Identification of Dynamical Models of Spatio-temporal Pattern Formation with Applications to Slime Mould

Shu Wang

A thesis submitted to the University of Sheffield for the degree of

Doctor of Philosophy

Department of Automatic Control and Systems Engineering

University of Sheffield

Sheffield, UK

November 2011

Abstract

Spatio-temporal systems are non-linear dynamical systems which include both time and space information. Many natural phenomena and processes can be described by spatio-temporal models, such as pattern formation, chemical reactions and physical dynamics. The main focus of this thesis is on the investigation of analysis and identification methods for spatio-temporal systems and the application of these methods to the dynamics of slime mould.

This thesis starts with a review of recent developments of spatio-temporal systems. Three general classes of spatio-temporal systems, Cellular Automata (CA), Coupled Map Lattices (CML) and Partial Differential Equations (PDE), which can be applied to modelling the behaviours of slime mould are discussed. Some basic problems associated with the identification of these models are addressed from various viewpoints. The main objective of this thesis is to develop the previous work in this area and to derive effective models from observed spatio-temporal data.

The dynamics of slime mould at the aggregation stage can be viewed as a spatio-temporal system. Three models which represent different types of spatio-temporal models respectively are introduced to model pattern formation of slime mould. All models can produce similar spirals and circle patterns with the observed patterns in experiments.

For the identification problem of the above mentioned models, one commonly used method is the orthogonal least squares (OLS) algorithm or the orthogonal forward regression (OFR) algorithm. However, when this classical identification method is applied to spatio-temporal data it may select spurious model terms in some cases, so a new algorithm called Orthogonal Forward Regression using Mutual Information (OFR-MI) algorithm is proposed. A new criterion of selecting model terms based on mutual information (MI) is employed in the new identification method and this effectively avoids the problem in the classical OLS or OFR algorithm.

Inspired from a Reaction-Diffusion-Chemotaxis (RDC) model for the gathering problem of slime mould, a new CA model which is called probabilistic multi-rule CA model is proposed. Unlike general CA models, this new model has two or more transition rules with associated probabilities, so that it has the potential to be used to model random processes in some spatio-temporal systems.

The identification of the probabilistic multi-rule CA system is a challenging topic, because of the stochastic character of this model. Based on the OLS algorithm and statistical methods, a new identification algorithm for probabilistic multi-rule CA models is proposed. Simulation results show that this new algorithm can work well either on the noise-free patterns in one-dimensional and two-dimensional cases or on spatio-temporal patterns with static noise or dynamic noise.

Acknowledgements

First and foremost, I would like to express my deep sense of gratitude to my two supervisors Professors Stephen A. Billings and Daniel Coca for their insightful supervision, valuable guidance, and constant encouragement during my PhD study at the University of Sheffield. I cannot say thank you enough for their tremendous support and help for this thesis. Without their thoughtful and creative suggestions on the research and writing, I could not have finished my thesis.

I wish to acknowledge the University of Sheffield, EPSRC(UK) and the European Research Council for the financial support for this project.

Besides, I would like to thank Dr. Hualiang Wei, Dr. Yuzhu Guo, Mr. Pengfei Guo for their generous help and useful suggestions on my PhD work. My thanks and appreciations also go to my colleagues and supportive technician in the department of Automatic Control and Systems Engineering for their constant support, in particular, Dr. Yifan Zhao, Mr Dazhi Jiang, Mr. Xiliang Zhang, Dr. Yang Li, Dr. Zhuoyi Song, Miss Yu Liu, Mrs Frances Gail Bright and Mr. Paul Staniforth.

I am grateful to all my friends who were with me in Sheffield for their kind accompany and encouragement, in particular, Mr. Richard J Webb, Miss Jennifer Butler, Miss Alexandra Wood, Miss. Lena Wang, Miss Xiaowen Hu, Miss. Hong-Anh Nguyen, Miss. Lucy Elder, Mr. Matt Chapman, Miss Mei-Hsien Du, Mr. Xuelei Wang, Mr. Lim Gim Tiong, Miss. Janniefer Chan, Miss Jingjing Xie, Miss Shiyu Xie, and Mr. Farhad Bazyari.

Finally, yet importantly, I am forever indebted to my beloved parents and husband Mian Qin for their understanding, endless love, patience and support.

Paper Arising from This Thesis

Shu Wang, Hua-liang Wei, Daniel Coca, Stephen A Billings, “Model term selection for spatio-temporal system identification using mutual information’’, International Journal of Systems Science, Received 17 June 2010; final version received 9 May 2011; available online 02 Aug 2011.

Daniel Coca, **Shu Wang**, Stephen A Billings, “Identification of statistical cellular automata with probabilistic transition rules’’, submitted to On Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE.

Contents

List of Figures

Lists of Tables

Chapter 1

Introduction

1.1 Background and Motivation

Bacteria have exuberant vitality so that they exist everywhere on the planet. There are more and more studies in bacteria, because they have vast influences on human beings. For example, bacteria are responsible for plenty of diseases and also play an important role on the recycling processes in the life food chain. In this thesis, we are interested in slime moulds, whose name comes from their appearances as gelatinous "slime". They feed on microorganisms that live in any type of dead plant materials, so that they can usually be found on logs, fruits and in the soil. The research on slime moulds is essential. Take one common type of slime moulds- *Dictyostelium discoideum* for example. Many complicated experiments on human diseases are based on research of the cell behaviours of *D. discoideum* due to its analogous gene structures compared to humans. Individual cell behaviours such as cytokinesis, chemotaxis and phagocytosis, account for many phases of health and disease. Cytokinesis can be related to research on human immune response and tissue maintenance, especially to cancer research in the phase of cell proliferation; Chemotaxis is used to study inflammation and is usually associated with arthritis, asthma, and lymphocyte trafficking; Phagocytosis is commonly applied to immune surveillance and antigen presentation. Therefore, these organisms have been used to test anti-cancer drugs, immune-cell diseases, and bacterial intracellular pathogenesis.

In order to better understand the mechanisms of bacteria and slime moulds, there are three fundamental questions that need to be answered: What are the vital signs of slime moulds and what kind of patterns can slime moulds create? Why do these patterns form in the life cycle of slime moulds or what is the reason for cell movements? How can we deal with the problem of pattern formation and how can mathematical methods be used to decipher the mystery of slime moulds.

Dictyostelium discoideum is a unicellular amoebae, which has attracted much attention as a model system. The life cycle of *Dictyostelium discoideum* is shown in Fig.1.1. At the initial phase with adequate nutrients, cells of *Dictyostelium* grow and divide individually. The more active stage of the life cycle can be caused by the starvation or lack of nutrients. Chemotactic cell movement is a key mechanism at this aggregation phase. cAMP (3'-5'-cyclic Adenosine Monophosphate) is a signal chemical as a second messenger in cell movement. By periodically producing and relaying cAMP signals, cells can form spiral waves and stream structures as shown in Fig.1.2. The cells then head towards the aggregation centre and pile up to form a three dimensional mound. In the late aggregation, the cells start to differentiate into prestalk and prespore cells.

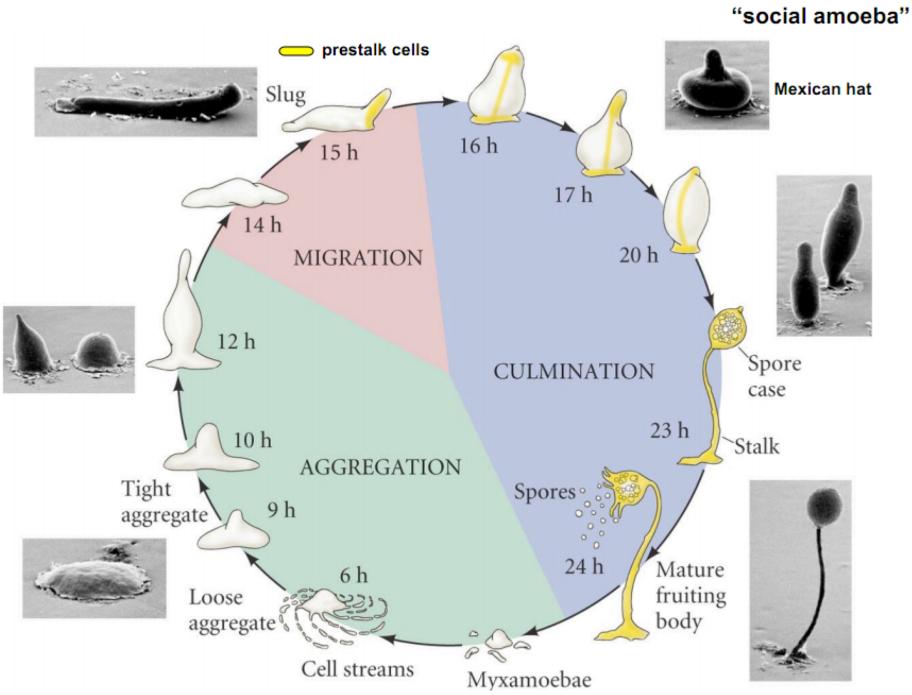


Figure 1.1: Life cycle of *Dictyostelium discoideum* [1-3]

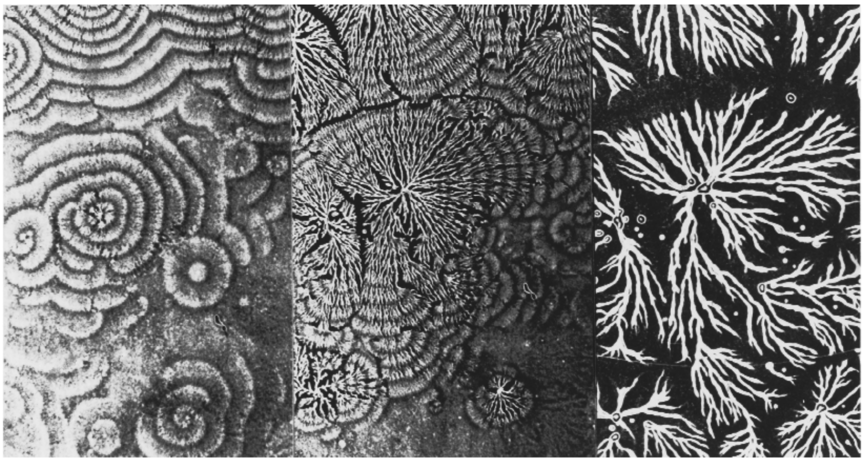


Figure 1.2: Biological patterns in the aggregation of *Dictyostelium* cells [4]. Left: concentric spiral patterns; Middle and Right: cell streaming patterns.

Prestalk cells form the tip on the top of the mound and act as a signalling source to direct migration and movement during slug formation. Prespore cells move to posterior ends, following the movement of signalling tips. The slug moves towards attractants such as light, heat and humidity. When the slug has reached a suitable environment, *Dictyostelium* will enter the final transformation of the developmental cycle, which is called culmination. The tip and end of the slug will form the stalk and spores of the fruiting body respectively [5].

This thesis will focus on the aggregation phase of slime moulds. The pattern formation in this stage can be regarded as several dynamic systems with both time and space information. Hence, a spatio-temporal system is a good option to describe the dynamics of slime moulds. Spatio-temporal systems can represent a complex class of dynamic systems, which contain both time and space. If spatio-temporal systems are analysed using a uniform lattice, the inputs and outputs can be represented by a set of states or values of all the cells in the lattice. The output or the state of each cell location in a spatio-temporal system not only depends on values of the inputs and the outputs in past time, but also have a strong relationship with the states at other spatial sites or cells within a neighbourhood.

It is true that many classical control problems are expressed by temporal systems, where the current output is only related to values of input and output at previous time. However, most observed phenomena in nature are much more complex. Spatio-temporal behaviours, which are difficult to be described by temporal systems alone are usually found in phenomena such as the weather, biological pattern formation, and some chemical reactions. In such systems, space and time cannot be separated. The formation of sea shell patterns is another typical example from biology. Different shells have individual features, so patterns become the criterion of differentiating shell types. Most shells produce their surface pattern in a one-dimensional way as time progresses. They grow with pigments by continual accretion of calcified material onto the margin, the growing edge, and follow some specific ‘rules’ to form regular strips and dots across the surface. The ‘rules’ record how the current output is affected by the pigmentation distribution of other sites at past times. There is another typical example in the field of chemistry, namely the *Belousov-Zhabotinsky* (BZ) reaction. The BZ reaction was first discovered by Russian Boris Belousov in 1951. Basically the BZ reaction [6] is an oscillating process. In some widely studied experiments, ferroin was put in the oxidation of citric acid by bromate. The oscillation is manifested by the colour changes as the ferroin changes from Fe^{+2} to Fe^{+3} and the colour changes from brick red to bright blue. The dramatic oscillatory and wavelike properties form spatio-temporal self-organisation patterns. These reactions can be described as reaction-diffusion systems. The advantage of reaction-diffusion systems is that the spatio-temporal pattern formation can often be described using simple equations relating the dynamics of the local concentrations of chemicals. Spatio-temporal systems analysis is more complex than for purely temporal systems, because both time and space domains have to be considered at the same time, rather than only time as in temporal systems. Hence, as an extension of temporal systems, spatio-temporal systems can be applied to more complicated natural processes which exist in most disciplines as diverse as physics, biology, chemistry, ecology, engineering, and even social sciences.

Modelling the pattern formation in spatio-temporal systems has received some attention recently. Basically, the natural language of patterns is mathematics, which enables us to understand and unravel the essence of pattern formation. Mathematics is also an efficient tool to describe the pattern formation process at the most fundamental level in order to determine how features can be reproduced. It has been found that some complex phenomena can be represented by simple mathematical forms or descriptions. Take the growth of a mollusc shell as an example, where a graceful spiralling pattern is formed as it grows. If this process is considered from an evolution or natural selection point of view, or how the surface cells get together to form such complex patterns, the problem will be very complicated. However, once it is recognised that the pattern on the shell follows a precise mathematical form leading to spiral growth patterns, the growth law seems to be nothing more than a simple and plausible mathematical description. The model is known as ‘a small-scale replica of the real thing’, but actually there is still a large gap between known mathematical pattern models and phenomena in the real world. Scientists always try to look for an exhaustive explanation for what has been observed, but most results are not as good as expected. That is because sometimes when some influences of little significance are added to the system or current equations, they will make the solution harder to obtain without altering the solutions. Also for some important factors, it is difficult to include them in the equations or to find a way to solve these equations. Hence, a good approximation which is easy to solve, is often a good compromise for modelling. In other words, models are scientific descriptions of phenomena, but they may not always fully capture reality perfectly, so that there is not a single or unique model that is completely ‘right’. Models are differed by choices of what to put in and leave out. Thus, some phenomena may be presented successfully by more than one model.

Pattern evolution is viewed as a forward or direct problem in spatio-temporal system analysis. In this process, the output is recorded by putting the input into a pre-known system whose model has already been confirmed. The model is used to simulate a real phenomenon to produce the output behaviours.

The forward problem has been well-studied, but system identification, the reverse problem of pattern formation is still an open problem. System identification plays a vital role in system analysis, control and prediction. The aim of system identification is to construct a model from the experimental data from a given system of interest. For example, assume that a given system is represented by an equation $y = f(x)$, in which x and y are the *n*-dimensional input and output respectively. f is a functional relationship between the output y and the input $x = (x_1, x_2, \dots, x_n)^{\mathrm{T}}$. The objective of system identification is to construct an approximation for the relationship f using a set of input and output data. Once the relationship and rules have been obtained, the system can be investigated and analysed by figuring out how one thing leads to another or how the input affects the output. In addition, good predictions can be made using a good model based on the current states. Take the stock market for instance, if investors can find out the model of the stock market and know the tendency of stock prices, fortunes can be made by investing in the right stock at the right time.

However, nobody can make long-term predictions of the stock market, because so many factors with complicated relationships are involved in such a complex dynamic system. For example even a rumour may be sufficient to trigger sales, lowering quotations suddenly. This example also shows the challenge in system identification. The difficulties of the reverse problems can be summarised as follows:

a) Many reverse problems are ill-posed, because small fluctuations in the input may cause big differences in the output, or the outputs might not depend continuously on the inputs, or a solution may be nonunique, which violates Jacques Hadamard’s postulates of well-posed problems.

b) For ill-posed problems, priori information is required sometimes even the observation data are enough.

c) A large number of candidate data are usually involved in identification, so that it is obvious to have an increase in computational complexity.

1.2 Slime Mould Dynamics

1.2.1 Migration Process of Slime Mould

In the early aggregation stage, slime moulds are often studied at the cellular level. ‘Talks’ between different cells begin under the condition of suffering starvation using the language of cAMP signals. The principle of the movement is resulted from three main competencies:

a) The cAMP signal is generated periodically by the aggregation centre.
b) The cAMP signal is relayed from cell to cell around the centre.
c) Spiral waves with the rising phase are produced because of chemotaxis.

