Monograph:
Guo, L.Z. and Billings, S.A. (2003) Identification of binary cellular automata from spatiotemporal binary patterns using a fourier representation. Research Report. ACSE Research Report no. 913 . Automatic Control and Systems Engineering, University of Sheffield

Identification of Binary Cellular Automata from Spatio-temporal Binary Patterns Using a Fourier Representation

Guo, L. Z. and Billings, S. A.

Department of Automatic Control and Systems Engineering
University of Sheffield
Sheffield, S1 3JD
UK

Research Report No. 913
February 2006

Identification of binary Cellular Automata from spatio-temporal binary patterns using a Fourier representation

Guo, L. Z. and Billings, S. A.

Department of Automatic Control and Systems Engineering
University of Sheffield
Sheffield S1 3JD, UK

Abstract

The identification of binary cellular automata from spatio-temporal binary patterns is investigated in this paper. Instead of using the usual Boolean or multilinear polynomial representation, the Fourier transform representation of Boolean functions is employed in terms of a Fourier basis. In this way, the orthogonal forward regression least-squares algorithm can be applied directly to detect the significant terms and to estimate the associated parameters. Compared with conventional methods, the new approach is much more robust to noise. Examples are provided to illustrate the effectiveness of the proposed approach.

1 Introduction

Cellular automata (CA) have been widely studied in recent years. The great importance of these studies comes from the fact that simple CA rules can be used to produce very complex patterns, such as spirals of the Belousov-Zhabotinsky reaction, spots and strips on animal skins, and the contours on human fingerprints etc., but also to provide an explanation about pattern formation in a variety of scientific fields (Ilachinski 2001, Deutsch and Dormann 2005, Gerhart, Schuster, and Tyson 1990, Greenberg, Hassard, and Hastings 1978). Whilst there is plenty of literature focusing on developing CA models theoretically and investigating the dynamical behaviour and pattern formation revealed by given CA models, the problem of identification or extraction of CA rules from observed spatio-temporal patterns has received far less attention. The CA identification problem is very important because of the importance of being able to produce a CA model directly from observed patterns. These extracted CA models can subsequently be used for the analysis of the mechanism of the observed pattern formation, for the simulation of the dynamical behaviours, and even for hardware implementation.

The CA identification problem consists of extracting the local transition rules and the associated neighbourhood over which the rule is operated, from observed, possibly noisy, spatio-temporal patterns. The identified CA rules should be parsimonious so that the set of the rules is as small as possible and the size of the neighbourhood is minimal. Several researchers have studied this problem (Adamatzky 1994, 1997, Adamatzky and Bronikov 1990, Burton 1996, Richards, Meyer, and Packard 1990, Yang and Billings 2000, Billings and Yang 2003a,b). Sequential and parallel algorithms for computing the local transition table were presented by Adamatzky (1994), and Richards (1990) introduced a method using genetic algorithms (GAs). However, no clear structure of the related neighbourhoods was obtained in either of these studies and the neighbourhood detection process was complicated. GAs were also employed in Yang and Billings (2000) to determine the rules as a set of logical operators. Simple local rules were found for low-dimensional problems, but when CAs with large-size neighbourhoods are involved the search process can be computationally demanding, this is due to the nature of the GA evolution. A multilinear polynomial form was used by Billings and Yang (2003a,b) to represent the underlying binary Boolean rules and a modified orthogonal least-squares algorithm was employed to detect the neighbourhood and the unknown model parameters. It has been shown that this method can be used to deal with high-dimensional problems in a very effective way. However, the method may deteriorate when the data is corrupted by noise. This is due to the fact that the coefficients of the multilinear polynomial representation of binary Boolean rules are integers. It follows that an identified CA model, from noisy data, may produce noninteger values and hence the method is not totally robust to noise. To retain the advantages of this approach, that is being able to deal with high-dimensional problems, while overcoming the sensitivity to noise, in this paper a new identification approach is proposed by associating the Orthogonal Forward Regression (OFR) algorithm (Billings, Chen, and Kronenberg 1989) with a threshold Fourier transform representation of binary Boolean functions. The main advantage of the approach arises because the Fourier transform representation of a binary Boolean function is in the field of real numbers so that the OFR algorithm can be applied directly whereas in the case of the multilinear polynomial representation, integer constraints must be considered (Billings and Yang 2003a, b). Moreover, the probability of an error when predicting using the threshold Fourier representation is bounded by the modelling error, which means the possibility of an error in the predictions can be made sufficiently small as long as the approximation error is sufficiently small (Mansour 1994).

The paper is organised as follows. Section 2 presents a brief introduction to binary CA models of spatio-temporal systems. Several different representations of binary CA rules are discussed in section 3. The new identification algorithm is derived in section 4 and section 5 describes simulation examples to demonstrate the potential of the proposed approach. Finally conclusions are given in section 6.

2 Binary CA models of spatio-temporal dynamical systems

The CAs, of interest in this paper, are dynamical systems in which space and time are discrete over regular grids of cells, each of which can be in one of a finite number of possible states,

updated synchronously in discrete time steps according to a local, identical transition rule. The state of a cell is determined by the local rule which depends on the states of a surrounding neighbourhood of cells. In this paper, binary CAs are investigated, that is the state of the cells can only take two values.

Formally a binary CA can be defined as follows.

Let I be a d-dimensional lattice consisting of the set of all integer coordinate vectors $i = (i_1, \cdots, i_d) \in \mathbf{Z}^d$. Suppose the state of a cell i in I at time instant t is determined by the previous states of a surrounding neighbourhood $\mathbf{n}_i = (i+n_1, i+n_2, \cdots, i+n_m)$ of the cell i. The CA model of a spatio-temporal dynamical system defined over I can be described as

$$x_i(t) = f(x_{\mathbf{n}_i}(t-1), \cdots, x_{\mathbf{n}_i}(t-k)) \tag{1}$$

where $x_i(t)$ is the state of the ith cell in I at time instant t, f is a function describing the local transition rule, and $x_{\mathbf{n}_i}(t-j) = (x_{i+n_1}(t-j), x_{i+n_2}(t-j), \cdots, x_{i+n_m}(t-j)), j = 1, 2, \cdots, k$ represents the previous states of the neighbourhood of the cell i with k the maximal time lag.

