40 iterations before they,
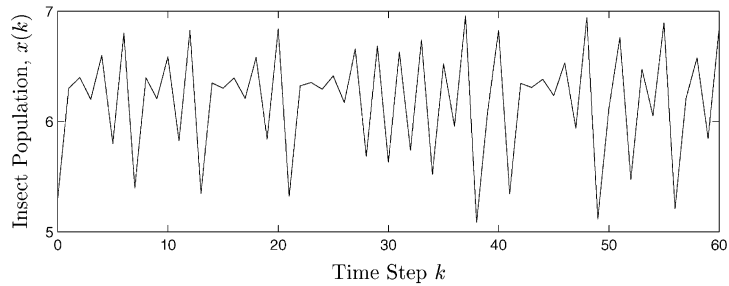
Fig. 13. Chaotic time series for the population size under the chaotic model of an insect population for an initial population of $x(0) = 5.3001$.
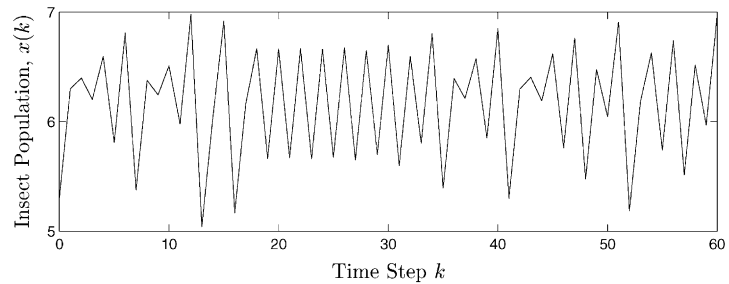

Fig. 14. Chaotic time series for the population size under the chaotic model of an insect population for an initial population of $x(0) = 5.3007$.

once again, separate. A sufficient condition for chaos, in the sense of Li and Yorke, is given by the following theorem [32]:

Theorem 2 (Li and Yorke). *If the function $f : I \rightarrow I$ is continuous on a compact interval, I, and there exists a point, $a \in I$, for which $f^3(a) \leqslant a < f(a) < f^2(a)$ (or $f^3(a) \geqslant a > f(a) > f^2(a)$), then f has a cycle of length 3 and is chaotic, in the sense of Li and Yorke.*

This theorem will now be applied to the function, f, appearing in Eq. (15), for the case of the example under consideration. Letting $a = 5$, we have $f(a) = f(5) = 6, f^2(a) = f(6) = 7$, and $f^3(a) = f(7) = 5$, as may readily be seen from Fig. 12 and verified using Eq. (15). The conditions of Theorem 2 are thus satisfied and f is chaotic, in the sense of Li and Yorke. However, the existence of a cycle of length 3, $a \rightarrow f(a) \rightarrow f^2(a) \rightarrow f^3(a) = a$, and thus compliance with Theorem 2, is already apparent from both the rule base (cf. Fig. 10) and the state graph (cf. Fig. 11) involved.

This example allows us to hope that criteria for the existence of chaotic behavior that may be analyzed based on the rule bases or state graphs involved will be derivable. However, before we proceed to derive those criteria, let us first cover those aspects of chaos theory that will be required.

7. Chaos in dynamic systems

The definition of chaos presented in the preceding section is but one of many that may be applied to discrete-time, dynamic systems. Another definition that is commonly employed is that given by Devaney [13].

Definition 7 (Chaos in the sense of Devaney). A continuous function, $f : I \to I$, is said to be chaotic on an invariant set, $M \subseteq I$, if its restriction $f|_M$ to the set M satisfies the following conditions:

1. It is topologically transitive, i.e., if there exists a $k > 0$ such that $f^k(U) \cap V \neq \{\}$, where $f^k(U) = \{f^k(x) | x \in U\}$, for any pair of open sets, $U, V \subseteq M$.
2. Its periodic points are dense in M.
3. It is sensitive to the initial conditions, i.e., if there exists a $\delta > 0$ such that for any $x \in M$ and any neighborhood N of x there exists a $y \in N$ and an $n > 0$ such that $|f^n(x) - f^n(y)| > \delta$.

However, the third condition is redundant in the nontrivial case where M is infinite, since it follows from Conditions 1 and 2 [26,6]. If the set I is a compact interval, then chaos in the sense of Devaney implies chaos in the sense of Li and Yorke [6] and the two definitions of chaos are reduced to being virtually identical. A finer distinction between the two definitions of chaos and deeper insights into chaotic behavior can be gained by a theorem that was first stated by Sarkovskii [40,13]. It addresses the existence of cycles having certain period lengths. Consider the following ordering of the natural numbers:

$$3 \triangleright 5 \triangleright 7 \triangleright \cdots \triangleright 2 \cdot 3 \triangleright 2 \cdot 5 \triangleright \cdots \triangleright 2^2 \cdot 3 \triangleright 2^2 \cdot 5 \triangleright \cdots \triangleright 2^3 \cdot 3 \triangleright 2^3 \cdot 5 \triangleright \cdots \triangleright 2^3 \triangleright 2^2 \triangleright 2 \triangleright 1.$$