Assume there are two forms of receptors, R (unbound and active) and D (unbound and inactive), located on the outer face of the membrane. Cells become sensitive to cAMP after the starvation, which triggers the secretion of cAMP in response to extracellular cAMP stimuli. Firstly, binding processes happen between the extracellular cAMP and the two outer receptors in different reactions, which is shown in Fig.1.3(a). The active complex (RP) from the covalent modification of R receptor and cAMP (P) binds to adenylate cyclase, which results in the activation of the enzyme. cAMP is produced by the enzyme from intracellular ATP and secreted to outside of the membrane, which can bind to outer receptors again. Therefore, this process in Fig.1.3(b) forms a positive feedback loop of cAMP production and amplification. At the same time, the amount of cAMP decreases continuously by intracellular and extracellular phosphodiesterase. When the extracellular cAMP reaches some high concentration, cells start to enter a period of adaptation or desensitisation, which is a warning response to the persistent stimulation and causes adenylate cyclase no longer to be activated and then the extracellular cAMP levels are going to fall. In this way, the system is back to basal states and waiting for being activated again, which results in a cAMP oscillation. More details can be found in [7], [8], [9], [10] and [11].

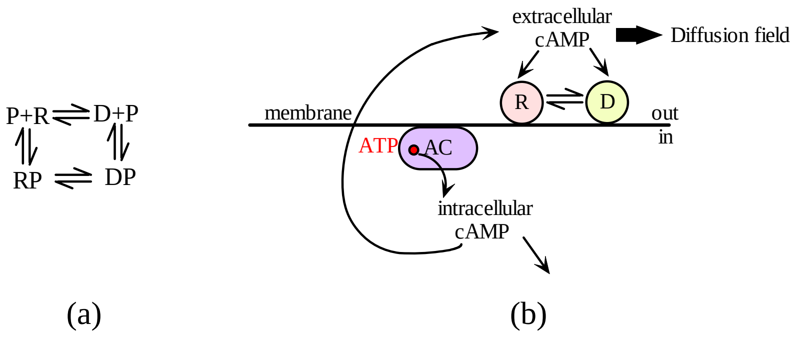


Figure 1.3: Model description of the production of cAMP. (a) are reactions between outer receptors and extracellular cAMP (P), where R is the unbound and active receptor which binds to the extracellular cAMP. D is the receptor cannot couple to the adenylate cyclase and represents an inactive and desensitised state. (b) shows the process of the cAMP production.

The secreted cAMP of the aggregation centre diffuses and reaches to its neighbours. Cells in the neighbourhood are then activated in turn and begin to produce cAMP and stimulate their neighbours afterwards. Thus, cAMP waves can spread throughout all the cells, and finally form spiral waves because of the adaptation process which keeps the cAMP waves travelling unidirectionally. It is because those cells become refractory as soon as finishing the signal relay during desensitisation, that cannot be activated again until detecting the new stimulation [5].

cAMP makes cells become more and more excitable, and cAMP concentration conducts cells to move. All the cells show chemotaxis activities in which cells move towards the area with higher cAMP concentrations, while when cells detect the decrease of cAMP concentrations, they will slow down and then stop [12]. This chemotactic reaction, navigated by outward propagating cAMP waves, leads to the formation of spiral patterns by the cell periodically moving inwards the aggregation centre. In the spiral density waves of Fig.1.2, the light band shows excitable cells while the darker band represents desensitised cells with no movements. Cell density waves also reflect periodic travelling cAMP signals indirectly [13].

Cells do not simply move straight to the aggregation centre, but are divided into domains where cells joined in branching streams moving to their centre, the sources of cAMP waves. Stream formation has been shown to result from the instability in the cAMP wavefront, which is caused by high cell density. Guided by the spiral cAMP signals, cells migrate towards the high density area, which hence raises the cAMP density and causes a deformation of the cAMP wave fronts. As an autocatalytic process, streaming dynamics will spread all over the population [14].

1.2.2 Maze Solving Problems in Slime Mould

Intelligence and information processing are not exclusive to human and animals, but have also been observed in plants and microorganism. It is interesting to find several smart and intelligent behaviours of microorganisms for the sake of survival. Fungi in [15] show the excellent capacity of solving geometrical maze problems. The *basidiomycetous fungi* modulate their behaviours according to the geometrical change of the environment such as being stopped by physical obstructions. A long-range directional memory has also been shown in the route searching, which makes fungi find nutrients efficiently. There is another example of the amoeba-like slime mould *Physarum polycephalum*. In Nakagaki's experiments [16], food sources were placed at the two exits of the maze and the slime mould can solve the labyrinth and connect the two food points by the shortest route. Similar researches have also been shown in [17], [18], [19] and [20]. These living organisms show surprisingly 'smart' behaviours in most complicated situations, which have triggered much attention of biologists, mathematicians, physicists and engineers. The 'smartness' of microorganism has been applied to other disciplines such as transport network design [21], robot control [22] and computer science [23].

1.2.3 The Simulation of Slime Mould

Due to the complicated intracellular biochemical reactions, the aggregation of slime moulds is more complex rather than a simple pattern formation. There is an unknown law controlling or conducting the movement of amoebae. How individual cells react upon one another; how active amoebae decide where to go and what the underlying law causing the formation of special patterns is, are still open questions. The theoretical study in the aggregative pattern formation suggests a better understanding of the cell-to-cell communication and provides a clear way to explore the directional cell movement.

Keller and Segel [24] first proposed a cell density reaction-diffusion model based on the idea of the Turing pattern formation [25] to describe the aggregation of *Dictyostelium* as an instability. After then more and more researches have focused on the aggregation of slime moulds. There are two main basic methods to model slime mould aggregation. Modelling amoebae as discrete cells is one approach. For example, the chemotactic response of the cellular slime mould were simulated in one spatial dimension by Parnas & Segel [26]; Two dimensional simulation of *Dictyostelium Discoideum* was shown in Mackay's study [27], which produced many primary observed patterns in the aggregation such as rotating spiral waves and branching stream patterns; The simulation of streaming formation using a discrete cell model is shown in [28]. The other modelling approach for aggregation is to use continuous models which describe the density of amoebae as a continuous variable. The modelling for the streaming stage discussed in [29-31] are examples of this continuous approach.

During the later phase of the aggregation cells begin to heap up to form a mound structure and sort into two types, prestalk and prespore cells. Prestalk cells then form the tip on the top of the mound and prepore cells go to the bottom. When the tip elongates enough, it falls down and a slug is formed. The slug, guided by the front tip, migrates away and all the cells move together all the time. Thus slime mould is entering the slug stage. Previous studies of the cell sorting can be found in [32-34]. Chemotaxis and the combination of excitation waves have also been found during this slug stage. Based on the model formulated by Savill and Hogeweg [14], Marée and Panfilov proposed a model using hybrid cellular automata and partial differential equations to describe the thermotactic motion of the *Dictyostelium discoideum* slug [35]. They also modelled the phototaxis of slime mould as an extension of the thermotaxis model [36]. Vasiev and Weijer presented a hydrodynamic model to describe the cell flow which is directed by propagating waves of the chemoattractant cAMP, so that experimentally observed movement behaviours of the *Dictyostelium discoideum* slug can be explained [37]. In addition, a three-dimensional model based on the hybrid cellular automata and partial differential equations was first presented to describe the whole process from single cells to crawling slug at one time [14].

1.3 Objectives

The main objective of this thesis is to model the aggregation and migration processes of slime mould, which can be split into two basic problems: the simulation and the identification. The simulation, defined as a forward problem, focuses on the investigation of characteristic properties and special patterns shown in the life cycle of slime mould through various statistical and analytical approaches. The simulation is to simulate the dynamic process or to evolve patterns which are similar to real biological features of slime mould. There have been many simulation methods in previous studies as mentioned in Section 1.2.3, for example using Cellular Automata (CA) models and Coupled Map Lattices (CML) models. However, the reverse problem of modelling slime mould has received little attention. The main aim of the reverse problems is to determine a particular mathematical model which can explain the movement of slime mould and can satisfy general sets of per-specified and special constraints or properties in slime mould dynamics. One of the most important problems in the reverse problem is the identification of the mathematical model or a set of equations from a given set of spatio-temporal patterns produced by the observed spatio-temporal systems.

In most cases the dynamics of slime mould belong to spatio-temporal behaviours. It is because slime mould aggregates and migrates when suffering the shortage of food, which causes the movement of slime mould and produces various patterns changing over time. Hence, such dynamic biological process can be described as a spatio-temporal system which is one of most important parts studied in this thesis. Among many spatio-temporal models, cellular automata (CA), coupled map lattices (CML) models, and partial differential equations (PDE) are the most often used in approximating the dynamical behaviours of slime mould. CA is one of the simplest spatio-temporal models, but can generate highly complicated spatio-temporal patterns from a very simple rule structure. Examples using CA to model the amoebae movement can be found in [38-40]. Using this cellular automata approach, biological patterns that resemble experimental observations can be produced but with the limited insight because of the lack of the consideration of the forces involved in cell–cell interactions and the cell–cell signalling. Yet another approach is to use continuous spatio-temporal models such as CML and PDE models in [41-43]. In the Palssona and Othmer model [41], cells can generate active forces, that interact via surface molecules, and can detect and respond to cAMP chemotactically. cAMP dynamics were also coupled with the movement of amoebae in the Nanjundiah model [43]. To better understand the dynamics of slime mould, the studies on spatio-temporal systems and how to apply these systems in modelling slime mould have carried out in this thesis.

The identification is a core and still an open problem in modelling. The aim of identification in this thesis is to build up a mathematical expression to explain or reproduce the dynamic process of slime mould, which makes the manual intervention and the monitoring of such process possible. Besides, the ‘philosophy’ of slime mould can also be a good reference for the studies on similar processes with some same properties such as the infection, tumour growth and organ development.

Model selection or model term selection is a fundamental task in identification, which is to select important model terms from a set of potential choices based on the training data in an attempt to obtain a best approximation model with the best inductive bias. Thus determining model terms is often linked directly to the model prediction, so the model term selection is one of most essential steps in the identification of spatio-temporal systems. Once the set of possible model terms has been selected, the model can be determined easily by mathematical approaches. Obviously selecting the wrong model terms into the model may cause big bias prediction results. Hence, a right set of model terms is the premise of optimal identified models, which can considerably reduce the complexity of the parameter estimation.

In order to understand the Reaction-Diffusion-Chemotaxis aggregation scheme of slime mould, research has been carried on the Nazim Fatès model [44]. This model is efficient to exhibit self-organisation properties of slime mould using cellular automata. Though this model has simple descriptions and better simulation results, the identification of the model becomes a challenge because of some random processes involved in the model. For a typical cellular automata rule, the evolution is based on a specific rule all the time, which means it is a predictable and known process once the CA rule has been known. However, the random process in the Nazim Fatès model makes the process unpredictable and it is difficult to determine one CA rule to describe this randomness. Therefore, a probabilistic multi-rule cellular automata model has been inspired from the Nazim Fatès model and introduced to describe such random CA process. The new model is efficient to reduce the randomness in the unpredictable process.

The problem of the identification of the probabilistic multi-rule CA model is then staring us in the face. Firstly, how to determine the rule for the probabilistic multi-rule CA model becomes a problem. The most commonly used method for the identification of spatio-temporal systems is known as the orthogonal least squares (OLS) algorithm or a variant of this the orthogonal forward regression (OFR) algorithm [45, 46]. However, these methods can only identify one specific rule or one equation which may be difficult to describe highly complex probabilistic multi-rule CA patterns. Hence, a new identification method for the probabilistic multi-rule CA model has been developed. The identified model can help to uncover the underlying law of the probabilistic multi-rule CA patterns. This new method can also be extended to other similar dynamic systems with random processes, which benefits for the further study of some random systems.

No real systems can be the same as an ideal theory system free from the disturbance of noise. The probabilistic multi-rule CA system is a discrete spatio-temporal system so that the noise in these systems is different from that in the continuous systems such as partial differential equations (PDEs). The variables in the probabilistic multi-rule CA system rules can only involve two states normally 0 and 1, so the noise corrupts the system by means of flipping cells from one state to another like from 0 to 1 or from 1 to 0. Maybe using the traditional signal-to-noise ratio to measure the noise influence in these systems is not a good idea, since the magnitude of the noise is the same as the original signal so that the ratio is essentially 100%. Therefore, the measurement and effect of the noise in probabilistic multi-rule CA systems is worth studying, and the identification methods for noise-free models should be applicable to the noisy cases.

1.4 Contents of This Thesis

This thesis consists of eight chapters. Chapter 1 answers the question of what the project is and why this project is worthy of study. Chapter 2 to Chapter 7 are dedicated to how to solve important problems related to the slime mould. Chapter 8 briefly summarises the work has been done in this thesis and also gives some discussion about further studies.

Chapter 2 briefly reviews spatio-temporal systems which are the main tool used in slime mould studies in this thesis. This chapter begins with a concise introduction to spatio-temporal systems and the connection with slime mould modelling. Three general classes of spatio-temporal systems are then introduced and models related to slime mould in previous studies are reviewed. The main identification methods for each spatio-temporal system are listed afterwards.

In Chapter 3, three different spatio-temporal models are employed to simulate the patterns produced by slime mould at the first aggregation stage. The Greenberg-Hasting Model (GHM) [47] is a CA model commonly used for excitable media, and here it can be applied to model slime mould because slime mould has similar features with excitable media. Followed by Solé's model [48] and Kawasaki's model [49], more simulations using continuous models are introduced. The simulation results demonstrate that patterns such as spirals and concentric circles which are observed in the slime mould aggregation can be generated by these models suggesting that modelling can be used as a basis to capture and analyse the properties of complex slime mould behaviours.

Chapter 4 concentrates on the reverse problem-the identification based on some of the simulation results from Chapter 3. A new identification method, OFR-MI (Orthogonal Forward Regression using Mutual Information) algorithm, is derived. Previous studies on the model term selection using mutual information are reviewed and mutual information is used for first time as the criterion for selecting essential terms of spatio-temporal systems. The new algorithm is then tested on several benchmark spatio-temporal models including Cellular Automata (CA), Coupled Map Lattice (CML) and Partial Differential Equations (PDE) models.

Chapter 5 focuses on the study of a Reaction-Diffusion-Chemotaxis model originally proposed by Nazim Fatès. This model can obtain good simulation results of cell movements in the slime mould aggregation. The chapter starts with a brief review of Reaction-Diffusion-Chemotaxis models in previous studies and an introduction to the Nazim Fatès model. Next, a probabilistic CA model is formulated by simplifying the original Nazim Fatès model. Simulation studies are carried out for the original model and simplified models respectively to derive a representative model formulation for slime mould. Extracted from the Nazim Fatès model, a new type of spatio-temporal model, the probabilistic multi-rule CA model, is proposed with a specific definition and simulation examples.

In Chapter 6, as a new CA model, the probabilistic multi-rule CA model is differentiated from other spatio-temporal systems by having more than one rule that operates in a model. The identification problems of these new models are discussed and a new algorithm for the identification of probabilistic multi-rule CA models is introduced. One-dimensional and two dimensional model examples are tested using the new identification method.

Chapter 7 analyses two types of noise in probabilistic multi-rule CA models. They are static and dynamic noise. The measurement of these two categories of perturbations and their effects on probabilistic multi-rule CA models are explained first. The corresponding simulation and identification are studied. Finally, the new identification method for the probabilistic multi-rule CA models proposed in Chapter 6 is modified and enhanced so that it can be applied to noisy patterns.

The conclusion Chapter 8 is a summary of the main work and contributions in this thesis, followed by the discussion and suggestion for further studies.

Chapter 2

Overview of Modelling Spatio-temporal Systems

2.1 Introduction

A spatio-temporal system is a dynamical system that can model the behaviours of complex non-linear systems. Spatial-temporal models arise from the spatio-temporal phenomena where data are collected across time and space. Spatio-temporal phenomena where the dynamics at any point depends upon the time evolution and the dynamics at other spatial locations are ubiquitous. Hence, spatio-temporal systems can represent an enormous class of highly complex dynamical systems, and spatio-temporal modelling then becomes a crucial research area related to the statistical analysis of data arising from wide and various applications in nature such as ecology, biology, geology, epidemiology, and environmental health.

The simulation of spatio-temporal systems produces patterns generated from complex interacting behaviours in real life. Spatial interactions and temporal dynamics need to be considered at the same time, so these systems are more difficult than purely temporal systems or spatial systems which are only time dependent or space dependent respectively.

The identification of spatio-temporal systems, the reverse problem of simulation, has received considerably less attention up to now. However, it is essential to research on this problem because the identification is to explore the causes of the phenomenon and to obtain or compute a mathematical model from the data in real evolution cases or from simulation data. Once the model is known, it is easy to control or add manual intervention into the real systems by adjusting corresponding parameters or variables in the model according to our expectations, thereby benefitting human beings. For example, infection can be regarded as a spatio-temporal dynamic process. If the reproduction and movement of microparasites like viruses and bacteria can be formulised by mathematical models, the reasons of the infection can be clarified and effective measures can be taken to deal with or control the infection. In this way, a big progress may be made on many difficult miscellaneous diseases such as cancers and the HIV infection.