From the CA model (1) it can be observed that for a binary CA, the transition rule f is a Boolean function defined on $\{0,1\}^n$ taking values on $\{0,1\}$, where $n = m \times k$ is the total number of neighbouring cells. Note that the local transition rules can be defined in several equivalent ways. In what follows, some of these representations are discussed.

3 Binary CA rules and their representations

Consider the space of real functions from $\{0,1\}^n$ into $\mathbf{R}$, that is the set of $\mathcal{F}_n = \{f | f : \{0,1\}^n \to \mathbf{R}\}$. It follows that all Boolean functions are included in $\mathcal{F}_n$. Obviously, each $f \in \mathcal{F}_n$ can be regarded as a vector of 2^n real values and $\mathcal{F}_n$ is then a vector space of dimension 2^n with its standard basis.

3.1 Binary CA rules represented by a transition table

The most common method to define a CA rule is to use a transition table analogous to a truth table where the first row describes the state of the neighbourhood and the second row indicates the next state of the cells. As an example, the rules are then labelled by specifying which neighbourhoods map to zero and which to one. The standard form of a 1-dimensional 3-cell von Neumann neighbourhood rule R is shown below

$x_{i-1}(t-1)x_i(t-1)x_{i+1}(t-1)$	000	001	010	011	100	101	110	111
$x_i(t)$	r_0	r_1	r_2	r_3	r_4	r_5	r_6	r_7

(2)

where $r_l, l = 0, 1, \cdots, 7$ indicate the next states of the cells. The numerical label D assigned to the rule R is given uniquely by $D = \sum_{l=0}^{2^3-1} r_l 2^l$, which is simply the sum of the coefficients associated with all the non-zero components. For example, a 1-dimensinal 3-cell von Neumann neighbourhood rule $Rule30$ is defined as $Rule30 = (01111000)$ and the numerical label $D(Rule30) = 2^1 + 2^2 + 2^3 + 2^4 = 30$. From the vector space point of view, this rule can be expressed as a Boolean function in $\mathcal{F}_3$ with the standard basis of $\mathcal{F}_3$. In fact, the standard basis of vector space $\mathcal{F}_n$ can be defined by the so-called 'delta' functions, that is for each $\alpha \in \{0,1\}^n$, define $e_\alpha : \{0,1\}^n \to \{0,1\}$ as

$$e_\alpha(x) = \begin{cases} 1, & \text{if } x = \alpha \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

then all 2^n functions e_α forms a set of basis of the space $\mathcal{F}_n$. It follows that any Boolean function $f \in \mathcal{F}_n$ can be represented as

$$f(x) = \sum_{\alpha \in \{0,1\}^n} \theta_\alpha e_\alpha(x) \tag{4}$$

where the coefficients are $\theta_\alpha = f(\alpha)$.

For the above special case $n = 3$, it is clear that the first row of the rule can be regarded as the basis functions and with the second row, the above rule can be written as a Boolean function as follows

$$R(x) = r_0 e_{000}(x) + r_1 e_{001}(x) + r_2 e_{010}(x) + r_3 e_{011}(x) + r_4 e_{100}(x) + r_5 e_{101}(x) + r_6 e_{110}(x) + r_7 e_{111}(x) \tag{5}$$

Accordingly, a binary CA model of form (1) of this system can be written as

$$\begin{aligned} x_i(t) &= R(x_{i-1}(t-1), x_i(t-1), x_{i+1}(t-1)) \\ &= \sum_{\alpha \in \{0,1\}^3} r_\alpha e_\alpha(x_{i-1}(t-1), x_i(t-1), x_{i+1}(t-1)) \end{aligned} \tag{6}$$

where $\alpha \in \{0,1\}^3$.

3.2 The Boolean form of CA rules

The Boolean form of CA rules can be constructed using some simple logical operators such as NOT, AND, and OR etc. The rules for all 1-dimensional CA with a 3-cell neighbourhood can be found in Wolfram (1994). The Boolean form of $Rule30$, for example, is

$$\begin{aligned} x_i(t) &= Rule30(x_{i-1}(t-1), x_i(t-1), x_{i+1}(t-1)) \\ &= (x_{i-1}(t-1) \wedge \bar{x}_i(t-1) \wedge \bar{x}_{i+1}(t-1)) \vee (\bar{x}_{i-1}(t-1) \wedge \\ &\quad x_i(t-1)) \vee (\bar{x}_{i-1}(t-1) \wedge x_{i+1}(t-1)) \end{aligned} \tag{7}$$

where $-, \wedge$, and $\vee$ denote the NOT, AND, and OR operators respectively.