Under this ordering, the sequence of all odd integers ranging from 3 to ∞ is followed by another sequence whose terms consist of that same sequence multiplied, in turn, by 2, 2^2, 2^3, etc. Appearing at the end of that sequence is yet another sequence consisting of all integral powers of 2, arranged in descending order and terminating in $2^0 = 1$. The theorem by Sarkovskii states, that a function that has a cycle of length k will also have a cycle of length l, provided that $k \triangleright l$. This theorem allows categorizing functions, f, in terms of their maximum period lengths using this particular ordering, where three major cases may be distinguished:

(1) Functions, f, that have a cycle of length n, where n is not an integral power of 2, and are thus, according to [6], chaotic, both in the sense of Devaney and in the sense of Li and Yorke [26,6].

(2) Functions, f, all of whose cycles have period lengths that are both integral powers of 2 and less than some maximum, in which case the dynamic system will converge to a limiting cycle for every initial value and not be chaotic anywhere [22].

(3) Functions, f, between these two cases, i.e., all of whose cycles have period lengths for all integral powers of 2. Such functions are also termed "functions of type 2^∞." Although some of these functions are chaotic in the sense of Li and Yorke, none of them are chaotic in the sense of Devaney. Readers are referred to the literature [42,33,22,6] for further information on such functions.

In the following, we shall regard only those functions that are chaotic in the sense of Devaney as chaotic. This class of one-dimensional chaotic functions has the convenient property that it forms an open subset in the set of continuous functions [8,6], which is of importance when considering

functions that are continuously dependent upon one or more parameters. If one such function is chaotic for certain values, u_p, of their parameters, then all such functions whose parameters have values sufficiently close to u_p will be chaotic.

Several other sufficient conditions for chaos may be derived from Theorem 2. For example, as mentioned earlier, the following theorem appears in [26,6]:

Theorem 3. *If $f : I \to I$ is a continuous function on a compact interval, I, and has a cycle of period length n, where n is not an integral power of 2, i.e., if $n \neq 2^k$, $k \in \mathbb{N}_0$, then f is chaotic.*

In addition Theorem 2 may be extended to read as follows:

Theorem 4. *If $f : I \to I$ is a continuous function on a compact interval, I, and there exists a point, $a \in I$, for which $f^n(a) \leqslant a < f(a) < f^2(a)$ (or $f^n(a) \geqslant a > f(a) > f^2(a)$) for any $n \geqslant 3$, then f is chaotic.*

Proof. According to Theorem 9 appearing in the second chapter of [8], it follows from the condition that $f^n(a) \leqslant a < f(a)$ that either f has a cycle whose period length is an odd integer, in which case, f will be chaotic according to Theorem 3, or that $f^k(a) < f^j(a)$ for all even integers, k, and all odd integers, j, for which $0 \leqslant j \leqslant n$ and $0 \leqslant k \leqslant n$. However, the second case is excluded in view of the fact that $f(a) < f^2(a)$. $\square$

Finally, chaos criteria that are not confined to considering single orbits may also be derived with the aid of Theorem 2, as the following theorem shows:

Theorem 5. *If $f : I \to I$ is a continuous function on a compact interval, I, and there exist points, $a, b \in I$, for which $f(b) \leqslant a < b < f(a) \leqslant f^2(a)$ (or $f(b) \geqslant a > b > f(a) \geqslant f^2(a)$), then f is chaotic.*

Proof. (1) Assume that $f(b) \leqslant a < b < f(a) \leqslant f^2(a)$. The other case may be treated in similar fashion.

(2) The value of the function, f, at the point b is $f(b) < b$ and its value at the point $x = f(a) > b$ is $f(x) = f^2(a) \geqslant f(a) = x$, i.e., $f(x) \geqslant x$. There thus exists a fixed point, $p_1 \in (b, x] = (b, f(a)]$.

(3) From (2) and (1), we have that $f(a) \geqslant p_1 > b$ and $b > f(b)$, i.e., $f(a) \geqslant p_1 > f(b)$. According to the mean-value theorem, there thus exists a point, p_2, on the interval $[a, b]$ for which $f(p_2) = p_1$.

(4) From (1), (3), and (2), we have that $f(b) \leqslant a$, $a \leqslant p_2 < b$, and $b < p_1 = f(p_1)$, i.e., that $f(b) \leqslant p_2 < f(p_1)$. According to the mean-value theorem, there thus exists a point, p_3, on the interval $[b, p_1)$ for which $f(p_3) = p_2$.

(5) Since $p_2 \in [a, b)$ and $p_3 \in [b, p_1)$, it follows that $p_2 < p_3$. Applying the result, $f(p_3) = p_2$, obtained under (4), we then have that $f(p_3) < p_3$.

(6) From (5) and (4), we have that $f(p_3) < p_3$ and $p_3 < p_1 = f(p_1)$, i.e., $f(p_3) < p_3 < f(p_1)$. Applying the mean-value theorem once again, we find that there exists a point, p_4, on the interval (p_3, p_1) for which $f(p_4) = p_3$.