Owing to the characteristics of spatio-temporal systems detailed above, the slime mould behaviours can also be viewed as spatio-temporal systems. Slime mould seems to act with ‘a brain’, which is intelligently sensitive to an external stimulus. According to the movement and aggregation of cells, slime mould always can find the better environment for survival. The spatio-temporal behaviours of slime mould are the key problem discussed in this thesis and the spatio-temporal model is a good and effective tool to study this problem, which is reviewed and explained in detail in this chapter.

2.2 Spatio-temporal Systems

Three main elements in the spatio-temporal system are space, time and cell states. Generally, when classifying spatio-temporal systems by the state of individual cells, there are two main types, discrete models and continuous models. The discrete case is known as Cellular Automata or simply CA. These type of systems have finite values at each site and the rules are usually represented by the combination of different Boolean rules such as *and*, *or*, *not* etc. The class of systems that have continuous state values at each site can be described by Partial Differential Equations (PDE), or when based on a discrete lattice space as Lattice Dynamical System (LSD) or Coupled Map Lattices (CML). However, LSD, CML and CA models have the property of discrete time and space, but PDE models not only have a continuous state, but also continuous time and space.

These models mentioned above will be described in detail in the following sections.

2.2.1 Cellular Automata

Cellular Automata (CA) were conceived by von Neumann in the early 1950’s [50]. CA are dynamical systems in which space and time are both discrete. Each cell which is arranged in the form of a regular lattice structure has a finite number of states. All the states in the cells are updated synchronously by a specific transition rule based on the information of the individual states and of cells in a neighbourhood at past times.

An n-dimensional cellular automata is defined on a lattice structure. For a two-dimensional lattice, choices of a square lattice, a triangular lattice and a hexagonal lattice [51] are the common types. The selection of lattice type varies according to different investigation objectives and different lattice types lead to different CA rules in the evolution process. The typical lattice type which is widely used is a square lattice. This lattice can be represented as $\mathcal{X}^{(d)}$, where $d = 2r + 1$, and r is a finite integer which determines the size of the neighbourhood. $\mathcal{X}$ is a finite set of states of all cells in the lattice.

The neighbourhood is a cell set capable of directly influencing the evolution of the outputs. The cells involved in the neighbourhood can be from different spatial and temporal scales. For example the one-dimensional 3-site neighbourhoods in Fig.2.1(a), (b) and (c) are the cases that the neighbourhood of a cell at the position j and the time instant $t + 1$, denoted as $c(j, t + 1)$, are from the same time scale t but with different space scales. Fig.2.1 (d) is an example of neighbour cells having two cells from different temporal scales at time step $t - 1$, $c(j - 1, t - 1)$ and $c(j + 1, t - 1)$, as well as two other cells from different spatial scales of time step t, $c(j - 1, t)$ and $c(j + 1, t)$.

j - 1, t	*j, t*	*j + 1, t*

(a)

j, t	*j+1, t*	*j+2, t*

(b)

j - 2, t	*j-1, t*	*j, t*

(c)

j - 1, t	*j, t*	*j + 1, t*
j - 1, t-1		*j + 1, t-1*

(d)

Figure 2.1: Examples of neighbourhoods for CA from different spatial and temporal scales. (a) von Neumann neighbourhood (b) right-shift neighbourhood (c) left-shift neighbourhood (d) the neighbourhood that involves cells both from different temporal and spatial scales.

The dynamic of CA is then described by a neighbourhood function $f\colon \mathcal{X}^d \to \mathcal{X}$. For example, if $d = 5$, the neighbourhood in this case can be defined as the other four cells around the original one. Thus, the output of each cell is produced by following the rule f. The transition function f shows the interaction of cells which can be listed in a finite look-up table. A time shift operator then upgrades all the cells synchronously at each discrete time step when time passes.

The transition rule f can be viewed as a logical function or Boolean function of d variables, because the transition function is equal to the transition table of length 2^d. For example, a 3-site CA rule with the neighbourhood set of $\{c(j-1,t),\ c(j,t),\ c(j+1,t)\}$ described by the Boolean equivalent is

$$c(j,t+1) = c(j,t) \vee (c(j-1,t) \wedge c(j+1,t)) \tag{2.1}$$

where '∨' denotes the *OR* operation and '∧' denotes the *AND* operation. It has been shown in [52] that CA rules can be expressed in a polynomial form for the model in Eqn.(2.1), gives

$$\begin{aligned} c(j,t+1) = {} & -2.0c(j-1,t)c(j,t) - 2.0c(j,t)c(j+1,t) + c(j,t) \\ & -2.0c(j-1,t)c(j+1,t) + c(j-1,t) \\ & +3.0c(j-1,t)c(j,t)c(j+1,t) + c(j+1,t) \end{aligned} \tag{2.2}$$

CA is one of the simplest spatio-temporal systems, but it can produce complex patterns even through a simple rule. It has been found that CA can be used to approximate some natural systems and give the general behaviours of these systems. Hence, there have been numerous applications of CA models in various areas including biology [53], chemistry [54], sociology [55], image processing [56], etc.

2.2.2 Coupled Map Lattices

Although CA have been widely applied in many disciplines, there are still some limitations when they are used in some continuous spatio-temporal systems. For example, when a continuous system is modelled using a CA model, the continuous states of the cells need to be discretised into discrete values. In this way, it is obvious that information, sometimes even important and useful data, can be lost. However, most of the problems mentioned above could be solved, using Coupled Map Lattice (CML) models which were introduced in a model simulation by K. Kaneko in the 1980s [57]. The model Kaneko used consists of a continuous sequence of logistic maps coupled to their neighbours and with parameters chosen to produce a chaotic behaviour. The system of coupled mappings was regarded as a CML model. The CML model is a typical model of extended dynamical systems with discrete time and space, but with continuous state variables. The CML model lies somewhere between CA and PDE models. CML models are convenient for computer simulations especially in theoretical physical analysis, for the interpretation of experimental results, and for mathematical analysis. Therefore, CML models have been applied in many fields including chemistry [58], biology [59], ecology [60], physics [61, 62] and so on.

There are four key parts in CML models, that is, a lattice structure, the neighbourhood, the lattice states and a dynamical process. The process of a CML model can be defined by the following steps [63]:

a) A lattice architecture

Let c be a cell in a lattice $\mathcal{X}$, so that $c \in \mathcal{X}$. The neighbourhood can be described as

$$Nhd(c) = \{x, y_c^{\varphi(c)}\} \tag{2.3}$$

where $y_c^{\varphi(c)}$ represents neighbouring cells corresponding to the cell c. $\varphi(c)$ represents the selection of neighbourhood cells and also specifies the size of the neighbourhood.

b) Lattice state description

The mapping for the state of the lattice is $\sigma: \mathcal{X} \to \mathcal{A}$, where $\mathcal{A}$ describes the states in the lattice. Thus, the state of a cell c in a lattice can be described as $\sigma(c)$.

c) Dynamic process

There are two basic processes involved: isolated local processes and interaction processes. For an isolated local mapping $f_c: \mathcal{A} \longrightarrow \mathcal{M}$, $\mathcal{M}$ is the set of all possible values of cells in a lattice, and $f_c(a)$ shows the output value at point c when the input value is a. Unlike the isolated process, the interaction process couples the states generated from the cells in the neighbourhood. This is represented by $g_c: \mathcal{M}^{\varphi(c)} \longrightarrow \mathcal{A}$. A global description of such a dynamic process was expressed by [63]

$$V_c: \mathcal{T} \times [\mathcal{X} \to \mathcal{A}] \to \mathcal{A} \tag{2.4}$$

where $\mathcal{T}$ is a time delay matrix. $V_c(0, \sigma) = \sigma(c)$ when $t = 0$, so for $t > 0$,

$$V_c(t+1, \sigma) = g_c\left(f_c\big(V_c(0, \sigma)\big), f_{y_{c,1}}\left(V_{y_{c,1}}(t, \sigma)\right), \dots, f_{y_{c,\varphi(c)-1}}\left(V_{y_{c,\varphi(c)-1}}(t, \sigma)\right)\right) \tag{2.5}$$

CML models are more complicated than CA models. They have continuous state variables instead of discrete states, so that the capability of local information production is enlarged. However, being discrete in time and space, they are simpler than Partial Differential Equations (PDE) which require a large amount of information commonly described by a continuous function. Thus, for PDE, larger computational resources are involved. When the characteristics of these different models are well understood, it is very helpful for numerical analysis both in mathematical and physical aspects.

2.2.3 Partial Differential Equations

PDE models represent one of the most important parts of mathematical analysis. PDE models were originally developed in the areas of physics and mechanics, because some distinct physical phenomena such as the propagation of sound, fluid flow and electrodynamics seem to have identical mathematical formulations with PDEs. Due to more and more attention on these models, PDEs have already been extended into other fields such as biology, finance, computer science (especially in image analysis) and so on.

PDEs demonstrate a relation between an unknown function with several independent variables and its partial derivatives. The general form of a PDE based on a function $u(x_1, x_2, \dots, x_n)$ is

$$F\left(x_1, x_2, \dots, x_n, u, u_{x_1}, u_{x_2}, \dots, u_{x_{11}}, \dots\right) = 0 \tag{2.6}$$

where $x_1, x_2, \dots, x_n$ are the independent variables, u is the unknown function, and u_{x_i} represent the partial derivative $\frac{\partial u}{\partial x_i}$. Generally, additional conditions such as initial conditions and boundary conditions are included.

PDE models are not only continuous in the time and space domains, but also the state of each point. With these continuous properties, PDE models provide an effective tool to understand or reconstruct continuous spatio-temporal systems in the real world. It is because such models may easily be related to the original system parameters and there is a potential to provide a clear physical explanation. For example, a reaction-diffusion process can be represented by the following PDEs as

$$\frac{\partial y(x,t)}{\partial t} = D\frac{\partial^2 y(x,t)}{\partial x^2} + R(y) \tag{2.7}$$

$y(x,t)$ represents the concentration of one chemical, when it is applied in an oscillatory chemical reaction. D is a diagonal matrix of diffusion coefficients and $R(y)$ accounts for all local reactions.

There are many PDE models which have been applied in many different areas. For example, the Keller & Segel model [24] was introduced to model the chemotactic movements of bacteria. The FHN model [64] and the Oragonator models [65] were used in the analysis of the BZ reaction–diffusion system [6].

2.3 Identification of the Spatio-temporal Systems

The identification of the spatio-temporal system can be viewed as a process of seeking and perceiving the essence of the system behaviour through the spatio-temporal phenomena. This identification problem is of great important but still a challenging problem even today. Finding equations to describe the observed phenomena using an analytical modelling approach is not easy, because usually complex interactions are involved in real cases which are highly nonlinear.

In system identification, there are two main objectives, model structure selection and parameter estimation. The model structure selection is very important in the identification, because simply increasing the number of the system terms will lead to much more complicated models, and the more complex the model structure is, the more expensive computation it will cause, as well as more limitations in practical implementation. Once the model structure, or which terms to be selected in the model has been determined, the rest of the identification is to estimate the parameters of the model. Parameter estimation can thus be regarded as a standard minimisation type problem which can be easily solved using various well-developed numerical techniques.

There are many types of non-linear models including radial basis function networks, wavelet networks, polynomial models, neural networks, and rational models. But how to select the variables and model terms is a key focus in the model structure determination whatever type of the model is selected. The model terms and variables in non-linear systems can affect the system performance more than that in linear regression models. Take a simple non-linear system with three input variables for instance. The number of model terms will turn out to be ten in a second-degree case. Thus, the number of model terms will increase dramatically by adding more variables or increasing the system degree. Therefore, the ideal selection method should search over a significant set of variables and terms to achieve a relatively small set of terms which can construct the final prediction model with a desired accuracy. The system description and reviews of present identification methods will be discussed in detail in the sections below.

2.3.1 Model Description

In order to give an introduction of the identification algorithms, the NARMAX model, (Non-linear Auto Regressive Moving Average with eXogenous inputs) which can represent many non-linear dynamic systems [66] is defined as

$$y(t) = \mathcal{F}(y(t-1), \dots, y(t-n_y), x(t-1), \dots, x(t-n_x), e(t-1), \dots, e(t-n_e)) + e(t) \tag{2.8}$$

where $y(t)$, $x(t)$ and $e(t)$ denote output, input and noise sequences respectively. When this model is extended to the multi-input multi-output (MIMO) case with m_y variables in the system output and m_x variables in the input, the variables can be written as vectors $y(t) = \left[y_1(t)\ y_2(t) \dots\ y_{m_y}(t)\right]^T$, $x(t) = \left[x_1(t)\ x_2(t) \dots\ x_{m_x}(t)\right]^T$ and $e(t) = [e_1(t)\ e_2(t) \dots e_{m_y}(t)]^T$. n_y, n_x and n_e are the maximum time delay. $e(t)$ is a zero mean independent sequence, and $\mathcal{F}$ is some non-linear function. The objective of system identification is to find a proper approximation with respect to $\mathcal{F}$. A common choice is to describe $\mathcal{F}$ by a polynomial representation with a given degree l,

$$y(t) = \theta_0 + \sum_{i_1=1}^{n} \theta_{i_1} T_{i_1}(t) + \sum_{i_1=1}^{n} \sum_{i_1=i_2}^{n} \theta_{i_1 i_2} T_{i_1}(t) T_{i_2}(t) + \cdots$$

$$+\sum_{i_1=1}^{n}\cdots\sum_{i_l=i_l-1}^{n}\theta_{i_1\ldots i_l}T_{i_1}(t)\cdots T_{i_l}(t)+e(t) \qquad (2.9)$$

Where $n = n_y + n_x + n_e$ and $T(t)$ represents y, x or e with time lags. Eqn.(2.9) can be written as a linear-in-parameters regression model,

$$y(t)=\sum_{i=1}^{M_s}\theta_i\phi_i(t)+\xi(t), \qquad t=1,\ldots,N \qquad (2.10)$$

where N is the data length. θ_i are unknown parameters to be estimated. $\phi_i(t)$ are important model terms which are generated from the input and output data using various basis functions. M_s is the number of final model terms which is selected from a set of candidate model terms $\{\phi_i\}_{i=1}^{M}$.Generally, the set of candidate model terms is constituted by all the possible combinations of system variables up to degree l. $\xi(t)$ is the modelling error.

The linear-in-parameters model structure in Eqn.(2.10) can be used to express most parametric and non-parametric models, so that it is an effective tool to find an approximation to an unknown nonlinear function.

2.3.2 Orthogonal Least Squares

An optimised identified model should contain all the significant terms which are mentioned in Eqn.(2.10), and the redundant terms should have been removed from the model. Leaving out any significant terms may cause wrong model to be identified, meanwhile redundant terms can complicate the identification process and make the computation time consuming. Therefore, developing a good method for model structure determination becomes essential in the identification procedure. The orthogonal least squares (OLS) algorithm together with the Error Reduction Ratio (ERR) has proved to be one of the most efficient methods both for term selection and parameter estimation in nonlinear system identification [45, 46]. The OLS algorithm is based on the orthogonalisation of the regressors which are the terms in the models. This method was initially applied to single-input single-output (SISO) systems, but it has been widely extended to many MIMO systems.

The OLS algorithm, developed by extending least squares theory, is efficient in the model term selection. This algorithm includes a stepwise orthogonalisation of regressors and a forward selection of terms according to their contributions to the overall model measured by the ERR [45]. The classical OLS algorithm results in a particularly simple estimation procedure which is described by the following steps:

a) Orthogonalising all the regressors in a model so that the correlations between all the terms are removed.
b) Determining significant terms by comparing the Error Reduction Ratios or ERRs with all the candidate model terms.
c) Computing the corresponding parameters with respect to selected terms.

To further understand the OLS algorithm, a simple description for the algorithm is shown mathematically as follows. The vector format of Eqn. (2.10) is

$$Y = \Phi\Theta + \mathbf{e} \tag{2.11}$$

where Φ is the regression matrix, Θ is the parameters vector and $\mathbf{e}$ is the residual sequence. Eqn.(2.11) can be converted to another expression including orthogonal regressors using the orthogonalisation, say

$$Y = Wg + \mathbf{e} \tag{2.12}$$

where

$$\Phi = W\mathrm{A}, \mathrm{A}\Theta = g \tag{2.13}$$

and

$$\mathrm{A} = \begin{pmatrix} 1 & a_{1,2} & \cdots & a_{1,M_s} \\ 0 & 1 & \cdots & a_{2,M_s} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \tag{2.14}$$

In (2.12), W is the orthogonal regression matrix.