The Boolean form of CA rules can also be represented using only AND and XOR operators (Billings and Yang 2003b). For the same example $Rule30$, this kind of Boolean form is

$$\begin{aligned} x_i(t) &= Rule30(x_{i-1}(t-1), x_i(t-1), x_{i+1}(t-1)) \\ &= a_0 \oplus a_1 x_{i-1}(t-1) \oplus a_2 x_i(t-1) \oplus a_3 x_{i+1}(t-1) \oplus a_4(x_{i-1}(t-1) \wedge x_i(t-1)) \oplus \\ &\quad a_5(x_{i-1}(t-1) \wedge x_{i+1}(t-1)) \oplus a_6(x_i(t-1) \wedge x_{i+1}(t-1)) \oplus \\ &\quad a_7(x_{i-1}(t-1) \wedge x_i(t-1) \wedge x_{i+1}(t-1)) \end{aligned} \tag{8}$$

where $\oplus$ denotes the XOR operator, $a_i, i = 0, 1, \cdots, 7 = 2^3 - 1$ are binary numbers and $a_i = 1$ indicates that the corresponding term is included in the Boolean function while $a_i = 0$ indicates that the corresponding term is not included. A general form of (9) for the n-variable case with an m-site neighbourhood of $\mathbf{n}_i = (i + n_1, i + n_2, \cdots, i + n_m)$ and maximal time lag k ($n = m \times k$) can be written as

$$x_i(t) = a_0 \oplus a_1 x_{i+n_1}(t-1) \oplus \cdots \oplus a_{2^n-1}(x_{i+n_1}(t-k) \wedge \cdots \wedge x_{i+n_m}(t-k)) \tag{9}$$

Note that this representation is unique, that is for an n-variable problem, one set of $\{a_i, i = 0, 1, \cdots, 2^n - 1\}$ corresponds to one and only one n-variable CA rule. This holds for higher dimensional CAs.

3.3 Multilinear polynomial form of CA rules

Let $f \in \mathcal{F}_n$ be a Boolean function. A real multilinear polynomial $p : \mathbf{R^n} \to \mathbf{R}$ is called a representation of f over the real field if for every $x \in \{0,1\}^n$, $f(x) = p(x)$. It is well known that every Boolean function f can be represented as a polynomial f_p over the ring of integers (Schneeweiss 1989, 1998, Billings and Yang 2003a, b), that is

$$f_p(x) = f_p(x_1, x_2, \cdots, x_n) = c_0 + \sum_{i=1}^{m} (c_i \prod_{j \in I_i} x_j), I_i \subset \{1, 2, \cdots, n\} \tag{10}$$

This representation is unique and the coefficients are integers (c_0 is Boolean). With this multilinear polynomial representation of the rules, a binary CA with an m-site neighbourhood of $\mathbf{n}_i = (i + n_1, i + n_2, \cdots, i + n_m)$ and maximal time lag k ($n = m \times k$) can be written as

$$x_i(t) = c_0 + \sum_{j=1}^{m}\sum_{l=1}^{k} c_{j,l} x_{i+n_j}(t-l) + \sum_{j_1 \geq j_2=1}^{m}\sum_{l_1 \geq l_2=1}^{k} c_{j_1,j_2,l_1,l_2} x_{i+n_{j_1}}(t-l_1) x_{i+n_{j_2}}(t-l_2) + \cdots + c_{2^n-1} x_{i+n_1}(t-1)\cdots x_{i+n_m}(t-k) \quad (11)$$

3.4 CA rules represented by the Fourier basis

Clearly, each $f \in \mathcal{F}_n$ can be regarded as a vector of 2^n real values and $\mathcal{F}_n$ is then a vector space of dimension 2^n with its standard basis. Moreover, the space $\mathcal{F}_n$ can be endowed with an inner product as follows

$$< f, g >= \frac{1}{2^n} \sum_{x \in \{0,1\}^n} f(x) g(x) \quad (12)$$

The norm of a function f in $\mathcal{F}_n$ is then $\|f\| = \sqrt{< f, f >}$.

The Fourier basis of the space $\mathcal{F}_n$ consists of 2^n functions: for each $\alpha \in \{0,1\}^n$, $\chi_\alpha : \{0,1\}^n \to \{-1,+1\}$ is defined by

$$\chi_\alpha(x) = (-1)^{\sum_{i=1}^{n} x_i \alpha_i} \quad (13)$$

There are several important properties of the above Fourier basis:

- The basis is normal, that is $\|\chi_\alpha\| = 1$.
- The basis is orthogonal, that is

$$< \chi_\alpha, \chi_\beta >= \begin{cases} 1, & \text{if } \alpha = \beta \\ 0, & \text{otherwise} \end{cases} \quad (14)$$

- The number of Fourier basis is 2^n.

  Following the above three properties, the Fourier basis forms an orthonormal basis and any function f in $\mathcal{F}_n$ can uniquely be represented as a linear combination of the basis functions, that is

$$f(x) = \sum_{\alpha \in \{0,1\}^n} \theta_\alpha \chi_\alpha(x) \quad (15)$$

  where $\theta_\alpha = \hat{f}(\alpha) =< f, \chi_\alpha >$.

- Parsevals identity: $\sum_{\alpha\in\{0,1\}^n} \hat{f}(\alpha)^2 = 1/2^n \sum_{x\in\{0,1\}^n} f(x)^2$.

  It follows that for Boolean functions taking the values on $\{-1,+1\}$, Parsevals identity is $\sum_{\alpha\in\{0,1\}^n} \hat{f}(\alpha)^2 = 1$.

- Let h be a function in $\mathcal{F}_n$. Define the sign function or threshold to be

$$Sign(h)(x) = \begin{cases} +1, & \text{if } h(x) \geq 0 \\ -1, & \text{otherwise} \end{cases} \tag{16}$$

  If $f \in \mathcal{F}_n$ is a Boolean function taking values on $\{-1,+1\}$, then

$$Pr[f(x) \neq Sign(h(x))] \leq E[(f-h)^2] \tag{17}$$

  where $Pr[\cdot]$ denotes the probability and $E[\cdot]$ is the expectation.