(7) Applying (6) and (5), we obtain the chain inequality $p_1 > p_4 > p_3 > p_2$. From the relations $p_1 = f(p_2)$, $p_2 = f(p_3)$, and $p_3 = f(p_4)$, we obtain the further relations $p_1 = f^3(p_4)$, $p_2 = f^2(p_4)$,

and $p_3 = f(p_4)$. Making these substitutions in the above chain inequality, we obtain $f^3(p_4) > p_4 > f(p_4) > f^2(p_4)$. The conditions of Theorem 2 are thus satisfied and f is chaotic. $\square$

In the section that follows, we shall apply the above theorems to recurrent fuzzy systems, which will yield further theorems that will allow concluding whether chaos occurs in recurrent fuzzy systems, based on their state graphs or rule bases alone.

8. Chaos in recurrent fuzzy systems

As represented in the preceding section, chaos in one-dimensional dynamic systems is a consequence of the existence of certain cycles and certain orbits. The aim of this section is applying those results to recurrent fuzzy systems in a manner that will allow determining whether recurrent fuzzy systems are chaotic based on their state graphs or rule bases alone. However, there are several terms that need to be defined before we will be in a position to define useful criteria for the occurrence of chaos in recurrent fuzzy systems:

Definition 8 ("linguistic orbits and core position orbits"). For a given fixed linguistic input vector, $\mathbf{L_q^u}$, a linguistic automaton generates a sequence of linguistic state values, $L(0), L(1), \ldots$, by mapping the linguistic state values, $L(k)$, onto $L(k+1)$ for $k = 0, 1, 2, \ldots$. This sequence of linguistic state values is termed a linguistic orbit.

For a given fixed input vector at a core position vector, $\mathbf{s_q^u}$, a recurrent fuzzy system generates a sequence of states at core positions, $s(0)$, $s(1), \ldots$ by $s(k+1) = f(s(k), \mathbf{s_q^u})$ for $k = 0, 1, 2, \ldots$. This sequence of states at core positions is termed a core position orbit.

Definition 9 ("linguistic cycles and core position cycles"). A linguistic orbit, $L(0), L(1), \ldots$, whose elements satisfy $L(k) \neq L(0)$ for $k = 1, 2, \ldots, n-1$ and $L(n) = L(0)$, is termed a linguistic cycle of length n.

A core position orbit $s(0), s(1), \ldots$, whose elements satisfy $s(k) \neq s(0)$ for $k = 1, 2, \ldots, n-1$ and $s(n) = s(0)$, is termed a core position cycle of length n.

Definition 10 ("neighborhood of a core position orbit"). Given is a core position orbit, $s(0), s(1), \ldots$, for an input vector at a core position vector $\mathbf{s_q^u}$. An open set, $N \subseteq X \times U$, in the product space of the state space X and the input space U, which contains all vectors, $(s(k), \mathbf{s_q^u})$ for $k = 0, 1, 2, \ldots$, is termed a neighborhood of the core position orbit involved.

Definition 11 ("chaos in recurrent fuzzy systems"). Given is a recurrent fuzzy system and a set, $N \in X \times U$, such that for all input vectors, $\mathbf{u}$, in the case of all nonempty sets, $\tilde{X}(\mathbf{u}) = \{x \in X \mid (x, \mathbf{u}) \in N\}$, the transition function, $f(x, \mathbf{u})$, of this recurrent fuzzy system is chaotic on a subset of $\tilde{X}(\mathbf{u})$. This recurrent fuzzy system is termed chaotic in this set, N.

Lemma 1 states that linguistic automata are isomorphic to recurrent fuzzy systems, confined to the core positions for their state values and the core position vectors for their input vectors. Therefore,

each linguistic orbit $L(0)$, $L(1)$, ... for a linguistic input value, $\mathbf{L_q^u}$, corresponds to, i.e. is isomorphic to, a core position orbit $s(0), s(1), ...$, for the input vector at the core position vector, $\mathbf{s_q^u}$.

In addition, the ordering relation for the core positions, $s_1^x < s_2^x < s_3^x < \cdots$, also generates an ordering relation for the corresponding linguistic values, $L_1^x < L_2^x < L_3^x < \cdots$. Theorems 2–5 may thus be immediately applied to linguistic values appearing in the rule base of recurrent fuzzy systems. From Theorems 3 and 4, we then obtain the following theorems, Theorems 6 and 7, applying to linguistic cycles:

Theorem 6. *If a one-dimensional, continuous, recurrent fuzzy system has a linguistic cycle of length $n \neq 2^k$, where $k = 0, 1, ...$, then that recurrent fuzzy system will be chaotic in a neighborhood of the corresponding core position cycle.*