2.3.3 Identification of CA

As noted in Section 2.2.1, a CA system consists of three main elements: a discrete lattice, a neighbourhood and a transition rule. Similarly, the identification of CA includes the neighbourhood detection and the rule determination based on a lattice structure. Neighbourhood detection is part of the procedure of the model structure determination, and is also the crucial premise of the rule determination. A correct detected neighbourhood can provide the feasibility of producing the optimal model. For a simulation case, a candidate neighbourhood can be manually set in advance according to experience. But for a real system without any *prior* information about the model, the neighbourhood detection algorithms are necessary to be developed. After the candidate neighbourhood is obtained, CA rule can be identified using an estimator such as the orthogonal least squares based algorithm in this thesis, so that model structure is determined and coefficients are estimated.

At present, there are a few methods for identifying cellular automata rules. Fred et al. [67] employed a learning algorithm-genetic algorithm (GA) to extract CA rules directly from observed spatio-temporal patterns with no a priori knowledge about the physical system. But there was not a clear neighbourhood structure or parsimonious explanation in this method. Yang and Billings [68] introduced an improved method which can extract precise Boolean rules from CA patterns by a multi-objective genetic algorithm, but the detection process was complicated and very time-consuming. Billings and Yang [69] then proposed a Cellular Automata Orthogonal Least Squares algorithm (CA-OLS). This was the first time an adapted orthogonal least squares algorithm was used in the identification of CA. In the CA-OLS algorithm, Boolean rules were mapped into a polynomial form, and the OLS algorithm was employed, resulting in an efficient method for the neighbourhood detection, term selection and parameter estimation even for complex CA patterns. A modified algorithm of CA-OLS with fast computing speed, named as FCA-OLS algorithm, was presented by Billings and Mei in 2005

[70]. However, all these algorithms need an initial candidate neighbourhood which contains, as a subset, all the correct neighbourhood cells. Candidate sets with insufficient cells or wrong cell members will cause an infeasible identification or an incorrect model. As part of the selection the ERR cut-off value also affects the results of the identification, because when the system is corrupted by noise, a slightly larger cut-off value can lead to a number of redundant terms in the model and contrarily a smaller cut-off value can cause the elimination of some vital terms [52]. In order to optimise CA-OLS, Zhao and Billings [71] introduced a new algorithm of neighbourhood detection, which provides an indication of the temporal and spatial range of the initial candidate terms based on mutual information (MI). Such a coarse-to-fine identification approach, including no a priori knowledge, significantly reduces the initial neighbourhood choices, so that the CA-OLS algorithm becomes more efficient when applied in system identification. In 2008, Guo, Billings, et al extended the CA identification method from binary cellular automata to n-state systems [72].

2.3.4 Identification of CML

The basic idea of the identification of CML models is to find the model function with the knowledge of experimental input and output data. There are two commonly used approaches for the identification of CML models. One is to identify a global model constructed by local models based on the embedding theorem [73]. In the methods proposed by Parlitz and Merkwirth [74], as well as Mandelj, et al [75], models were predicted according to the values in a rectangular or triangular neighbourhood region. These results, however, are only suitable for some simple CML systems. The other approach is based on the NARMAX model which was described in Section 2.3.1. This method has been widely and successfully used to identify the CML model for a wide range of nonlinear dynamical systems. Methods which were introduced by Coca and Billings [76, 77], Guo and Billings [78] were based on a modified OLS algorithm to achieve a powerful method for model construction. But some prior information such as empirical knowledge should be gained to determine a finite range of initialisation in the time and space domains. In addition, Pan and Billings [79] proposed a neighbourhood detection approach for CML model identification, which was based on embedding theory. The main idea of this method was to test the continuity property of the independent model function in order to obtain a reasonable neighbourhood size. Another improved method of neighbourhood detection was introduced by Guo, Mei et al. [80], which applied CA neighbourhood detection methods to preselect the initial neighbourhood before using CML identification methods to produce the final model. This coarse-to-fine strategy results in less expensive computations.

2.3.5 Identification of PDEs

Due to the continuous property of PDEs, these models may easily be related to the original variables in many real systems and can provide a mathematical expression with a clear physical explanation. Therefore, most physical, chemical processes and biology systems, such as thermal processes, and diffusion-reaction processes can be more easily described by a set of partial differential equations. However, the identification of these systems becomes a challenge because of their complex spatio-temporal nature. CML can provide a finite-dimensional model to approximate an infinite-dimensional system which is generally represented by partial differential equations, so the identification methods applied on CML models may also fit PDEs models through the discretisation step.

To date the identification of PDEs models has been attracted many attentions. Voss et al [81] developed an identification algorithm for continuous PDE systems, in which the alternating conditional expectation algorithm (ACE) is used to solve an optimisation problem involved in the identification. This method is less efficient for high dimensional cases than for low dimensional ones. The algorithms introduced by Porcu, et al [82] and Xu et al [83] are two identification methods based on a statistical approach. Another method based on artificial neural network architectures was proposed by Gonzalez-Garcia et al in 1998 [84].

One of the most widely used methods is the least squares method, which has been studied by many authors such as Coca and Billings [85, 86], Qi, Zhang, and Li [87], Guo and Billings [88, 89], and so on.

In Qi, Zhang and Li’s study [87], the spatio-temporal modelling was reduced to a traditional temporal modelling problem using the Galerkin method, and both singular value decomposition and least squares estimation were applied in the estimation of the unknown parameters. Guo and Billings [89] proposed a method based on the orthogonal least-squares algorithm, in which the parameters were estimated consistently. This iterative algorithm based method can be implemented online easily.

2.4 Conclusions

Spatio-temporal systems have received more and more study recently, because most natural phenomena have spatio-temporal behaviours which can be described by spatio-temporal models. A brief review of spatio-temporal systems has been illustrated in this chapter, including the basic concepts, typical applications, forward and reverse problems of spatio-temporal systems and so on.

The simulation and identification problem are two central and key tasks, but still challenging topics in the study of spatio-temporal behaviours in real life, since there is normally no priori information about the model structure in these practical applications. The case of slime mould is not an exception. Hence, this thesis will centre on these two problems related to slime mould and besides further discussion will be explained in the following chapters.

Chapter 3

Modelling of Slime Mould Dynamics

3.1 Introduction

Slime mould dynamics have become a more and more popular research topic recently, because the dynamics of these systems exhibit unexpected intelligent behaviours which often form various interesting and beautiful patterns. For example, concentric rings or spirals can be found in the aggregation of the slime mould *Dictyostelium Discoideum* and branch structures are typical in the colonisation of *physarum plasmodium*. However, these organised patterns are not unique to slime mould, and can also be found in many other examples such as the *Belousov-Zhabotinsky* reaction (or BZ reaction) in chemistry, population dynamics in ecology and colonial phenomena found in other biological species. Research on the evolution of these similar patterns in different disciplines can promote the understanding of similar dynamical processes.

The dynamics of slime mould can be regarded as a non-linear spatio-temporal system. In this chapter, three types of spatio-temporal models, Cellular Automata (CA), Coupled Map Lattices (CML) and Partial Differential Equations (PDEs), are used to model slime mould dynamics. The aggregating slime mould behaves as an excitable medium when food is scarce, so that the Greenberg-Hasting model (GHM) [47] as a CA model, which is commonly used for excitable media modelling, is employed to describe slime mould behaviours. In addition, a CML model based on Solé's model [48] is also applied to modelling slime mould at the aggregation stage, which is able to capture typical reaction-diffusion features of slime mould. Also, Kawasaki's model [49], a PDE model, is demonstrated to describe the colonial behaviours of slime mould which is very similar to that of bacteria. The description of these three models is followed by several simulation examples.

3.2 CA models-The Greenberg-Hasting Model

Aggregating slime mould cells can be viewed as excitable media, so that they share certain characteristics with other excitable media such as chemical oscillating reactions [90], nervous and cardiac tissue [91], and intracellular calcium signalling [92]. Excitable media are commonly described by Cellular Automata (CA) and Partial Differential Equations (PDE). Therefore, A CA model - the Greenberg-Hasting model [47] is employed in this section to show by simulation that slime mould at the aggregation stage can be represented by this model class.

3.2.1 Aggregating Slime Mould as Excitable Media

Excitable media were first introduced by Wiener and Rosenblueth in the research of heart arrhythmia in 1946 [93]. Excitable media are one of the nonlinear and spatio-temporal dynamic systems, which propagate waves of excitation at a velocity which depends upon the properties of the medium. There are two main elements in excitable media, excitation and refractivity. During the excitable period, the medium begins with an excitable state as a stable equilibrium. Once the cells of the medium receive a stimulus which is above a certain fixed threshold, the cells become excited states and meanwhile generate unified signals to excite the cells in the neighbourhood so that neighbouring cells produce identical signals. Following excitation, excitable media enter the refractory period during which they cannot support wave propagation until a certain amount of time has passed. Excitation always alternates with refractivity, which forms travelling excitation waves of various geometries such as concentric circles and spiral waves.

It has been shown that the morphogenetic movement of slime mould at the aggregation stage is governed by a control system, which has similar properties with excitable media. Therefore, excitable media are capable of interpreting the investigated results obtained in experiments of the aggregation period of slime mould [94].

The slime mould aggregation is caused by chemotaxis toward higher concentrations of the chemoattractant cAMP which acts as an intracellular messenger. At the beginning of the aggregation stage, slime mould is in a stable excitable state. When suffering short of food in the environment, slime mould cells begin to secrete cAMP. This extracellular cAMP then binds to its own receptors, which causes more production of cAMP, and meanwhile this excites or stimulates other cells in the neighbourhood to secrete cAMP. Thus all the cells in the neighbourhood are then in an excited state. However, a reverse process of inhibition is also involved at the same time, which slows down the cAMP secretion with the gradual increase of the cAMP concentration and at last stops the autocatalysis. In this way, cells become refractory states. When sensing the decline of the cAMP, slime mould recovers the capability of cAMP synthetisation. The excitation-and-inhibition cyclic process results in a travelling wave which diffuses cAMP isotropically and excites more cells to produce more cAMP. The cAMP wave propagation guides the movement of amoebae and leads to the formation of ring patterns or spiral waves which are similar to these in excitable media.

Cellular Automata are one of the main approaches to model excitable media. CA exhibit an excellent ability to produce complicated patterns at low computational cost, so that the CA can provide a relatively simple model to understand excitable media. For the case of slime mould, each cell in the CA can be occupied by an individual amoeba or a small group of slime mould cells. Each cell in the CA can be in one of three states: excitable, excited and refractory. The transitions between these three states are controlled by a CA rule, which updates the state of a cell at the current time according to the states of the cells in its neighbourhood at previous time steps. A cell cannot be excited until one or more neighbouring cells are excited and the excited cells become refractory in the next time step. After a refractory period, cells go back to be excitable again. It is obvious that the CA rule is consistent with the physical explanation of the aggregating slime mould, which further explains the feasibility of the slime mould modelling using CA models.

3.2.2 The Greenberg-Hasting Rules

The Greenberg-Hastings Model (GHM), which was introduced by Greenberg and Hasting [47], is one of the most parsimonious models which has been proposed to capture reaction-diffusion phenomena. The GHM is in a CA form which was firstly used to model the neuron activities observed in a neurons network, and then it has been extended to model many other excitable media such as the *Belousov–Zhabotinsky* (BZ) reaction via generalising GHM to incorporate more detailed aspects of these excitable media [95].

The GHM normally emulates an excitable media on a lattice $\mathcal{X}^{\mathrm{d}}$. A two-dimensional square lattice ($\mathcal{X}^2$) as the most common lattice type, is employed in all examples of this chapter. Each cell in the lattice can be represented as $c \in \mathcal{X}^2$, and at each time t, the state of each cell has one of N possible values $0, \dots, N-1$, which also can be viewed as the way that each cell is painted with one of N different colours which are arranged in a cyclic 'colour wheel' and labelled $0, \dots, N-1$ [96]. The update of a cell state is like the colour change by advancing one step at each time instant only in one direction around the wheel. The evolution of the GHM is determined by a transition rule with a finite neighbourhood $\mathcal{N}_{c(i,j,t)}$ based on a discrete lattice structure. There are three main elements in the GHM rule: a number N of all possible colours, a number E of colours in excited states, and a threshold number T of cells in the neighbourhood $\mathcal{N}_{c(i,j,t)}$ needed for excitation. If writing the GHM rule in a CA form, the state of each cell at position (i,j) at time step t is denoted as $c(i,j,t)$ and $c(i,j,t) \in$

$\{0, \dots, E, E+1, \dots, N-1\}$, which means the cell state can be one of the integer values in the set of $\{0, \dots, N-1\}$. When 0 is the excitable state, $\{1, \dots, E\}$ is the set of excited states and $\{E+1, \dots, N-1\}$ represents the set of refractory states. For given $\mathcal{N}_{c(i,j,t)}$, E, N and T, the evolution of the GHM starts from a initial setup and updates all the cells synchronously using the algorithm detailed as follows [95].

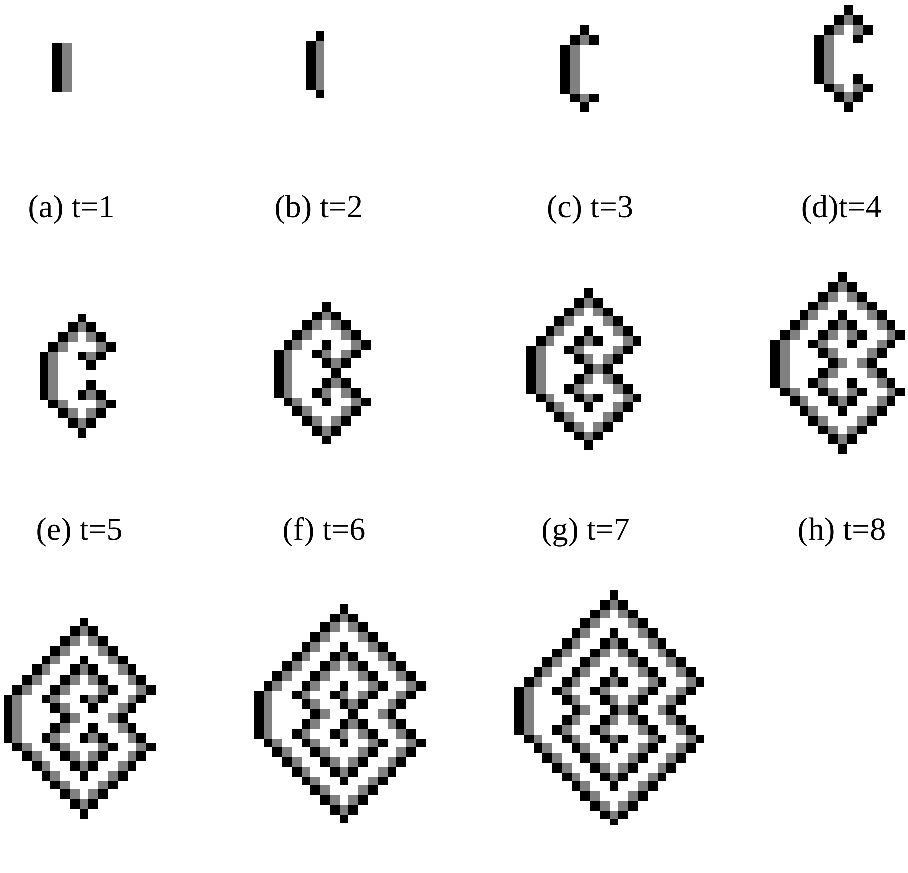
$$c(i,j,t+1) = \begin{cases} (c(i,j,t)+1) \; mod \; N, & \text{if } 1 \leq c(i,j,t) < N; \\ 1, & \text{if } c(i,j,t) = 0 \; AND \\ & \quad \#(\mathcal{N}_{c(i,j,t)}) \geq T; \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

where $\#(\mathcal{N}_{c(i,j,t)})$ denotes the number of excited cells in the neighbourhood $\mathcal{N}_{c(i,j,t)}$ of cell $c(i,j,t)$.

To date, GHM is widely and efficiently used to model many excitable media, because different complex patterns of reaction-diffusion systems can be generated using its simple rule. Here the application of the GHM is extended to model slime mould at the aggregation stage. Some simulation examples are given in the next section.

3.2.3 Simulation Studies

The simulation study of the GHM begins with a simple case which is applied on a square lattice 20×20 with a von Neumann neighbourhood. In this example, the initialisation of cell states is set at the time step of $t = 1$ as shown in Fig.3.1(a), in which black cells represent excited cells, the white are the cells in the excitable state, and others in grey colours indicate refractory cells. The parameters involved in this system Eqn.(3.1) are pre-set as $N = 3$, $E = 1$ and $T = 1$. Fig.3.1(b)-(k) clearly shows the evolution process of this model from $t = 2$ to $t = 11$, which produces a simple pattern with two spirals diffusing towards the same direction.