With this Fourier basis, a binary CA with an m-site neighbourhood of $\mathbf{n}_i = (i+n_1, i+n_2, \cdots, i+n_m)$ and maximal time lag k ($n = m \times k$) can be written as

$$x_i(t) = \sum_{\alpha\in\{0,1\}^n} \theta_\alpha \chi_\alpha(x_{i+n_1}(t-1), \cdots, x_{i+n_m}(t-k)) \tag{18}$$

3.5 Comments on CA identification using different representations of Boolean functions

The objective of binary CA identification for spatio-temporal dynamical systems is to obtain a binary CA model, that is the local transition rule, from observed spatio-temporal patterns. Any one of the above equivalent representations of the local transition functions can be used to describe the underlying spatio-temporal dynamics once the representation is estimated or extracted by using some identification algorithm. Ideally, the identification technique should be able to produce a concise expression of the rule. This ensures that the obtained CA model is parsimonious and can readily be interpreted either for simulation or hardware implementation of the CA. Note that all of the above four representations (5), (9), (11), and (18) of the CA rules are in a form that is linear-in-the-parameters based on either the field of real numbers, the ring of integers, or Boolean algebra, therefore, theoretically any least-squares-like algorithm can be employed to produce an approximation of the rule. However, there are several difficulties associated with the CA identification problem that need to be addressed:

1. The neighbourhood of a cell is the set of cells over both space and time that are directly involved in the evolution of the cell. The size of a neighbourhood determines the dimension of the local rules. Most commonly considered neighbourhoods are the von Neumann and the Moore neighbourhoods. However, there are many more possible neighbourhood structures for higher dimensional CAs. For instance, the CA model of reaction-diffusion systems

generally involves two different areas in the space domain, which interact with each other to mimic the reaction process and the diffusion process. Because the neighbourhood of a cell in a spatio-temporal system involves cells from different spatial and temporal scales, it follows that the size of the neighbourhood for a higher dimensional CA can be very large. This in turn can make the number of terms in the linear-in-the-parameters expression very large, which can then make the direct application of a least-squares-type algorithm difficult.

2. Generally, when identifying CA rules, the only *a priori* knowledge that is available will be the observed spatio-temporal patterns produced by the evolution of the underlying spatio-temporal system. The neighbourhood structure will most likely to be unknown and this means that the possible combinations can number into the millions. Furthermore, as the size of the neighbourhood, or the dimensionality, or both increase the combinational possibilities become very large.

3. Noise can be caused by imperfect measurements or uncertainties due to an incorrect neighbourhood structure. A most serious consequence caused by noise is the loss of the Boolean property in the identified transition functions.

A few authors have appreciated the neighbourhood size problem (Adamatzky 1994, Richards, et al 1990) and have therefore focused on a very limited class of low-dimensional CAs. Billings and Yang (2003a,b) presented a possible solution to this problem by combining the polynomial representation (11) with a modified orthogonal least-squares method. The main advantage of this method is that the identification problem is mapped into an integer polynomial in a linear-in-the-parameters form and an orthogonal least squares algorithm can then be used to detect the model structure or neighbourhood and produce the parameter estimates in a stepwise manner. Consequently, higher dimensional problems can be dealt with. However, the method may become sensitive to noise or uncertainty because the Boolean property of the identified CA rules can become corrupted by the approximate error and the estimated parameters can become real numbers rather than integers in the presence of noise. Because the multilinear polynomial representation of the rule is in terms of integers, integer parameter estimation techniques (Hassibi and Boyd 1998) can be applied. However, these authors showed that the estimates can be poor if the parameters are treated as being real and then rounded to be integers. Boolean regression methods (Boros, Hammers, and Hooker 1995) suffer from similar problems, and post processing using a GA algorithm (Billings and Yang 2003a) is computationally very expensive.

To overcome these problems, in this paper a totally new approach is proposed by applying the Orthogonal Forward Regression (OFR) least-squares algorithm to a thresholded Fourier representation of CA rules. The proposed method has the following characteristics.

- The Fourier representation of CA rules is in the field of real numbers. Compared with the representation in Boolean algebra and integer ring, there is no need to use any constraints when applying the identification algorithm.

- By using a thresholded Fourier representation the property of being Boolean can always be retained.

- The property of a thresholded Fourier representation guarantees as long as the modelling error is sufficiently small, the identified model can be sufficiently accurate and robust to noise.
- The OFR least-squares algorithm can effectively detect the neighbourhood structure and provide parameter estimates in a forward term selection manner when the neighbourhood structure is unknown.

In the following section, the OFR algorithm and some simulation examples will be presented based on the new Fourier basis approach.

4 The identification algorithm

Rewrite (18) using the threshold operation to give

$$x_i(t) = f(x_{i+n_1}(t-1), \cdots, x_{i+n_m}(t-k)) = Sign(\sum_{\alpha \in \{0,1\}^n} \theta_\alpha \chi_\alpha(x_{i+n_1}(t-1), \cdots, x_{i+n_m}(t-k))) \quad (19)$$

Note that in this way the output domain is $\{-1, +1\}$ while the input domain is $\{0, 1\}^n$. It follows that when doing model prediction after identification, the $x_i(t)$ should be transformed back to the domain $\{0, 1\}$ as the input at the next time instant if necessary. The objective of the CA identification for spatio-temporal dynamic systems is to determine f in (19), that is to determine the neighbourhood structure and to estimate the parameters. In order to obtain the unknown neighbourhood structure, initially a large possible neighbourhood should be chosen, which should include the correct neighbourhood as a subset. From the initial neighbourhood, a set of candidate model terms can be constructed according to the Fourier basis. The Orthogonal Forward Regression algorithm (OFR) (Chen, Billings, and Luo 1989) is then employed. The OFR algorithm involves a stepwise orthogonalisation of the regressors and a forward selection of the relevant terms based on the Error Reduction Ratio (ERR) criterion (Billings, Chen, and Kronenberg 1989). The algorithm provides the optimal least-squares estimate of the associated coefficients θ.