Proof. If the system's input vector, $\mathbf{u}$, coincides with the core position vector, $\mathbf{s_q^u}$, of its linguistic input vector, $\mathbf{L_q^u}$, then this linguistic cycle of length n is isomorphic to the corresponding core position cycle of the same length. In this case the transition function $f(x, \mathbf{u})$ will thus be chaotic according to Theorem 3. Since the set of chaotic functions is open on the set of continuous functions and the functions $f(x, \mathbf{u})$ are continuous in the parameters u_p, there exists a neighborhood around the vector $\mathbf{u} = \mathbf{s_q^u}$ such that these $f(x, \mathbf{u})$ will also be chaotic for all $\mathbf{u}$ falling within that neighborhood. □

Theorem 7. *If a one-dimensional, continuous, recurrent fuzzy system has a linguistic cycle, $L(0)$, $L(1), L(2), ...$, whose linguistic values satisfy the chain inequality $L(0) < L(1) < L(2)$ (or $L(0) > L(1) > L(2)$), then that recurrent fuzzy system will be chaotic in a neighborhood of the corresponding core position cycle.*

The proof of Theorem 7 proceeds similarly to that for Theorem 6, where, in this case, Theorem 4, instead of Theorem 3, is used.

In the case of our example of an insect population, the occurrence of chaos may be proven using either of these theorems, provided that the food supply is large. The linguistic cycle represented by the jagged line appearing in Fig. 11 has both a length that is not an integral power of 2 and three monotonically increasing linguistic state values. As mentioned earlier, the jagged line symbolizes the occurrence of chaos. The evolution of the insect population will thus proceed chaotically, even if the input value for the food supply differs from the input at the core position of "large," provided that it is sufficiently close to that value.

If we consider linguistic orbits, Theorems 4 and 5 yield the following theorems, Theorems 8 and 9, whose proofs proceed similarly to that for Theorem 6:

Theorem 8. *If a one-dimensional, continuous, recurrent fuzzy system has a linguistic orbit, $L(0)$, $L(1), L(2), ..., L(n), ...$, whose linguistic values satisfy the chain inequality $L(n) \leqslant L(0) < L(1) < L(2)$ (or $L(n) \geqslant L(0) > L(1) > L(2)$), then that recurrent fuzzy system will be chaotic in a neighborhood of the corresponding core position orbit.*

Theorem 9. *If a one-dimensional, continuous, recurrent fuzzy system has two linguistic orbits, a and b, for the same linguistic input vector $\mathbf{L_q^u}$, whose respective linguistic values, $L_a(0), L_a(1), L_a(2)$, and*
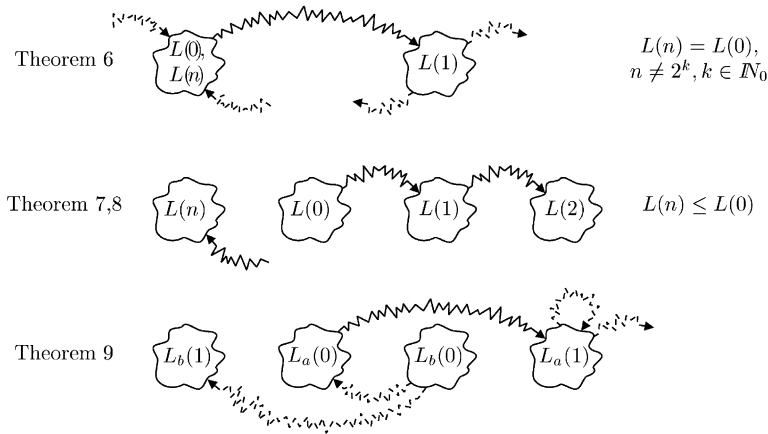
Fig. 15. The chaos criteria at a glance. The jagged dashed arrows indicate alternative paths. Further criteria may be obtained by reflecting the plots through the horizontal plane.

$L_b(0)$, $L_b(1)$, satisfy the chain inequality $L_b(1) \leqslant L_a(0) < L_b(0) < L_a(1) \leqslant L_a(2)$ (or $L_b(1) \geqslant L_a(0) > L_b(0) > L_a(1) \geqslant L_a(2)$), then that recurrent fuzzy system will be chaotic in the union of neighborhoods of the two corresponding core position orbits.

We now have a number of sufficient conditions for the occurrence of chaos in one-dimensional, recurrent fuzzy systems. Those conditions may be readily graphically analyzed using their state graphs if their linguistic state values are arranged in ascending or descending order. If certain sections of their state graphs exhibit patterns like those depicted in Fig. 15, then they will be chaotic.

As has been shown, a recurrent fuzzy system that is chaotic for all input vectors, $\mathbf{u}$, that coincide with core position vectors, $\mathbf{s_q^u}$, will also be chaotic on neighborhoods of those core positions. However, how large those neighborhoods will be is not readily determinable, as will become evident from the example shown in Fig. 16. x The figure shows the membership functions and core positions (cf. Fig. 16a) of the state variable x and some transition functions f for certain input values (cf. Fig. 16b–d). According to Theorem 9, the recurrent fuzzy system involved will be chaotic for input values for which $\mu_1^u(u) = 1$ (cf. Fig. 16b) and $\mu_2^u(u) = 1$ (cf. Fig. 16d). However, if the latter rules are only halfway satisfied, i.e., if $\mu_1^u(u) = 0.5$ and $\mu_2^u(u) = 0.5$, then the associated transition function, $f(x, \mathbf{u})$, will be constant (cf. Fig. 16c) and therefore nonchaotic.