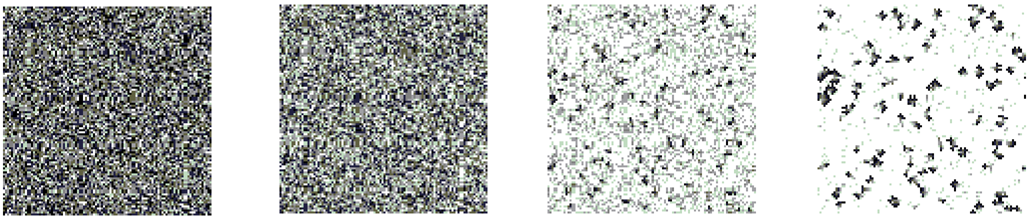
(a) t=1 (b) t=2 (c) t=3 (d)t=4

(e) t=5 (f) t=6 (g) t=7 (h) t=8

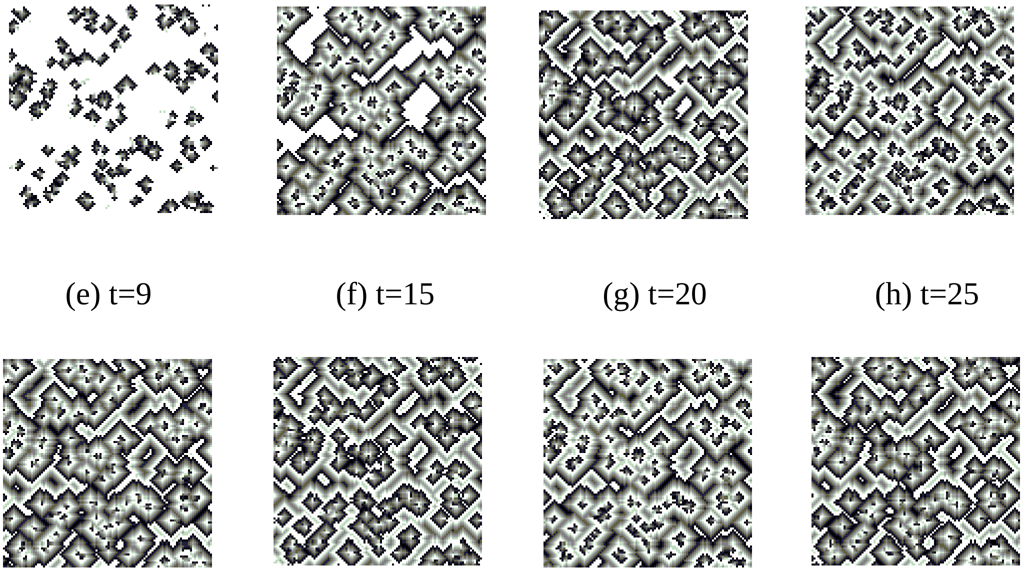
(i) t=9 (j) t=10 (k) t=11

Figure 3.1: The evolution of a GHM on a 20×20 lattice with the von Neumann neighbourhood and pre-set parameters of $N = 3, E = 1$ and $T = 1$.

Fig.3.2-3.4 demonstrates different GHM simulation results developed from different initialisations, neighbourhoods and values of parameters. They are all configured with random initialisations, and finally can generate complex ring and spiral patterns, which are close to the patterns investigated in slime mould aggregation.



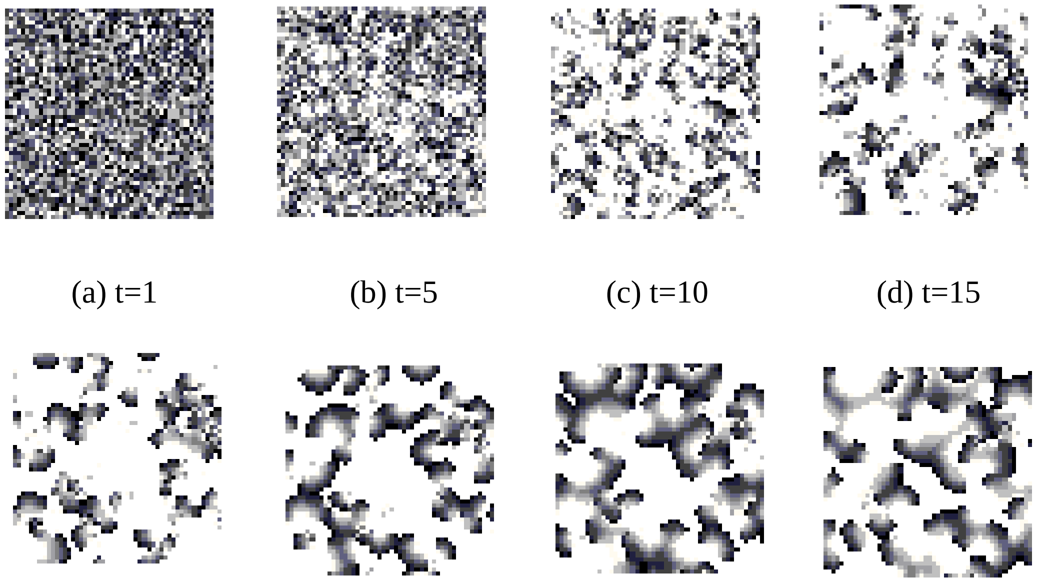
(a) t=1 (b) t=2 (c) t=5 (d) t=7



(e) t=9 (f) t=15 (g) t=20 (h) t=25

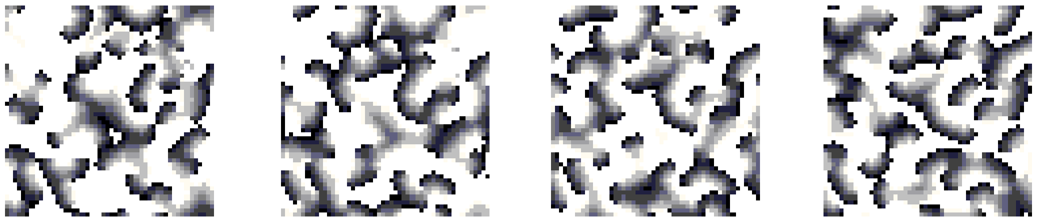
(i) t=30 (j) t=35 (k) t=40 (l) t=45

Figure 3.2: The evolution of a GHM on a 100×100 lattice with the von Neumann neighbourhood and pre-set parameters of $N = 7$, $E = 1$ and $T = 1$.



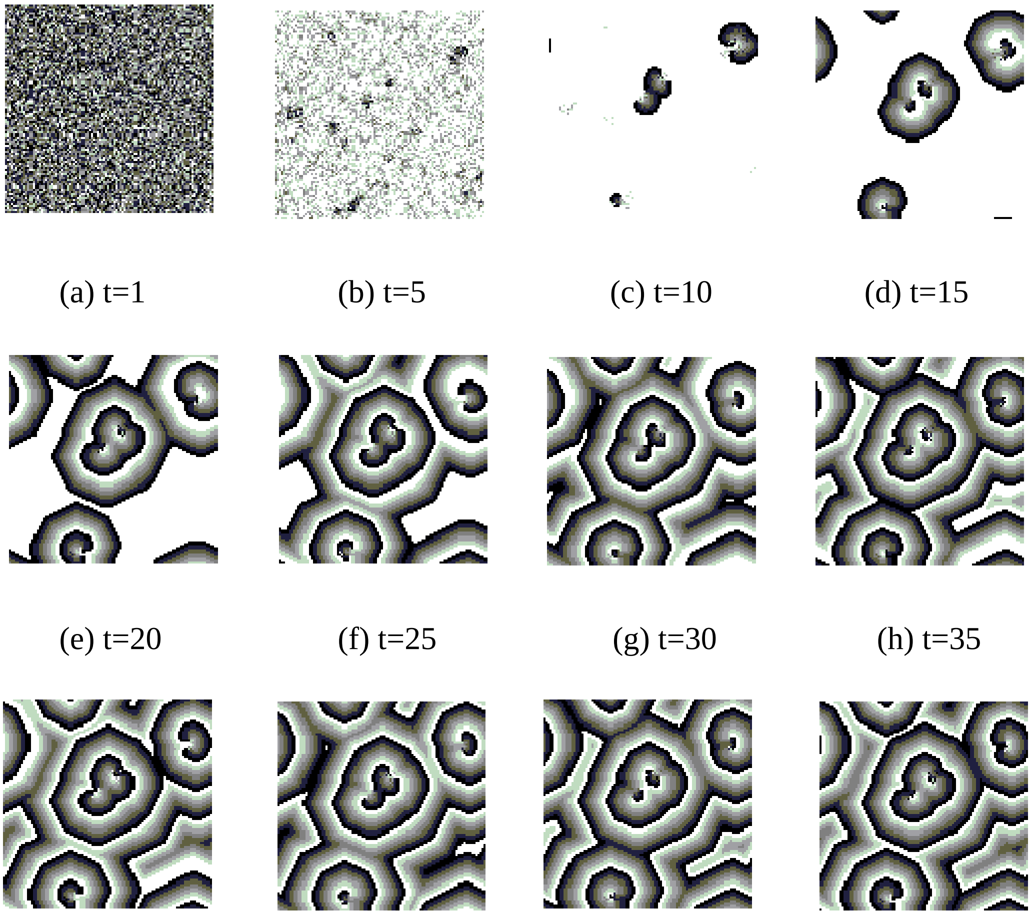
(a) t=1 (b) t=5 (c) t=10 (d) t=15

(e) t=20 (f) t=25 (g) t=30 (h) t=35



(i) t=40 (j) t=45 (k) t=50 (l) t=55

Figure 3.3: The evolution of a GHM on a 50×50 lattice with the Moore neighbourhood and pre-set parameters of $N = 11$, $E = 5$ and $T = 3$.



(a) t=1 (b) t=5 (c) t=10 (d) t=15

(e) t=20 (f) t=25 (g) t=30 (h) t=35

(i) t=40 (j) t=45 (k) t=50 (l) t=55

Figure 3.4: The evolution of a GHM on a 100×100 lattice with the Extended Moore neighbourhood and pre-set parameters of $N = 7$, $E = 1$ and $T = 3$.

3.3 CML Models-Solé's Model

The dynamics during the aggregation of slime mould can be regarded as a reaction-diffusion process which can be modelled using a discrete model of Coupled Map Lattices (CML). In this section, Solé's model [48] is extended to model pattern forming dynamics of aggregating slime mould, which is detailed as follows.

3.3.1 CML Models and Reaction-Diffusion Systems

Coupled Map Lattices (CML), a nonlinear model of space-time discrete dynamics with continuous states, may provide an efficient and rigorous basis for understanding the spatio-temporal dynamics involved in reaction-diffusion systems via its symbolic dynamic description [97]. Reaction-diffusion systems were initially applied in chemistry, which can describe oscillatory behaviours and diffusion of one or more chemical substances which are distributed in space. The reaction process is to transform substances into each other, and the diffusion process is to spread substances out over surface space, which causes a travelling wave.

Reaction-diffusion phenomena are common in physics, biology, geology and ecology, so that the study of the pattern formation in excitable reaction-diffusion systems has gradually become a research area of concern. The *Belousov-Zhabotinsky* (BZ) reaction is a typical example of reaction-diffusion systems [58]. Another example in biology is the excitable phenomena exhibited in aggregating colonies of slime mould *Dictyostelium Discoideum*, which has attracted great interest from biologists to investigate basic mechanisms of cellular interactions in such a reaction-diffusion system [29].

The traditional theoretical description of reaction-diffusion systems is usually in a continuous form of partial differential equations (PDEs). Generally, the bifurcation structure of PDEs is difficult to analyse, and the numerical solution is often time consuming, and sometimes even complicated to have an effective solution. In order to study the dynamical pattern formation in reaction-diffusion systems, discrete models based on a spatially distributed media are taken into account to simplify the modelling method. Discrete dynamical systems are directly suitable for numerical calculations and implementation on conventional digital computers. Another advantage of the discrete model is its simplicity and easy use for the simulation. This model sometimes can also capture many essential features of real reaction-diffusion systems such as oscillatory and excitable dynamics, so that considered from this point, the performance of the discrete model in modelling can sometimes be as good as that of PDEs. Hence, it is preferable to employ lattice models such as CMLs, when discrete models are available for some reaction-diffusion systems.

Several studies have applied CML approaches to approximate reaction-diffusion systems. For example, phase resetting dynamics are modelled using a coupled map model, which can faithfully generate the complex wave propagation process with major features in a relaxation oscillator reaction-diffusion system [98]. Another example shown in [99] is to use the CML model to investigate the phase transition and pattern formation of reaction-diffusion processes in chemical and biological systems.

When the CML model is applied to modelling slime mould, the original continuous system needs to be discretised both in space and time domains first. A lattice is used as an approximate approach to describe the continuous medium and time is discretised by sampling at every fixed time interval. Equations of the model should have a reasonable continuous limit with decrease of the spatial step and temporal step. One of main purposes in this section is to investigate the slime mould dynamics utilising discretised space and time corresponding to the continuous description of the reaction-diffusion process.

3.3.2 The Description of Solé's Model

A general CML model can be defined on a d-dimensional and n-cell lattice $\mathcal{X}^{\mathrm{d}}$. The cells in the lattice are indexed as integer vectors $c_i = (c_{i_1}, \dots, c_{i_n}) \in \mathcal{X}^{\mathrm{d}}$, $i \in \mathbb{Z}^{\mathrm{d}}$. This CML can be written as [76]

$$\begin{aligned} c_i(t) &= f_l + g_c \\ &= f_l\big(q^{-n_c}c_i(t), q^{-n_u}u_i(t)\big) + g_c(q^{-n_c}c_i(t), q^{-n_u}u_i(t), s^{m_c}q^{-n_c}c_i(t), \\ &\qquad s^{m_u}q^{-n_u}u_i(t)) \end{aligned} \tag{3.2}$$

where f_l and g_c represent the local mapping and the coupled function respectively. f_l depends on the local state and input variables, and g_c depends on local variables and the variables in a neighbourhood. q^{-n} is a backward time shift operator and s^m is a multi-valued spatial shift operator, both of which can be written as vectors,

$$q^{-n_c} = (q^{-1}, q^{-2}, \dots, q^{-n_c}) \tag{3.3}$$

$$q^{-n_u} = (q^{-1}, q^{-2}, \dots, q^{-n_u}) \tag{3.4}$$

$$s^{m_c} = \left(s^{\ell_c^1}, s^{-\ell_c^1}, \dots, s^{\ell_c^{m_c}}, s^{-\ell_c^{m_c}}\right) \tag{3.5}$$

$$s^{m_u} = \left(s^{\ell_u^1}, s^{-\ell_u^1}, \dots, s^{\ell_u^{m_u}}, s^{-\ell_u^{m_u}}\right) \tag{3.6}$$

Hence,

$$q^{-n_c}c_i(t) = \big(c_i(t-1), c_i(t-2) \dots, c_i(t-n_c)\big) \tag{3.7}$$

$$q^{-n_u}u_i(t) = \big(u_i(t-1), u_i(t-2) \dots, u_i(t-n_u)\big) \tag{3.8}$$

$$s^{m_c}c_i(t) = \left(c_{i+\ell_c^1}(t), c_{i-\ell_c^1}(t), \dots, c_{i+\ell_c^{m_c}}(t), x_{i-\ell_c^{m_c}}(t)\right) \tag{3.9}$$

$$s^{m_u}u_i(t) = \left(u_{i+\ell_u^1}(t), u_{i-\ell_u^1}(t), \dots, u_{i+\ell_u^{m_u}}(t), u_{i-\ell_u^{m_u}}(t)\right) \tag{3.10}$$

where $\ell_c, \ell_u \in \mathbb{Z}^{\mathrm{d}}$ are space shift indexes.

The CML model description in Eqn.(3.2) can be used to represent a wide range of nonlinear spatio-temporal dynamical systems, because the local reaction and the coupled function are both included in this model, so that the output depends on both local and coupling variables.