For a given candidate regressor set $G = \{\varphi_m\}_{m=1}^M$, the OFR algorithm can be summarised as follows

1. **Step 1**

$$I_1 = I_M = \{1, \cdots, M\}$$

$$w_m(t) = \varphi_m(t), \hat{b}_m = \frac{w_m^T y}{w_m^T w_m} \quad (20)$$

$$l_1 = arg\max_{m\in I_1}(\hat{b}_m^2 \frac{w_m^T y}{y^T y}) = arg\max_{m\in I_1}(err_m) \tag{21}$$

$$w_1^0 = w_{l_1}, c_1^0 = \frac{w_1^{0T} y}{w_1^{0T} w_1^0} \tag{22}$$

$$a_{1,1} = 1 \tag{23}$$

2. **Step** $j, j > 1$

$$I_j = I_{j-1} \backslash l_j - 1 \tag{24}$$

$$w_m(t) = \varphi_m(t) - \sum_{k=1}^{j-1} \frac{w_k^{0T} y}{w_k^{0T} w_k^0} w_k^0, \hat{b}_m = \frac{w_m^T y}{w_m^T w_m} \tag{25}$$

$$l_j = arg\max_{m\in I_j}(\hat{b}_m^2 \frac{w_m^T y}{y^T y}) = arg\max_{m\in I_j}(err_m) \tag{26}$$

$$w_j^0 = w_{l_j}, c_j^0 = \frac{w_j^{0T} y}{w_j^{0T} w_j^0} \tag{27}$$

$$a_{k,j} = \frac{w_k^{0T} \varphi_{l_j}}{w_k^{0T} w_k^0}, k = 1, \cdots, j-1. \tag{28}$$

The procedure is terminated at the M_s-th step when the termination criterion

$$1 - \sum_{m=1}^{M_s} err_m < \rho \tag{29}$$

is met, where ρ is a designated error tolerance, or when a given number of terms in the final model is reached.

The estimated coefficients are calculated from the following equation

$$\begin{pmatrix} \theta_{l_1} \\ \theta_{l_2} \\ \vdots \\ \theta_{l_{M_s}} \end{pmatrix} = \begin{pmatrix} 1 & a_{1,2} & \cdots & a_{1,M_s} \\ 0 & 1 & \vdots & a_{2,M_s} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}^{-1} \begin{pmatrix} c_1^0 \\ c_2^0 \\ \vdots \\ c_{M_s}^0 \end{pmatrix} \tag{30}$$

and the selected terms are $\varphi_{l_1}, \cdots, \varphi_{l_{M_s}}$.

Based on this algorithm, the identification procedure for CA models of spatio-temporal systems can be summarised as follows

1. **Step 1:** Select the largest possible spatial neighbourhood sites $n_1, \cdots, n_m$.
2. **Step 2:** Select the maximal possible time lags k and then calculate all the candidate model terms using the data and the Fourier basis.
3. **Step 3:** Apply the OFR algorithm to obtain the terms (neighbourhood) and parameters of the CA model.
4. **Step 4:** Apply model validity tests to evaluate the model. If no valid models are found, go back to Step 1 and reset the candidate terms to include a larger spatio-temporal neighbourhood.
5. **Step 5:** Validate the final CA model.

Remark 1 The final model and parameters should be validated as the final step in the identification. A commonly used approach is to check one-step-ahead predictions or model predictions. In this paper, model predictions will be used. Furthermore, the Parseval's identity for Boolean functions: $\sum_{\alpha\in\{0,1\}^n} \hat{f}(\alpha)^2 = 1$ can also be used to test the suitability of the final CA model.

Remark 2 Note that in the above identification procedure, the spatial neighbourhood sites and the time lags of the identified site need to be set known *a priori*. In other words, the neighbourhood of the identified site, that is, the region around that site which influences the dynamics of that site in the spatial domain and in the time domain need to be set known before starting the identification. In practice, these two factors are important in determining the spatio-temporal dynamics of the underlying system. This problem is related to the embedding dimension problem in system reconstruction theory (Casdagli, 1992).

5 Numerical simulations

5.1 Example 1: Identification of a 1-dimensional 3-site CA $Rule30$

As discussed earlier, the standard form of a 1-dimensional 3-cell rule with a von Neumann neighbourhood is shown below

$$\begin{array}{cccccccccc} x_{i-1}(t-1)x_i(t-1)x_{i+1}(t-1) & 000 & 001 & 010 & 011 & 100 & 101 & 110 & 111 \\ x_i(t) & r_0 & r_1 & r_2 & r_3 & r_4 & r_5 & r_6 & r_7 \end{array} \tag{31}$$

where $r_i, i = 0, 1, 2, \cdots, 7$ indicates the next state of the cells. The 1-dimensinal 3-cell rule $Rule30$ with von Neumann neighbourhood can then be written as $Rule30 = (01111000)$ with the

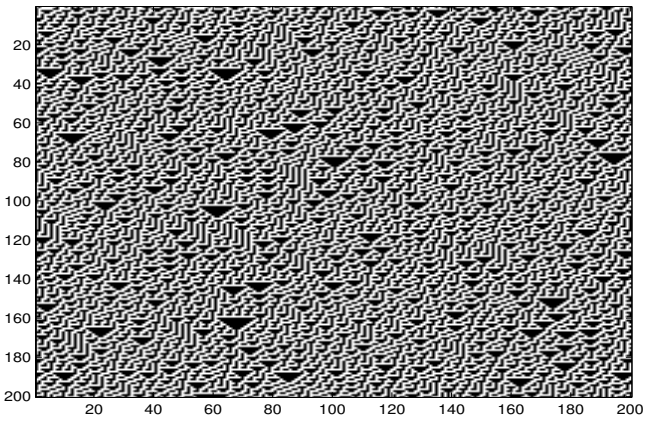

Figure 1: Example 1: Spatio-temporal patterns formed by the evolution of the one dimensional CA *Rule*30 with von Neumann neighbourhood

numerical label $D(Rule30) = 2^1 + 2^2 + 2^3 + 2^4 = 30$. The spatio-temporal patterns generated by the evolution of the *Rule*30 over a 200×200 spatio-temporal lattice with a von Neumann neighbourhood $x_{i-1}(t-1), x_i(t-1)$, and $x_{i+1}(t-1)$ is shown in Fig. (1).