On the other hand, a recurrent fuzzy system that is nonchaotic for input vectors that coincide with core position vectors may, nevertheless, be chaotic for other input values, as may be seen from Fig. 17. The figure shows the membership functions and core positions (cf. Fig. 17a) of the state variable x and some transition functions f for certain input values (cf. Fig. 17b–d). Both cases in which the input values coincide with core positions, i.e., the cases where $\mu_1^u(u) = 1$ (cf. Fig. 17b) and $\mu_2^u(u) = 1$ (cf. Fig. 17d), respectively, yield monotonic transition functions. As will subsequently be shown, chaos is thus precluded for these cases. For intermediate input values, i.e., for the case where $\mu_1^u(u) = 0.75$ and $\mu_2^u(u) = 0.25$, the transition function will exhibit a cycle of length 3 (cf. Fig. 17c) and, according to Theorem 2, will be chaotic. This example shows that the above theorems state sufficient conditions only.
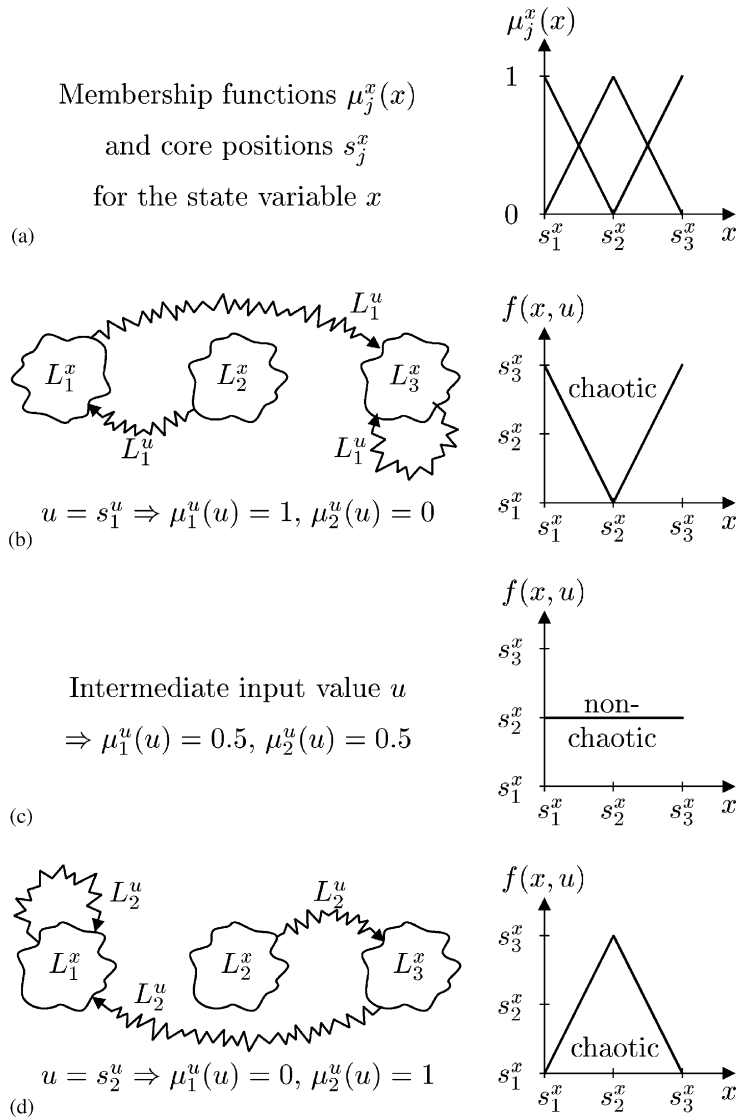
Fig. 16. The recurrent fuzzy system with membership functions and core positions for the state variable x shown in (a) and an appropriate rule base illustrates that although input values u at core positions lead to chaotic transition functions $f(x,u)$ (cf. (b),(d)), the transition function need not be chaotic for all intermediate input values (cf. (c)).