Solé's model, a CML model proposed by Solé and Valls [48], was initially employed to describe a ecosystem dynamic caused by the interaction between prey and predator populations. This mathematical model is

$$x_i(t+1) = \mu x_i(t)[1 - x_i(t)]\exp[-\beta y_i(t)] + D_1\nabla^2 x_i(t) \tag{3.11}$$

$$y_i(t+1) = x_i(t)\{1 - \exp\,[-\beta y_i(t)]\} + D_2\nabla^2 y_i(t) \tag{3.12}$$

where $i = (i_1, i_2) \in \mathbb{Z}^2$, D is the diffusion coefficient and the Laplace operator ∇^2 is a discrete diffusion operator,

$$\nabla^2 x_i(t) = \sum_j^n x_j(t) - n x_i(t) \tag{3.13}$$

where $j = (j_1, j_2) \in \mathbb{Z}^2$ and $x_j(t)$ are n neighbouring cells, usually taking $n =$ 4 or 8. In this case, $n = 4$, and then Eqn.(3.13) can be rewritten as

$$\nabla^2 x_{i_1,i_2}(t) = x_{i_1-1,i_2}(t) + x_{i_1+1,i_2}(t) + x_{i_1,i_2-1}(t) + x_{i_1,i_2+1}(t) - 4x_{i_1,i_2}(t) \tag{3.14}$$

According to Eqn.(3.9) and Eqn.(3.10), the neighbourhood of this model can be expressed using the spatial shift operator,

$$\begin{aligned} s^2 x_i(t) &= \left(s^{\ell^1}, s^{-\ell^1}, s^{\ell^2}, s^{-\ell^2}\right) x_i(t) \\ &= (x_{i+\ell^1}(t), x_{i-\ell^1}(t), x_{i+\ell^2}(t), x_{i-\ell^2}(t)) \end{aligned} \tag{3.15}$$

when $\ell^1 = (1\,, 0)$ and $\ell^2 = (0\,, 1)$,

$$s^2 x_{i_1,i_2}(t) = \left(x_{i_1-1,i_2}(t),\ x_{i_1+1,i_2}(t),\ x_{i_1,i_2-1}(t),\ x_{i_1,i_2+1}(t)\right) \tag{3.16}$$

Consider a small number of predators which appear at random positions initially and both populations are randomly generated as an initial condition,

$$\tau_1 < x_{i_0}(t) < \tau_2, \quad \forall i_0 \in \mathbb{Z}^2 \text{ and } t = 0 \tag{3.17}$$

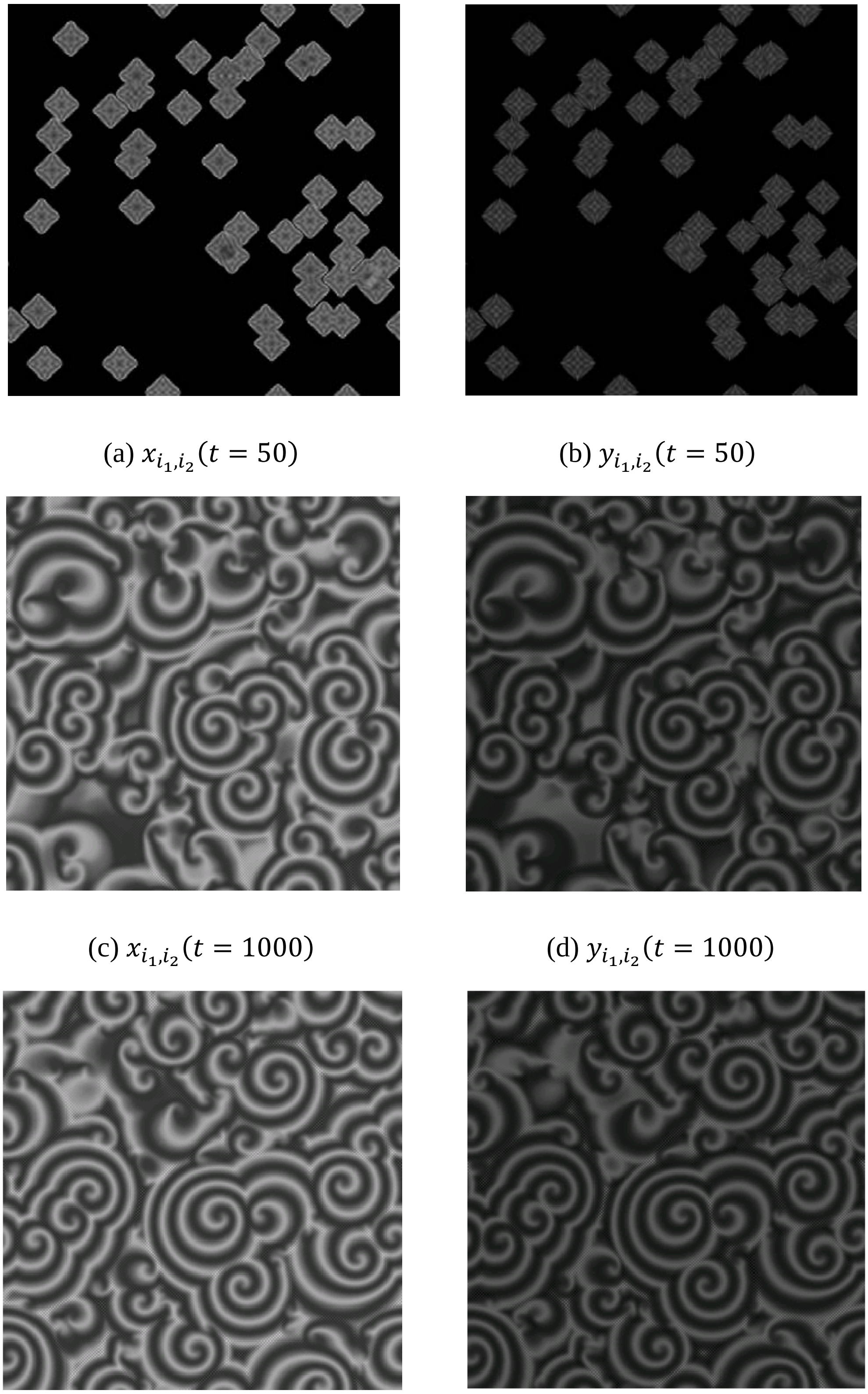
$$\tau_1 < y_{i_0}(t) < \tau_2, \quad \forall i_0 \in \mathbb{Z}^2 \text{ and } t = 0 \tag{3.18}$$

Here, $0 < \tau_1 < \tau_2 < 1$.

The examples demonstrated in [48] show the self-organisation caused by interacting populations of predators and preys under nonequilibrium conditions, which can produce chaotic spirals. This kinetic process is quite similar to that found in the aggregation of slime mould, which shares many properties with this predator-prey model, and similar patterns can also be found in this aggregation period. Hence, the model with two coupled equations of (3.11) and (3.12) can be extended to simulate the reaction-diffusion process of slime mould at the aggregation stage. The two variables in this CML model can respectively represent the normalised concentration of the chemoattractant cAMP which guides the movement of aggregating slime mould and activated receptors of slime mould cells. Spatial variation of the two species and diffusion of cAMP which are the two main processes of the reaction-diffusion system, are both included in this model. Some simulation examples of the model will be demonstrated in the next section.

3.3.3 Simulation Studies

When Solé’s model is applied to modelling slime mould, Eqn.(3.11) shows the changes of cAMP concentrations in space and Eqn.(3.12) shows the movement of slime mould. Consider simulations begin with 50 random seeds and every seed has two random values of $x_{i_0}(t)$ and $y_{i_0}(t)$ in the intervals of $\tau_1 < x_{i_0}(t) < \tau_2$ and $\tau_1 < y_{i_0}(t) < \tau_2$. Here, A 256×256 Lattice is used with the periodic boundary condition. The first example takes $\tau_1 = 0.2, \tau_2 = 0.4, \mu = 4.0, \beta = 4.0, D_1 = 0.002, D_2 = 0.2$ and simulation results are shown in Fig.3.5. Another simulation in Fig.3.6 exhibits smaller spiral patterns than those in the first example in Fig.3.5 by taking the same initialisation and parameter settings except $\beta = 5.0$.



(a) $x_{i_1,i_2}(t = 50)$

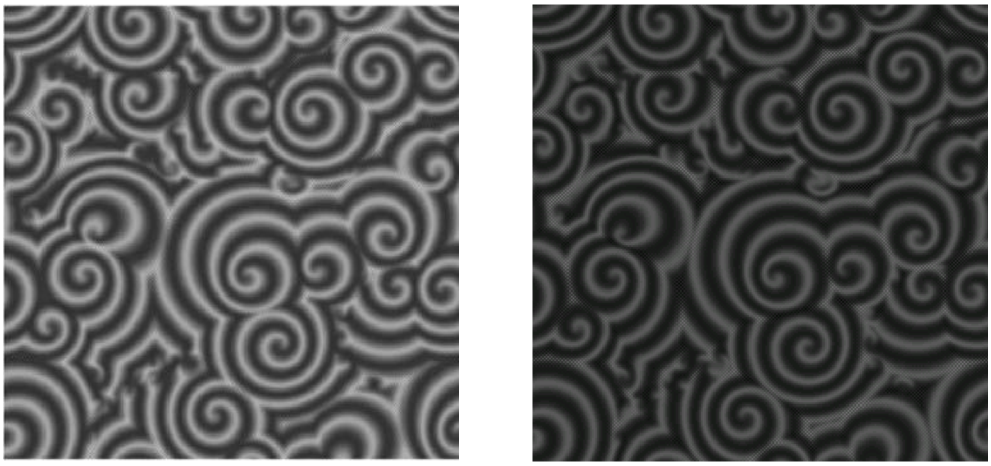
(b) $y_{i_1,i_2}(t = 50)$

(c) $x_{i_1,i_2}(t = 1000)$

(d) $y_{i_1,i_2}(t = 1000)$

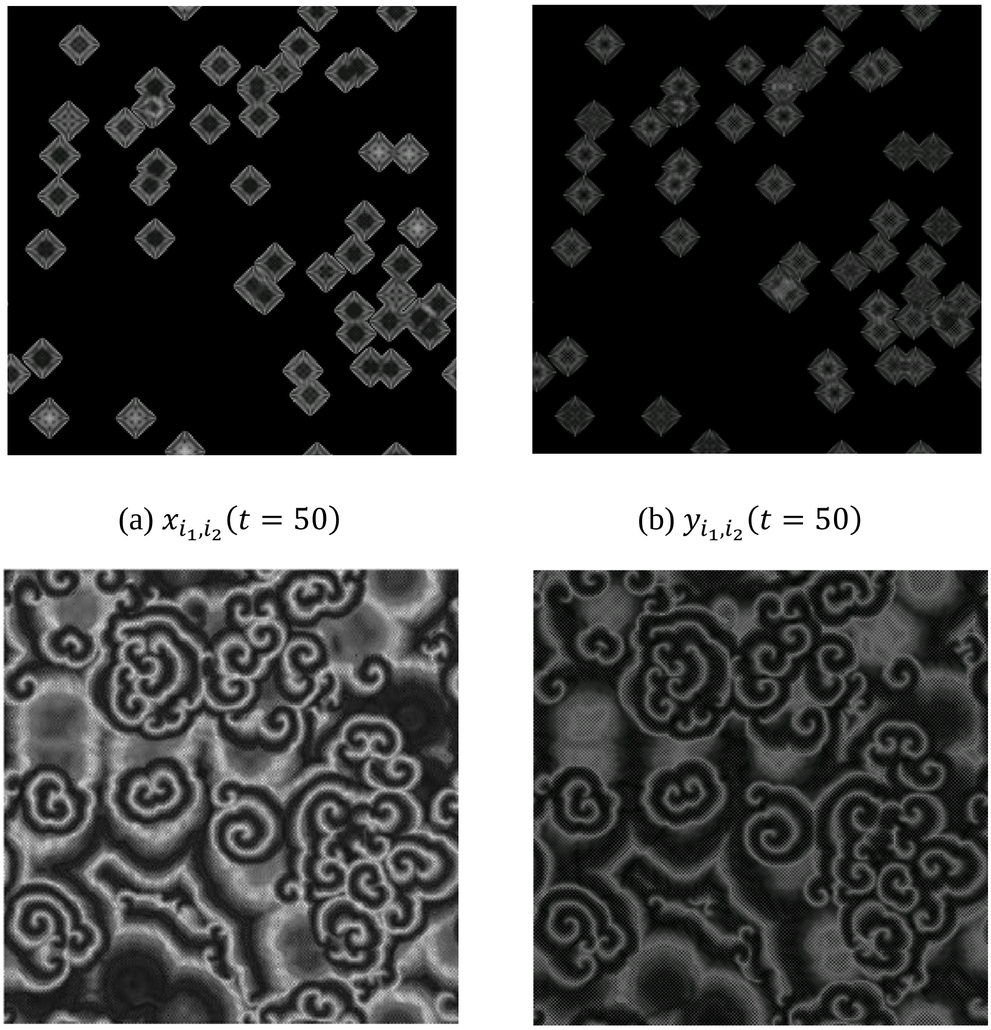
(e) $x_{i_1,i_2}(t = 2000)$

(f) $y_{i_1,i_2}(t = 2000)$



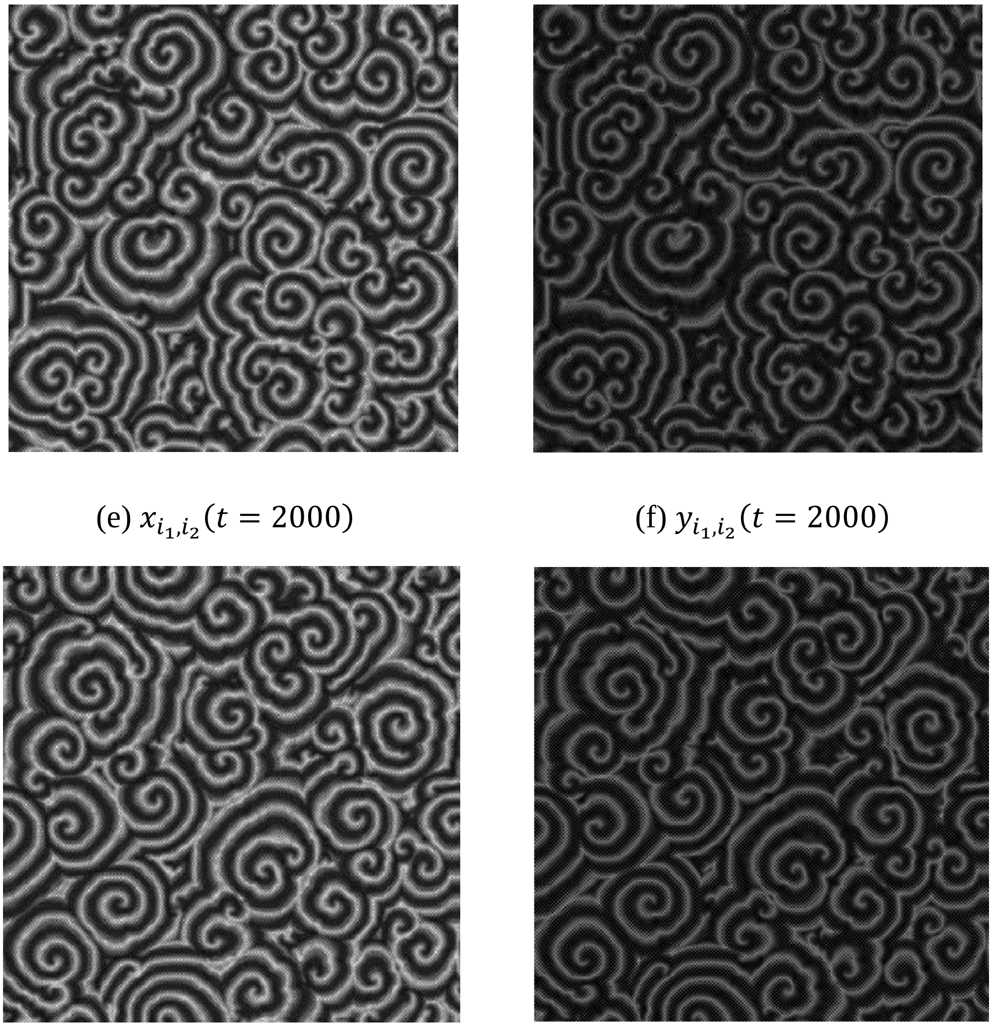
(g) $x_{i_1,i_2}(t = 4000)$ (h) $y_{i_1,i_2}(t = 4000)$

Figure 3.5: Simulations of the aggregating slime mould by a CML model on a 256×256 lattice with parameters of $\boldsymbol{\tau_1} = 0.2, \boldsymbol{\tau_2} = \mathbf{0.4}, \boldsymbol{\mu} = \mathbf{4.0}, \boldsymbol{\beta} = \mathbf{4.0}, \boldsymbol{D_1} = \mathbf{0.002}, \boldsymbol{D_2} = \mathbf{0}$.



(a) $x_{i_1,i_2}(t = 50)$ (b) $y_{i_1,i_2}(t = 50)$

(c) $x_{i_1,i_2}(t = 500)$ (d) $y_{i_1,i_2}(t = 1000)$



(e) $x_{i_1,i_2}(t = 2000)$

(f) $y_{i_1,i_2}(t = 2000)$

(g) $x_{i_1,i_2}(t = 4000)$

(h) $y_{i_1,i_2}(t = 4000)$

Figure 3.6: Simulations of the aggregating slime mould by a CML model on a 256×256 lattice with parameters of $\boldsymbol{\tau_1} = 0.2, \boldsymbol{\tau_2} = \mathbf{0.4}, \boldsymbol{\mu} = \mathbf{4.0}, \boldsymbol{\beta} = \mathbf{5.0}, \boldsymbol{D_1} = \mathbf{0.002}, \boldsymbol{D_2} = \mathbf{0.2}$

3.4 PDE models-Kawasaki’s Model

As mentioned in Section 3.3, partial differential equations (PDEs) are an alternative method to formulate reaction-diffusion systems. It is a direct and original modelling method widely used to describe reaction-diffusion phenomena, because generally the continuity of PDEs is consistent with that of practical systems and variables in this mathematical model are commonly set with corresponding physical significances, so that the PDE model may have a better physical explanation.