For the purpose of identification, the initial possible neighbourhood was assumed to be $x_{i-2}(t-1), x_{i-1}(t-1), x_i(t-1), x_{i+1}(t-1), x_{i+2}(t-1), x_{i+1}(t-2)$, which is the same as that in Billings and Yang (2003b). Then the thresholded form of the Fourier representation of *Rule*30 with this initial neighbourhood is given by

$$\begin{aligned} x_i(t) &= Sign\Big(\sum_{\alpha\in\{0,1\}^6} \theta_\alpha \chi_\alpha(x_{i-2}(t-1), x_{i-1}(t-1), x_i(t-1), x_{i+1}(t-1), x_{i+2}(t-1), x_{i+1}(t-2))\Big) \quad (32) \\ &= Sign\Big(\sum_{\alpha\in\{0,1\}^6} \theta_\alpha (-1)^{x_{i-2}(t-1)\alpha_1 + x_{i-1}(t-1)\alpha_2 + x_i(t-1)\alpha_3 + x_{i+1}(t-1)\alpha_4 + x_{i+2}(t-1)\alpha_5 + x_{i+1}(t-2)\alpha_6}\Big) \end{aligned}$$

where $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_6)$. The objective of CA identification is to determine which terms should be retained in this model and to obtain an estimate of the corresponding parameters θ_α. The data used in the identification were extracted randomly from the 200×200 spatio-temporal cells. After applying the algorithm in section 4, the final model and parameters obtained are shown in Table (1) and the model predicted output is shown in Fig. (2), where Terms represents the selected model terms, Estimates are the associated parameters, and ERR indicates the error reduction ratio from the OFR algorithm (Billings, Chen, and Kronenberg 1989). It is noteworthy that the estimated parameters satisfy Parseval's identity. Moreover, from Fig. (2) it can be observed that there is no prediction error, which indicates an excellent identification and prediction result.

To test the sensitivity of the proposed approach to noise or uncertainty, the data used for identification were corrupted with noise. This was achieved by randomly flipping some of the states

Terms	Estimates	ERR
$(-1)^{y_{i-1}(t-1)}$	5.0000e-01	3.6000e-01
$(-1)^{y_{i-1}(t-1)+y_i(t-1)+y_{i+1}(t-1)}$	-5.0000e-01	2.0364e-01
$(-1)^{y_{i-1}(t-1)+y_i(t-1)}$	-5.0000e-01	1.9166e-01
$(-1)^{y_{i-1}(t-1)+y_{i+1}(t-1)}$	-5.0000e-01	2.4471e-01

Table 1: Example 1: The terms and parameters of the final CA model using the threshold Fourier representation

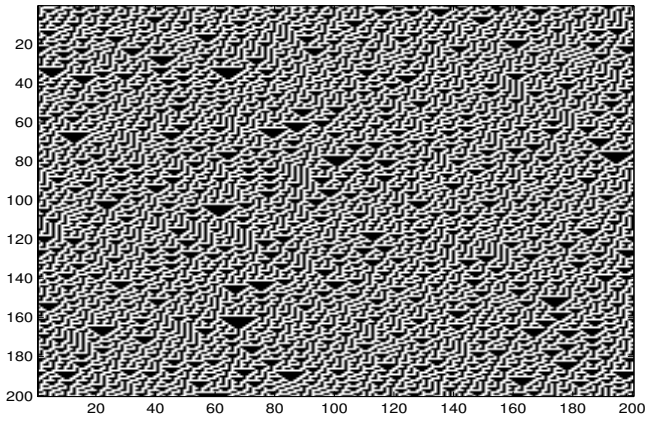

Figure 2: Example 1: Model predicted output using the identified CA model and the thresholded Fourier representation in Table 1

Terms	Estimates	ERR
$(-1)^{y_{i-1}(t-1)+y_i(t-1)+y_{i+1}(t-1)}$	-5.2748e-01	3.3640e-01
$(-1)^{y_{i-1}(t-1)+y_{i+1}(t-1)}$	-4.3018e-01	1.7217e-01
$(-1)^{y_{i-1}(t-1)+y_i(t-1)}$	-4.6830e-01	1.9075e-01
$(-1)^{y_{i-1}(t-1)}$	4.3440e-01	1.8934e-01
$(-1)^{y_{i-1}(t-2)+y_{i+1}(t-1)+y_{i+2}(t-1)}$	7.0325e-02	4.8764e-03

Table 2: Example 1: The terms and parameters of the final CA model using the threshold Fourier representation from noisy data

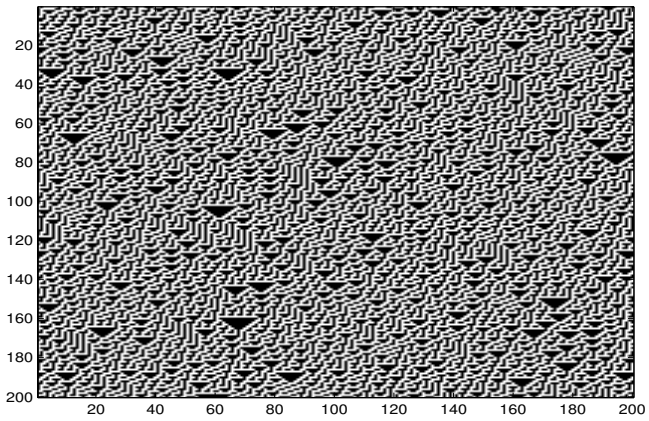

Figure 3: Example 1: Model predicted output using the identified CA model with noise and the thresholded Fourier representation in Table 2

of the updated cells from 1 to 0 or from 0 to 1. The identified model and estimated parameters based on this noisy data set are shown in Table (2) and Fig. (3). From the final model it can be seen that a false neighbour $y_{i-1}(t-2)$ has been chosen by the algorithm and also that the square-summation of the estimated parameters is $9.3608e-01$ when it should be 1 according to the requirement of Parseval's identity. All of these observations indicate that the model obtained from noisy data is an approximate CA model. Nevertheless, Fig. (3) shows that there is no prediction error which shows the very good prediction ability of the thresholded Fourier representation of Boolean functions.