It will thus be helpful to also have available statements that preclude chaos for all input values. Such a statement can be given in the case of one-dimensional, monotonic, recurrent fuzzy systems. They are characterized by the fact that, for every linguistic input vector, a pair of linguistic state values, $L_a(0) < L_b(0)$, will be mapped onto the linguistic state values $L_a(1)$ and $L_b(1)$, which, in the case of monotonically increasing fuzzy systems, will satisfy the condition $L_a(1) \leqslant L_b(1)$, or, in the case of monotonically decreasing fuzzy systems, the condition $L_a(1) \geqslant L_b(1)$. We then have that:
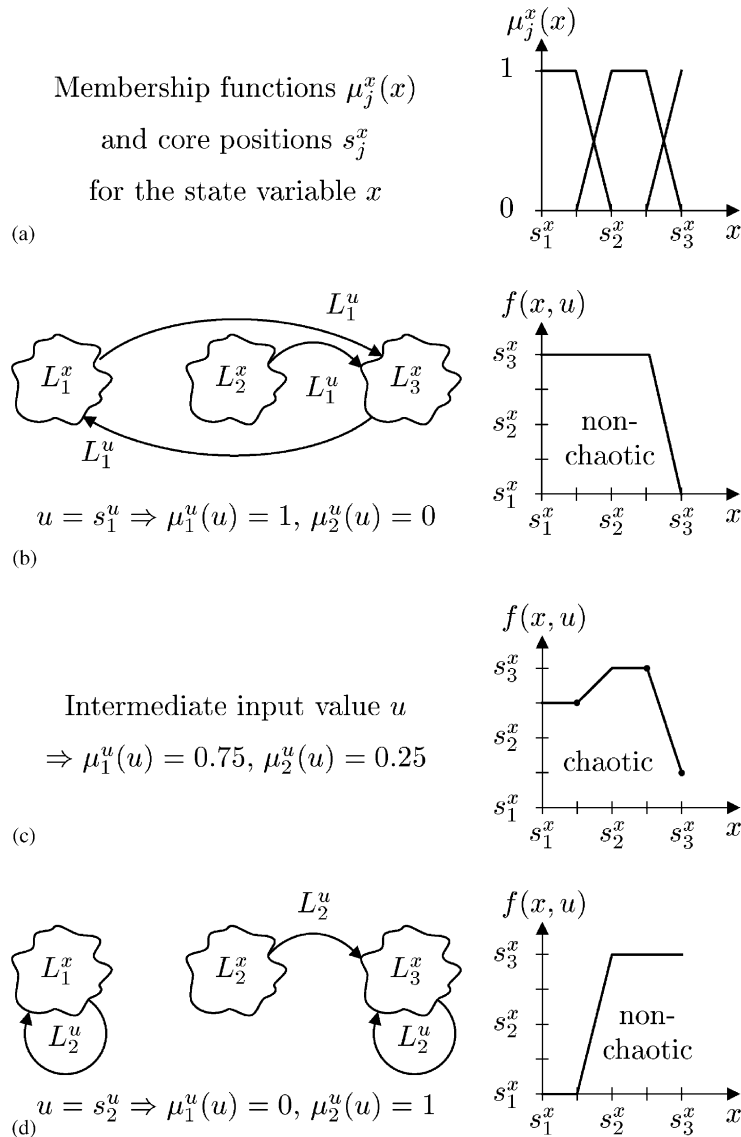
Fig. 17. The recurrent fuzzy system with membership functions and core positions for the state variable x shown in (a) and an appropriate rule base illustrates that although input values u at core positions yield nonchaotic functions $f(x, u)$ (cf. (b),(d)), chaos may occur for intermediate input values (cf. (c)).

Theorem 10. *Every one-dimensional, monotonically increasing or decreasing, recurrent fuzzy system is nonchaotic for any input value.*

Proof. According to Eq. (8), the transition function, f, will be monotonic for any input value. Monotonically increasing transition functions $f(x)$ have no cycles of period length 2, as the two points, $x_1 = f(x_2)$ and $x_2 = f(x_1)$, of such a cycle satisfying $x_1 < x_2$ would also have to satisfy the

contradicting condition $x_2 = f(x_1) \leqslant f(x_2) = x_1$ in the case of a monotonically increasing transition function $f(x)$.

Monotonically decreasing transition functions, $f(x)$, will yield monotonically increasing functions, $f^2(x)$, that have no cycles of period length 2. In this case it follows that $f(x)$ has no cycles of length 4, as this would imply that $f^2(x)$ would have cycles of length 2.

In both cases the maximum length n of a cycle in the ordering given by Sarkovskii is an integral power of 2 and the transition function, f, is nonchaotic for any input value, as discussed in Section 7. □

All of the above criteria will be of value in devising and analyzing recurrent fuzzy systems. Employment of chaotic recurrent fuzzy systems would appear to be a highly promising prospect, since systems with chaotic dynamics turn up in, among other fields, engineering, biology, ecology, epidemiology, economics, the social sciences, and physics [34,9,23]. Concrete examples include a microelectromechanical oscillator [47], chaos in digital communications systems [25], chaos in economic cycles [36,17], chaotic marketing models [16] and population dynamics [37]. Chaotic systems are also increasingly becoming the object of research in systems theory and control theory [44,26,19,18,38]. The tools commonly employed for their description are "classical" difference and differential equations. Supplementing these tools by making use of recurrent fuzzy systems would be a sensible approach. Their facilities for modeling employing verbal forms of description have numerous applications, particularly in those fields mentioned above.