Kawasaki's model [49] is a reaction-diffusion model including a pair of PDEs. This model was constructed and designed to capture colonial features of a bacterium named *Bacillus subtilis* on the surface of thin agar plates, and to produce various morphological patterns observed in the colonial growth such as a dense-branching morphology (DBM) with a smooth circular envelope and disk-like colony patterns, which are caused by different environmental conditions. Similar morphological patterns such as streaming patterns have also been found in the growth of slime mould *physarum plasmodium* [100]. When placed on a culture medium, the *plasmodium* extends and develops dendritic veins crawling freely away from the centre. In this way, the vein structure is formed with a sheet-like structure in the periphery. Hence, Kawasaki's model here is extended to explain characteristic patterns of colonial slime mould.

3.4.1 The Mathematical Description

Two coupled equations are involved in this model to represent cell movements with the consumption of nutrient and the change of nutrient concentration with cell proliferation respectively. Denote cell density at time step t and two-dimensional spatial position (i_1, i_2) as $w_i(t)$ and concentration of nutrient as $v_i(t)$, the mathematical model is written as

$$\frac{\partial v}{\partial t} = D_v \nabla^2 v - f(v, w) \tag{3.19}$$

$$\frac{\partial w}{\partial t} = \nabla \cdot \{D_w \nabla w\} + \vartheta f(v, w) \tag{3.20}$$

where D_v is the diffusion coefficient of nutrient and is set to be a constant, and D_w is the diffusion coefficient of slime mould cells which is determined by the cell density and the nutrient concentration, say

$$D_w = \alpha v w, \quad \alpha = \alpha_0(1 + \Delta) \tag{3.21}$$

where α is the agar concentration coefficient and Δ is a stochastic fluctuation variable for the cell movement. $f(v, w)$ in Eqn.(3.19) indicates the nutrient consumption and $\vartheta f(v,w)$ in Eqn.(3.20) is the cell growth amount. When the nutrient concentration is low, $f(v,w)$ can be described as:

$$f(v,w) = \kappa v w \tag{3.22}$$

where κ is the intrinsic consumption rate. Given $\vartheta = 1, \kappa = 1, D_v = 1$, Eqn. (3.21) and (3.22) are substituted into Eqn. (3.19) and (3.20), say

$$\frac{\partial v}{\partial t} = \nabla^2 v - wv \tag{3.23}$$

$$\frac{\partial w}{\partial t} = \nabla \cdot \{\alpha v w \nabla w\} + vw \tag{3.24}$$

with the initial conditions of $v_i(0) = v_0, w_i(0) = w_0$, and the no-flux boundary conditions,

$$\frac{\partial v}{\partial \mathbb{m}} = 0 \qquad \forall v \in \partial\Omega \tag{3.25}$$

$$\frac{\partial \boldsymbol{w}}{\partial \mathbb{m}} = 0 \qquad \forall w \in \partial\Omega \tag{3.26}$$

where $\mathbb{m}$ denotes the exterior normal to the boundary $\partial\Omega$.

3.4.2 The Numerical Simulation Method

There are several computer simulation methods used in numerical analysis such as the implicit method and the explicit method, but for this model better results can be obtained by using the explicit method. Thus, Eqn.(3.23) and (3.24) can be approximated explicitly as

$$\partial v_i(t) = v_i(0) + (\nabla^2 v_i(t-1) - w_i(t-1)v_i(t-1))\Delta t \tag{3.27}$$

$$\begin{aligned}\partial w_i(t) = w_i(0) + (\nabla \cdot \{\alpha v_i(t-1) w_i(t-1) \nabla w_i(t-1)\} \\ + v_i(t-1) w_i(t-1))\Delta t\end{aligned} \tag{3.28}$$

Here $i = (i_1, i_2) \in \mathbb{Z}^2$ and the two-dimension spatial expression of the Laplace operator ∇^2 can be described as

$$\begin{aligned}\nabla^2 v_{i_1,i_2}(t-1) = (v_{i_1-1,i_2}(t-1) + v_{i_1+1,i_2}(t-1) + v_{i_1,i_2-1}(t-1) \\ + v_{i_1,i_2+1}(t-1) - 4v_{i_1,i_2}(t-1))/(\Delta x)^2\end{aligned} \tag{3.29}$$

where Δt and Δx are the time interval for a time step and the width of a lattice unit respectively. $\nabla \cdot$ is the divergence operator and $\nabla = (\frac{\partial}{\partial i_1}, \frac{\partial}{\partial i_2})$ is the gradient operator.

According to the product rule, $\nabla \cdot \{\alpha v w \nabla w\}$ in Eqn.(3.24) can be derived as

$$\begin{aligned} \nabla \cdot \{\alpha v w \nabla w\} &= \nabla(\alpha v \cdot w) \cdot \nabla w + (\alpha v \cdot w)\nabla \cdot (\nabla w) \\ &= \nabla(\alpha v \cdot w) \cdot \nabla w + \alpha(v \cdot w)\nabla^2 w \end{aligned} \tag{3.30}$$

For the approximation of the derivative, forward difference, backward difference and central difference are commonly used forms. In this model, a central-difference approximation is applied as a more reasonable and better method. For example,

$$\frac{\partial v}{\partial i_1} = \frac{v_{i_1+\Delta x, i_2} - v_{i_1-\Delta x, i_2}}{2\Delta x} \tag{3.31}$$

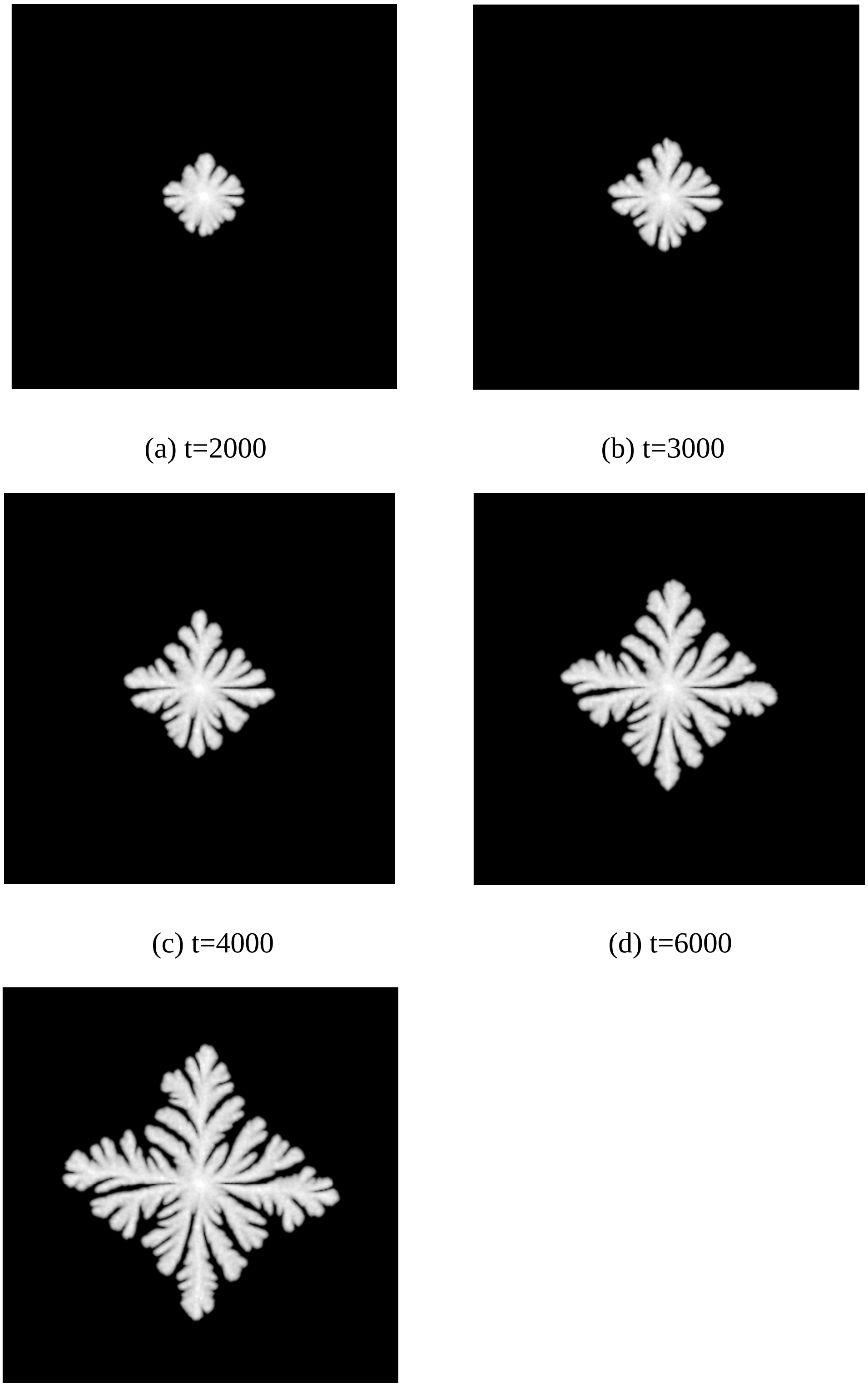
3.4.3 Simulation Studies

Computer simulations for the model in Section 3.4.2 are employed on a lattice with the size of 400×400 and no-flux boundary conditions. For the initialisation, slime mould cells are distributed in a round-shaped area in the centre of the lattice with the greatest density value in the centre and gradually reduced densities around, which can be formulated as,

$$w_i(0) = w_{i_1,i_2}(0) = w_M \exp\{-({i_1}^2 + {i_2}^2)/6.25\} \tag{3.32}$$

where w_M is the maximum density. Initial nutrient is evenly distributed at a level v_0. For $\alpha = \alpha_0(1 + \Delta)$, α is random and normally distributed with the mean α_0. The time and space units are fixed as $\Delta t = 0.2$, $\Delta x = 1$.

Simulation results of cell densities in Fig.3.7, 3.8 and 3.9 show different branches and dendritic patterns, which are close to the ones investigated in real experiments of slime mould, under different initial conditions and diffusion parameter settings.



(a) t=2000

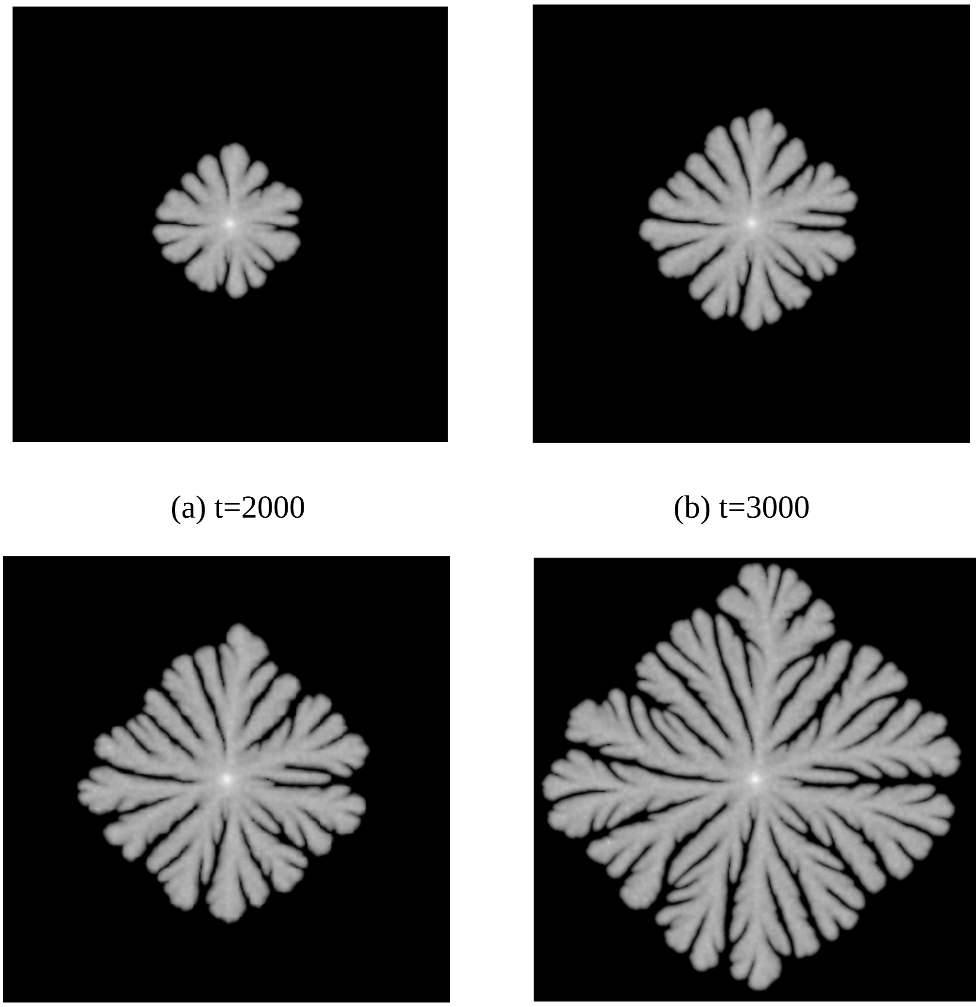
(b) t=3000

(c) t=4000

(d) t=6000

(e) t=8000

Figure 3.7: Streaming pattern revolutions with $w_M = 0.71$, $v_0 = 0.5$, $\alpha_0 = 2$



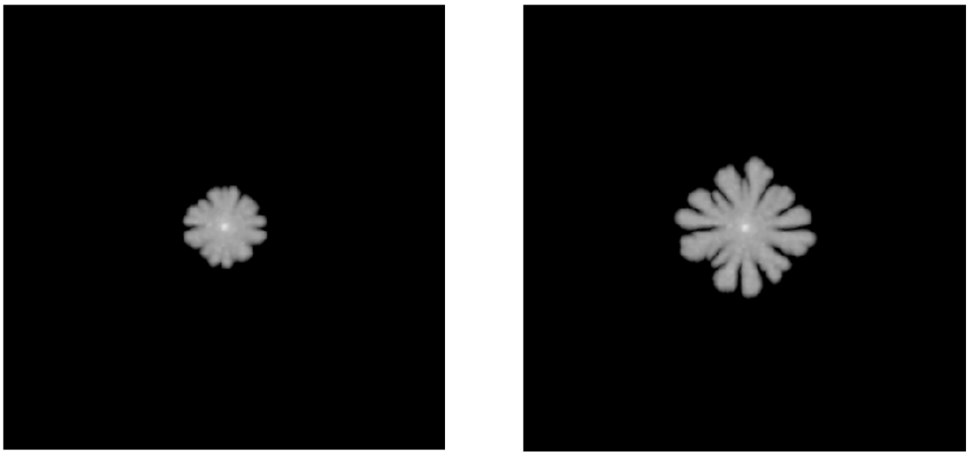
(a) t=2000

(b) t=3000

(c) t=4000

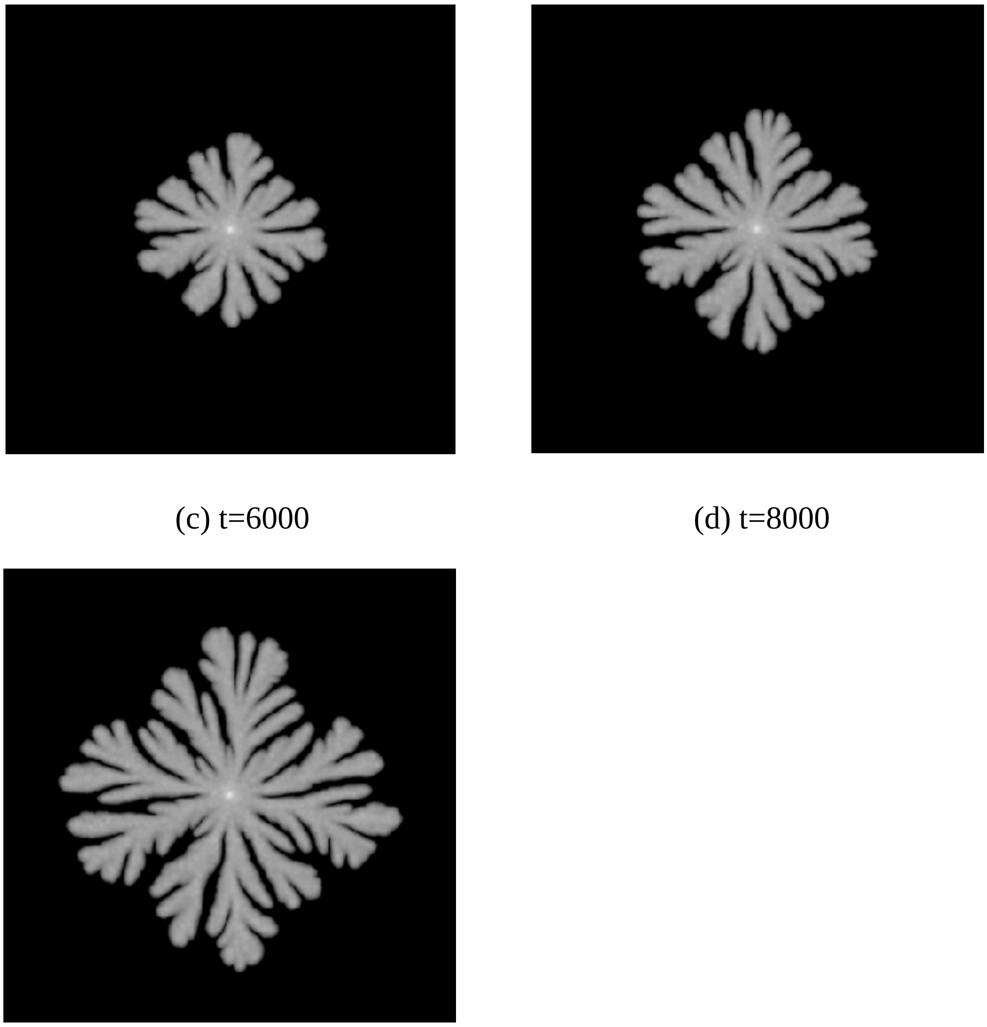
(d) t=6000

Figure 3.8: Streaming pattern revolutions with $w_M = 0.71$, $v_0 = 0.5$, $\alpha_0 = 4$



(a) t=2000

(b) t=4000



(c) t=6000

(d) t=8000

(e) t=12000

Figure 3.9: Streaming pattern revolutions with $w_M = 0.71, v_0 = 0.35, \alpha_0 = 4$

It is shown from the simulation results that when α increases, the diffusion velocity increases. In addition, the higher concentration of the initial nutrition, the more branches and faster diffusion velocity the evolution has.