5.2 Example 2: Binary CA rule identification of vertebrate skin patterns

It is well known that the patterns on vertebrate skin are of great importance to the survival of the species because they represent camouflage, species identification, and/or warning patterns.

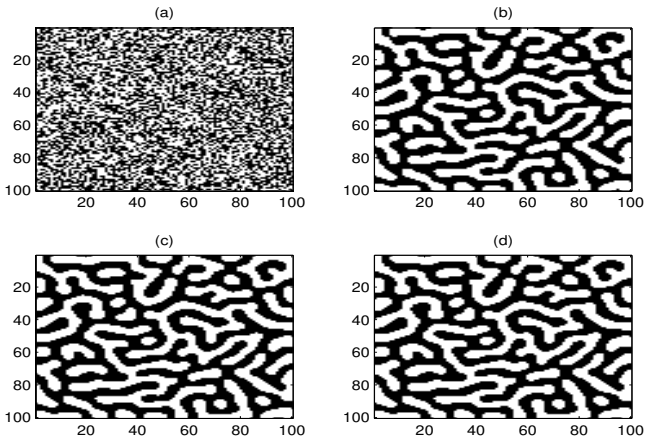

Figure 4: Example 2: Vertebrate skin patterns at the time steps: (a) $t = 1$, (b) $t = 8$, (c) $t = 9$, and (d) $t = 10$

Typical vertebrate skin patterns include spots and stripes. Striped patterns are not only seen in animals but also in human fingerprints. From both an evolutionary and a mathematical point of view, it is of great interest to investigate how such skin patterns are formed. Since Turing's reaction-diffusion model (Turing 1952) many different models have been presented, for example, Suzuki, Takayama, Motoike, and Asai (2006), and Young (1984). However, there are very few studies that have considered the identification of a CA model from observed or recorded vertebrate skin patterns. In this section, the new Fourier basis proposed identification method will be used to obtain a simple binary CA model directly from observed patterns. For the purpose of numerical simulation, example patterns were generated using Young's model (Young 1984) over a 100×100 spatial lattice and are shown in Fig. (4). The activation area had a radius of 2.30 and the inhibition area had an outer radius of 6.01. Therefore, roughly the neighbourhood involved about 225 spatial cells. If the time lag is considered as 1, the Fourier basis consists of $2^{225} = 5.3920e + 067$ terms which is not realistic for identification. In this identification example therefore, the initial spatial neighbourhood was set to be the Moore neighbourhood with a radius of 1 and the time lag was assumed to be 1.

The final identified CA model is shown in Table (3) and the model predicted patterns are shown in Fig. (5), which indicates a good result.

6 Conclusions

A new identification method for binary CA models of spatio-temporal dynamic systems has been proposed. The proposed method is a combination of the threshold Fourier representation of Boolean functions and an orthogonal forward regression identification algorithm. The new method introduced a new approach to CA identification techniques for spatio-temporal dynamical

Terms	Estimates	ERR
$(-1)^{y_{i,j}(t-1)}$	-4.3771e-01	6.4000e-01
$(-1)^{y_{i+1,j}(t-1)}$	-3.6346e-01	6.4533e-02
$(-1)^{y_{i-1,j+1}(t-1)}$	-3.5913e-01	4.3562e-02
$(-1)^{y_{i,j}(t-1)+y_{i+1,j}(t-1)+y_{i-1,j+1}(t-1)}$	1.2087e-01	4.1232e-02
$(-1)^{y_{i,j-1}(t-1)+y_{i-1,j}(t-1)+y_{i,j}(t-1)+y_{i-1,j+1}(t-1)+y_{i,j+1}(t-1)}$	-2.0270e-01	1.7281e-02
$(-1)^{y_{i,j-1}(t-1)+y_{i,j}(t-1)+y_{i-1,j+1}(t-1)}$	1.4639e-01	2.0924e-02
$(-1)^{y_{i,j-1}(t-1)+y_{i+1,j-1}(t-1)+y_{i,j}(t-1)}$	-2.7958e-01	1.4924e-02
$(-1)^{y_{i-1,j-1}(t-1)+y_{i-1,j}(t-1)+y_{i+1,j}(t-1)+y_{i-1,j+1}(t-1)+y_{i,j+1}(t-1)}$	2.2127e-01	1.6312e-02
$(-1)^{y_{i-1,j-1}(t-1)+y_{i+1,j}(t-1)+y_{i+1,j+1}(t-1)}$	-2.5408e-01	1.6102e-02
$(-1)^{y_{i,j-1}(t-1)+y_{i+1,j-1}(t-1)+y_{i,j}(t-1)+y_{i+1,j}(t-1)+y_{i+1,j+1}(t-1)}$	1.6063e-01	1.9184e-02
$(-1)^{y_{i-1,j}(t-1)+y_{i,j}(t-1)+y_{i+1,j}(t-1)+y_{i,j+1}(t-1)+y_{i+1,j+1}(t-1)}$	1.4089e-01	1.2709e-02
$(-1)^{y_{i-1,j-1}(t-1)+y_{i+1,j-1}(t-1)+y_{i,j}(t-1)+y_{i-1,j+1}(t-1)+y_{i,j+1}(t-1)}$	1.1902e-01	9.0570e-03

Table 3: Example 2: The terms and parameters of the final CA model using the threshold Fourier representation