Recurrent fuzzy systems are not the only possibility to use fuzzy principles in order to model chaotic dynamical processes. As mentioned in the introduction, there are also dynamical fuzzy systems [27–29,41] as well as iterated fuzzy systems [12,14]. They differ from recurrent fuzzy systems by the fact that they feed back not a defuzzified value, but a whole fuzzy set. Depending on their setup they may be parameterized and regarded as multidimensional, but finite-dimensional, mappings or have to be treated as maps in a fuzzy metric space. Due to the dimension of the domains of the functions involved in such systems, most of the results presented here are not directly applicable. Nevertheless, sufficient conditions for chaos in such systems are known [24,14].

9. Summary and conclusions

Recurrent fuzzy systems allow deriving mathematical descriptions of dynamic processes from their linguistic descriptions. It is shown in this article that recurrent fuzzy systems behave similarly to automata in many cases. There are also recurrent fuzzy systems that exhibit chaotic behaviors. The present article states criteria for the occurrence of chaos in recurrent fuzzy systems that both allow detecting chaos at the quantitative level by solely analyzing the qualitative, linguistically formulated models and devising recurrent fuzzy systems that will exhibit chaotic behavior. Since linguistically described systems play a role in many fields, such as ecology, economics, and engineering, recurrent fuzzy systems represent a useful tool for modeling such systems.

References

[1] J. Adamy, Breakout prediction for continuous casting by fuzzy mealy automata, Proceedings of EUFIT'95, Aachen, 1995, pp. 754–759.

[2] J. Adamy, Device for early detection of runout in continuous casting, Publication no. 4442087, German Patent application P4442087.0, 1994, European Patent EP 0 819 033 B1, 1998, US Patent 5,904,202, 1999.

[3] J. Adamy, J. Freitag, S. Lorenz, Process for parametering a fuzzy automaton that compares a measurement system to a pattern signal, European Patent EP0941504B1, 2001, US Patent US 6,345,206 B1, 2002.

[4] J. Adamy, W. Tautz, M. Niemann, H.-J. Nitsche, J. Göpel, Automatisierungsverfahren mit Fuzzy-Systemen und neuronalen Netzen für Stranggießanlagen und Stahlbehandlungslinien, in: GMA-Conference on 'Computational Intelligence', Berlin, VDI-Report 1381, 1998, pp. 69–86.

[5] J. Adamy, A. Yousif, Controller with dynamic discrete-time fuzzy-systems, German Patent DE19734711C1, 1999, European Patent EP 1002260 B1, 2002.

[6] B. Aulbach, B. Kieninger, On three definitions of chaos, Nonlinear Dynamics Systems Theory, Vol. 1, 2001, pp. 23–37.

[7] M.E. Begon, J.L. Harper, C.R. Townsend, Ecology: Individuals, Populations and Communities, Blackwell Science, Oxford, 1996.

[8] L.S. Block, W.A. Coppel, Dynamics in One Dimension, in: Springer Lecture Notes, Vol. 1513, Springer, Berlin, 1992.

[9] R. Blümel, W.P. Reinhardt, Chaos in Atomic Physics, Cambridge University Press, Cambridge, UK, 1997.

[10] W. Bock, A. Salski, A fuzzy knowledge-based model of population dynamics of the yellow-necked mouse (Apodemus flavicollis) in a beach forest, Ecol. Modelling 108 (1998) 155–161.

[11] T. Booth, Sequential Machines and Automata Theory, Wiley, New York, 1967.

[12] C.A. Cabrelli, et al., Iterated fuzzy sets systems: a new approach to the inverse problem for fractals and other sets, J. Math. Anal. Appl. 171 (1) (1992) 79–100.

[13] R.L. Devaney, An Introduction to Chaotic Dynamical Systems, Addison-Wesley, Redwood City, CA, 1989.

[14] P. Diamond, Chaos in iterated fuzzy systems, J. Math. Anal. Appl. 184 (1994) 472–484.

[15] H. Ehrig, Universal Theory of Automata, Teubner, Stuttgart, 1974.

[16] G. Feichtinger, C.M. Hommes, A. Milik, Complex dynamics in a threshold advertising model, OR-Spektrum 16 (1994) 101–111.

[17] G. Feichtinger, M. Kopel, Nichtlineare dynamische Systeme und Chaos: Neue Impulse für die Betriebswirtschaftslehre?, Z. Betrieb. 64 (1994) 7–34.

[18] A.L. Fradkov, A.Y. Pogromsky, Introduction to Control of Oscillations and Chaos, World Scientific, Singapore, 1998.

[19] P.M. Frank, Komplexe Systeme—Nichtlineare Rückkopplungssysteme jenseits der Stabilität, Automatisierungstechnik 46 (1998) 166–179.

[20] V. Gorrini, H. Bersini, Recurrent fuzzy systems, Proc. 3rd IEEE Conf. on Fuzzy Systems, 1994, pp. 193–198.

[21] J. Grantner, M. Patyra, Synthesis and analysis of fuzzy logic finite state machine models, in: Proc. IEEE Int. Conf. on Fuzzy Systems, Orlando, 1994, pp. 205–210.

[22] R. Kempf, On Ω-limit sets of discrete-time dynamical systems, J. Difference Equations Appl., 8 (12) (2002) 1121–1131.