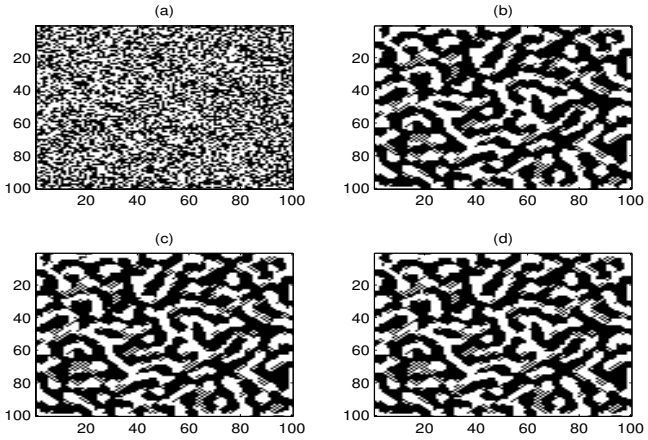

Figure 5: Example 2: Vertebrate skin patterns produced by the identified CA model at the time steps: (a) $t = 1$, (b) $t = 8$, (c) $t = 9$, and (d) $t = 10$

systems, which is more robust to the presence of noise or uncertainty. The application of the proposed method has been demonstrated by several numerical simulations.

7 Acknowledgement

The authors gratefully acknowledge financial support from EPSRC (UK).

References

[1] Adamatzky, A. I., (1994) *Identification of Cellular Automata*, London: Taylor & Francis.

[2] Adamatzky, A. I., (1997) Automatic programming of cellular automata: identification approach, *Kybernetes*, Vol. 26, No. 23, pp. 126-139.

[3] Adamatzky, A. I. and Bronikov, V. (1990) Identification of additive cellular automata, *J. Comput. Syst. Sci.*, Vol. 28, pp. 47-51.

[4] Billings, S. A., Chen, S., and Kronenberg, M. J., (1989) Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator, *Int. J. Contr.*, Vol. 49, pp. 2157-2189.

[5] Billings, S. A. and Yang, Y. X., (2003a) Identification of probabilistic cellular automata, *IEEE Trans. Syst., Man, Cybern. B*, Vol. 33, No. 2, pp. 225-236.

[6] Billings, S. A. and Yang, Y. X., (2003b) Identification of the neighbourhood and CA rules from spatio-temporal CA patterns, *IEEE Trans. Syst., Man, Cybern. B*, Vol. 33, No. 2, pp. 332-339.

[7] Boros, E., Hammers, P. L., and Hooker, J. N., (1995) Boolean regression, *Annals of Operation Research*, Vol. 58, pp. 201-226.

[8] Burton, H. V., (1996) *Computational Analysis of One-Dimensional Cellular Automata*, Singapore: World Scientific Series on Nonlinear Science, Series A, Vol. 15.

[9] Casdagli, M., (1992) A dynamical systems approach to modelling input-output systems, in *Nonlinear modelling and forcasting*, Casdagli and Eubank (eds.), Addison-Wesley Publishing Co., pp. 266-281.

[10] Chen, S., Billings, S. A., and Luo, W., (1989) Orthogonal least squares methods and their application to non-linear system identification, *International Journal of Control*, Vol. 50, No. 5, pp. 1873-1896.

[11] Deutsch, A. and Dormann, S., (2005) *Cellular Automaton Modelling of Biological Pattern Formation: Characterization, application and Analysis*, Boston: Birkh´auser.

[12] Gerhart, M., Schuster, H., and Tyson, J. J., (1990) A cellular automaton model of excitable media: II. Curvature, dispersion, rotating waves and meandering waves, *Physica D*, Vol. 46, No. 3, pp. 392-415.

[13] Greenberg, J. M., Hassard, B. D., and Hastings, S. P., (1978) Pattern formation and periodic structures in systems modelled by reaction-diffusion equations, *Bulletin of the American Mathematical Society*, Vol. 84, pp. 1296-1327.

[14] Hassibi, A. and Boyd, S, (1998) Integer parameter estimation in linear models with applications to GPs, *IEEE Trans. Signal Processing*, Vol. 46, pp. 2938-2952.

[15] Ilachinski, A., (2001) *Cellular Automata: a Discrete Universe*, Singapore: World Scientific.

[16] Mansour, Y., (1994) Learning Boolean functions via the Fourier transform, In *Theoretical Advances in Neural Computation and Learning*, V.P. Roychodhury and K-Y. Siu and A. Orlitsky, ed., Kluwer Academic, Boston, MA, pp. 391-424.

[17] Richards, F. C., Meyer, T. P., and Packard, N. H., (1990) Extracting cellular automaton rules directly from experiment data, *Physica D*, Vol. 45, pp. 189-202.

[18] Schneeweiss, W., (1998) On the polynomial form of Boolean functions: derivations and applications, *IEEE Trans. Computers*, Vol. 47, No.2, pp. 217-221.

[19] Schneeweiss, W., (1989) *Boolean Functions with Engineering Applications and Computer Programs*, Berlin, New York, Tokyo: Springer.

[20] Suzuki Y., Takayama T., Motoike I.N., and Asai T., (2006) Striped and spotted pattern generation on reaction-diffusion cellular automata – theory and LSI implementation–, *International Journal of Unconventional Computing*, vol. 2, no. 3 & 4.

[21] Turing, A. M., (1952) The chemical basis of morphogenesis, *Phil. Trans. R. Soc. Lond. B*, Vol. 237, pp. 37-72.

[22] Wolfram, S., (1994) *Cellular Automata and Complexity*, ser. MA. Reading: Addison-Wesley.

[23] Yang, Y. X. and Billings, S. A., (2000) Extracting Boolean rules from CA patterns, *IEEE Trans. Syst., Man, Cybern. B*, Vol. 30, pp. 573-580.

[24] Young, D. A., (1984) A local activator-inhibitor model of vertebrate skin patterns, *Mathematical Biosciences*, Vol. 72, pp. 51-58.