[23] L.D. Kiel, Chaos Theory in the Social Sciences: Foundations and Applications, University of Michigan Press, Ann Arbor, 1997.

[24] P.E. Kloeden, Chaotic iterations of fuzzy sets, Fuzzy Sets and Systems 42 (1991) 37–42.

[25] G. Kolumban, M.P. Kennedy, L.O. Chua, The role of synchronization in digital communications using chaos—Part II, Chaotic modulation and chaotic synchronization, IEEE Trans. Circuits Systems 45 (11) (1998) 1129–1140.

[26] W. Krabs, Dynamische Systeme: Steuerbarkeit und Chaotisches Verhalten, Teubner, Stuttgart, 1998.

[27] V. Krebs, E. Schäfers, Dynamische Fuzzy Systeme zur qualitativen Prozeßmodellierung, in: GMA-Conference on 'Computational Intelligence', Berlin, VDI-Report 1381, 1998, pp. 115–135.

[28] M. Kurano, M. Yasuda, J. Nakagami, Y. Yoshida, A limit theorem in some dynamic fuzzy systems, Fuzzy Sets and Systems 51 (1992) 83–88.

[29] M. Kurano, M. Yasuda, J. Nakagami, Y. Yoshida, A fuzzy relational equation in dynamic fuzzy systems, Fuzzy Sets and Systems 101 (1999) 91–99.

[30] H.-S. Lee, Minimizing fuzzy finite automata, The Ninth IEEE International Conference on Fuzzy Systems (2000) 65–70.

[31] S. Li, ω-chaos and topological entropy, Trans. Amer. Math. Soc. 339 (1993) 243–249.

[32] T.-J. Li, J.A. Yorke, Period three implies chaos, AMM 82 (1975) 985–992.

[33] V.J. Lopez, L. Snoha, There are no piecewise linear maps of type 2^∞, Trans. Amer. Math. Soc. 349 (4) (1997) 1377–1387.

[34] R.M. May, Simple mathematical models with very complicated dynamics, Nature 261 (10) (1976) 459–467.

[35] C. Omlin, K. Thornber, C. Gilies, Representation of fuzzy finite-state automata in continuous recurrent neural networks, in: Proc. IEEE Int. Conf. on Fuzzy Systems, 1996, pp. 1023–1027.

[36] T. Puu, A chaotic process with slow feed back: the case of business cycles, in: G. Haag (Ed.), Economic Evolution and Demographic Change, Lecture Note in Economics and Mathematical Systems, Vol. 395, Springer, London, 1992.

[37] R. Reiner, M. Munz, G. Haag, W. Weidlich, Chaotic evolution of migratory systems, Sistemi Urbani 2/3 (1986) 285–308.

[38] H. Richter, Chaotisches Verhalten dynamischer Systeme und seine regelungstechnische Behandlung, Automatisierungstechnik 48 (10) (2000) 471–477.

[39] J. Roughgarden, Primer of Ecological Theory, Prentice-Hall, New Jersey, 1998.

[40] A.N. Sarkovskii, Coexistence of cycles of a continuous mapping of the line into itself, Ukrain. Mat. Z. 16 (1964) 61–71 (in Russian).

[41] E. Schäfers, V. Krebs, Dynamic fuzzy systems for qualitative process modeling: principles of a new system theory, Automatisierungstechnik 8 (99) (1999) 382–389.

[42] J. Smital, Chaotic functions with zero topological entropy, Trans. Amer. Math. Soc. 297 (1986) 269–282.

[43] H. Takagi, M. Sugeno, Fuzzy identification of systems and its application to modeling and control, IEEE Trans. Systems, Man Cybernet. 15 (1985) 16–132.

[44] J.M.T. Thompson, S.R. Bishop, Nonlinearity and Chaos in Engineering Dynamics, Wiley, New York, 1994.

[45] F. Unal, E. Khan, A fuzzy finite state machine implementation based on a neural fuzzy system, Proc. IEEE Int. Conf. on Fuzzy Systems, Orlando, 1994, pp. 1749–1754.

[46] Fuzzy logic and fuzzy control—Terms and definitions, VDI/VDE-Richtlinien, VDI/VDE 3550 Blatt 2, Beuth Verlag GmBH, Berlin, 2000.

[47] Y.C. Wang, S.G. Adams, J.S. Thorp, N.C. MacDonald, P. Hartwell, F. Bertsch, Chaos in MEMS, parameter estimation and its potential application, IEEE Trans. Circuits Systems 45 (10) (1998) 1013–1020.

[48] W.G. Wee, K.S. Fu, A formulation of fuzzy automata and its application as a model of learning systems, IEEE Trans. System Science Cybernet. 5 (1969) 439–447.

[49] Ch. Wissel, Theoretische Ökologie, Springer, Berlin, 1989.

[50] H. Yamamoto, T. Furuhashi, A new sufficient condition for stable fuzzy control system and its design method, IEEE Trans. Fuzzy Systems 9 (5) (2001) 554–569.