3.5 Conclusions

GHM and Solé’s model are discrete models which can produce similar patterns of concentric circles or spirals with that of the aggregating slime mould. Discreteness of both models makes numerical simulations easy to implement with a reduced computational complexity. However, discretisation on the other hand causes some problems for modelling as well. For example, GHM is good for producing biological patterns with a simple rule, but it is only involved in the movement of slime mould cells and leaves out the information of the chemoattractant which functions essentially in chemotaxis of slime mould. For Solé's model, though it shows a good reaction-diffusion property consistent with that of slime mould, the sampling in time and space domains may affect the accuracy of predicted models. The PDE model such as Kawasaki's model is a continuous model which can provide better physical explanations corresponding to real systems than the other two spatio-temporal models - CA and CML, but it is more difficult for the computer simulation. Hence, these three models each has its advantages, so that all of them are widely used to model real systems in different areas such as chemistry, physics and biology. The simulation results in this chapter also proves that CA, CML and PDE models all have capabilities of modelling slime mould dynamics. Simulation studies can be summarised in Fig. 3.10.

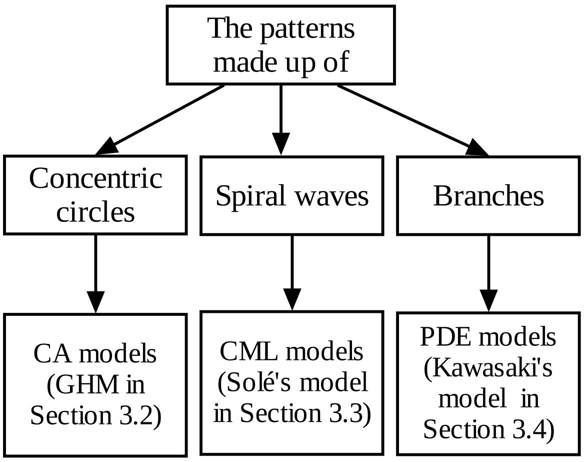

The patterns made up of
Concentric circles
Spiral waves
Branches
CA models (GHM in Section 3.2)
CML models (Solé's model in Section 3.3)
PDE models (Kawasaki's model in Section 3.4)

Figure 3.10: The flow diagram for simulation examples.

Chapter 4

Model Term Selection using Mutual Information for Spatio-temporal System Identification

4.1 Introduction

Spatio-temporal systems represent a class of complex dynamic systems, which contain both time and space information. The study of spatio-temporal systems may help to decipher many spatio-temporal phenomena and behaviours that appear in nature and to better understand and possibly control the formation of spatio-temporal patterns [6, 53, 61].

One of the key concerns in the analysis of spatio-temporal systems is system identification, the reverse problem of pattern formation, which is still an open problem. One of main tasks in spatio-temporal system identification is model structure selection which enables construction of a mathematical model from experimental data. The Orthogonal Forward Regression (OFR) algorithm is one of the effective methods for the identification for spatio-temporal systems. Given a large number of candidate model terms in an initial model, this algorithm can be used to determine which terms or regressors are significant and should be included in the model based on the Error Reduction Ratio (ERR) [45, 46]. However, the training data cannot be always ideal and often have some inherent problems such as data uncertainty and fast sampling rates, which can affect the selection of model structure in a spatio-temporal system identification. Hence, when applied to some spatio-temporal data sets with a not-so-good performance, the OFR algorithm can occasionally select some spurious model terms, which can then result in a comparatively more complex model with some possible insignificant or redundant model terms.

In this chapter, a new method, called the OFR-MI (Orthogonal Forward Regression using Mutual Information) algorithm is introduced for spatio-temporal system identification. Two previous studies of using the mutual information technique in nonlinear identification tasks are reviewed. For the new algorithm, mutual information is used as the criterion for detecting important terms in spatio-temporal models for the first time. It has been shown that the OFR-MI algorithm can effectively avoid the high ERR value problems which seems to occur for some spatio-temporal systems. The new algorithm is tested on several benchmark spatio-temporal models including CA, CML and PDE models, and then identified models are validated by a model length determination method.

4.2 Mutual Information

Mutual Information (MI) which was initially proposed by Shannon in 1948 [101], is one of the effective measurements of the similarity between two variables. If two variables are strictly independent, the MI between the two variables should be zero.

Consider Y_1 and Y_2 are two stochastic sequences with marginal probability density functions $p(y_1)$ and $p(y_2)$ respectively. $p(y_1, y_2)$ is the joint probability density function. The mutual information $I(y_1, y_2)$ is defined as

$$I(y_1, y_2) = \sum_{y_1 \in Y_1} \sum_{y_2 \in Y_2} p(y_1, y_2) \log\left(\frac{p(y_1, y_2)}{p(y_1)p(y_2)}\right) \tag{4.1}$$

Take the NARMAX model as a linear-in-parameters regression form, which is mentioned in Section 2.3.1, Chapter 2, for example,

$$y(t) = \sum_{i=1}^{M_S} \theta_i \phi_i(t) + \xi(t), \qquad t = 1, \dots, N \tag{4.2}$$

where $y(t)$ is the output and $\phi_i(t)$ is one of the orthogonal regressors in the model, the mutual information $I(\phi_i, y)$ between ϕ_i and y measures how a knowledge of ϕ_i reduces the uncertainty about y, or the information that ϕ_i and y share. Hence, the regressor ϕ_i with the biggest MI value may make the most contribution to the model. Thus, MI incorporated with an orthogonalisation procedure can be used as an alternative to the ERR term selection procedure in the classical OFR algorithm to aid the selection of significant model terms. Several algorithms have been developed to estimate mutual information from observed data, including the approach using a histogram based technique [102, 103], methods based on kernel density estimators [104], and parametric methods [105]. In this work, the adaptive histogram-based method proposed in [102] is employed, because this method is applicable to any distribution and appears to be asymptotically unbiased and efficient [106]. In this estimator, adaptive partitioning is used to approximate probabilities in mutual information by calculating relative frequencies on appropriate partitions and the partition is refined until conditional independence has been reached on its cells.

4.3 Previous Studies on Model Term Selection using Mutual Information

Model term selection which is directly related to the model structure determination is key to the nonlinear dynamic system identification. A good identified model is definitely constructed by all the true model terms, so that selecting correct terms from a large set of candidate model terms is the premise of the correct model structure. The commonly used method for model term selection is the OFR algorithm which is based on the orthogonalisation and squared correlation. This term selection algorithm has been viewed as one of the most effective methods to solve identification problems, so more and more studies have aimed to optimise this algorithm or improve the efficiency of it.

Following the description of the classic OFR algorithm, two previous studies of model term selection based on the OFR algorithm, are explained in this section.

Both of them adopted the mutual information technique which, however, is used in different stages of the identification. For the method employed by Zhao and Billings [71], MI is used in the neighbourhood detection which provides a smaller neighbourhood range for the later steps of the identification. Wei and Billings [107] took MI as the criterion for measuring the significance of candidate model terms in the identification for non-linear temporal systems.

4.3.1 Identification of Non-linear Dynamic Models using the OFR Algorithm

For a real system without any prior knowledge, the NARMAX model with a linear-in-parameters model structure can be expressed as

$$y(t) = \sum_{i=1}^{M} \theta_i \phi_i^*(t) + \varepsilon(t), \qquad t = 1, \dots, N \tag{4.3}$$

where $\phi_i^*(\cdot)$ are all the candidate model terms which are different combinations of various input and output variables. θ_i are the unknown parameters corresponding to terms $\phi_i^*(t)$. M is the number of all model terms and N is the length of data. $\varepsilon(t)$ is the modelling error.

Not all the terms in Eqn.(4.3) are significant to the model and some redundant model terms can be removed in order to finally obtain optimised model with low computational complexity as shown in Eqn.(4.2). The Orthogonal Forward Regression (OFR) algorithm is originally designed to determine the set of significant terms in the model and estimate corresponding unknown parameters [45, 46, 108-110]. As one of classical identification methods for non-linear systems, the OFR algorithm has been widely used to identify various dynamical systems such as radial basis function networks [111, 112], CML models [113], and CA models [114, 115]. The OFR algorithm is briefly describes as below [45].

According to Eqn.(2.11)-(2.14) in Section 2.3.2, Chapter 2, Eqn.(4.3) can be written as

$$Y = (\Phi A^{-1})(A\Theta) + \mathbf{e} = Wg + \mathbf{e} \tag{4.4}$$

with

$$Y = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}, \Phi = \Phi^{\mathrm{T}}(t) = \big(\phi_1(t), \dots, \phi_M(t)\big), \Theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_M \end{bmatrix}, \mathbf{e} = \begin{bmatrix} \varepsilon_1(t) \\ \vdots \\ \varepsilon_N(t) \end{bmatrix}$$

where $g = [g_1, g_2, \dots, g_M]^{\mathrm{T}}$ is an auxiliary vector. W is a $N \times M$ orthogonal matrix because

$$W^{\mathrm{T}}W = (\Phi A^{-1})^{\mathrm{T}}(\Phi A^{-1}) = \mathcal{D} = \mathrm{Diag}[d_1, d_2, \dots, d_M] \tag{4.5}$$

with $d_i = \langle w_i, w_i \rangle = \sum_{t=1}^{N} w_i^2(t)$, where $\langle \cdot, \cdot \rangle$ denotes inner product of two vectors. Hence, orthogonal regression matrix can be calculated recursively from

$$W = \Phi A^{-1} = \Phi - W(A - I) \tag{4.6}$$

so

$$\begin{aligned} &w_1(t) = \phi_1(t), \\ &w_k(t) = \phi_k(t) - \sum_{i=1}^{k-1} a_{ik}\, w_i(t), \qquad k = 2, \dots, M \end{aligned} \tag{4.7}$$

where

$$a_{ik} = \frac{\sum_{t=1}^{N} w_i(t)\phi_k(t)}{\sum_{t=1}^{N} w_i^2(t)}, \qquad k = 2, \dots, M \text{ and } i < k \tag{4.8}$$

because $= \mathcal{D}^{-1}W^{\mathrm{T}}\Phi$. The auxiliary regressor $w_i(t)$ can be used to help decide which terms are significant and should be included in the model. Assume the residual signals $\varepsilon(t)$ is independent and uncorrelated with any input and output variables at past times, which ensures $W^{\mathrm{T}}\mathbf{e} = 0$. Hence, the output variance can be represented as

$$\begin{aligned} \frac{1}{N}Y^{\mathrm{T}}Y &= \frac{1}{N}(g^{\mathrm{T}}W^{\mathrm{T}}Wg + \mathbf{e}^{\mathrm{T}}\mathbf{e} + g^{\mathrm{T}}W^{\mathrm{T}}\mathbf{e} + \mathbf{e}^{\mathrm{T}}Wg) \\ &= \frac{1}{N}(g^{\mathrm{T}}W^{\mathrm{T}}Wg + \mathbf{e}^{\mathrm{T}}\mathbf{e}) \end{aligned}$$

$$= \sum_{i=1}^{M} \left[g_i^2 \frac{1}{N} \sum_{t=1}^{N} w_i^2(t) \right] + \frac{1}{N} \mathbf{e}^{\mathrm{T}} \mathbf{e} \qquad (4.9)$$

so that $g_i^2 \frac{1}{N} \sum_{t=1}^{N} w_i^2(t)$ can be viewed as the contribution to the dependent variable variance introduced by $w_i(t)$ and the Error Reduction Ratio (ERR) at *i*th step can be defined as a criterion for model structure selection, say

$$ERR_i = \frac{\hat{g}_i^2 \langle w_i, w_i \rangle}{\langle Y, Y \rangle} = \frac{\hat{g}_i^2 \sum_{t=1}^{N} w_i^2(t)}{\sum_{t=1}^{N} Y^2(t)} \qquad (4.10)$$

where $\hat{g}$ is the estimated g.

$$\begin{aligned} \because g &= A\Theta = \mathcal{D}^{-1} W^{\mathrm{T}} \Phi \Theta \\ &= \mathcal{D}^{-1} W^{\mathrm{T}} \Phi \Phi^{-1} (Y - \mathbf{e}) = \mathcal{D}^{-1} W^{\mathrm{T}} (Y - \mathbf{e}) \\ &= \mathcal{D}^{-1} W^{\mathrm{T}} Y - \mathcal{D}^{-1} W^{\mathrm{T}} \mathbf{e} \end{aligned} \qquad (4.11)$$

$$\begin{aligned} \therefore \hat{g} &= \mathcal{D}^{-1} W^{\mathrm{T}} Y \\ &= \frac{\sum_{t=1}^{N} w_i(t) Y(t)}{\sum_{t=1}^{N} w_i^2(t)}, \qquad i = 1, \dots, M \end{aligned} \qquad (4.12)$$

In a forward-regression manner, a term corresponding to the maximum value of ERR_i is selected in the model of Eqn.(4.2) as the significant term at each step. This procedure is not terminated till at M_sth step when

$$1 - \sum_{i=1}^{M_s} ERR_i < \text{a threshold value}, \;\; \text{and } M_s \leq M \qquad (4.13)$$

4.3.2 Zhao’s Neighbourhood Detection using Mutual Information

Zhao and Billings [71] proposed a new neighbourhood detection approach for the identification of CA models. Based on mutual information, a coarse range of the candidate neighbourhood for a CA model can be detected without any a priori information about the model structure. According to the basic histogram method, marginal and joint probabilities in the MI which is defined in Eqn. (4.1) are approximated as,

$$p(y_{1i}) = \frac{s_{y_{1i}}}{N}, \qquad p(y_{2j}) = \frac{s_{y_{2j}}}{N}, \qquad p(y_{1i}, y_{2j}) = \frac{s_{y_{1i},y_{2j}}}{N}. \tag{4.14}$$

Here let $y_{1i} \in Y_1 = \{y_{11}, \dots, y_{1m}\}$ and $y_{2j} \in Y_2 = \{y_{21}, \dots, y_{2n}\}$. $s_{y_{1i}}$, $s_{y_{2j}}$ and $s_{y_{1i},y_{2j}}$ are counters when $y_1 = y_{1i}$, $y_2 = y_{2j}$ and $(y_1 = y_{1i}) \cap (y_2 = y_{2j})$ respectively. N is the number of pairs (y_1, y_2), so that the MI can be rewritten as

$$I(y_1, y_2) = \sum_{i,j} p(y_{1i}, y_{2j}) \log\left(\frac{p(y_{1i}, y_{2j})}{p(y_{1i})p(y_{2j})}\right)$$

$$= \sum_{i=1}^{m}\sum_{j=1}^{n} \frac{s_{y_{1i},y_{2j}}}{N} \log\left(\frac{\frac{s_{y_{1i},y_{2j}}}{N}}{\frac{s_{y_{1i}}}{N} \cdot \frac{s_{y_{2j}}}{N}}\right)$$

$$= \sum_{i=1}^{m}\sum_{j=1}^{n} \frac{s_{y_{1i},y_{2j}}}{N} \log\left(\frac{N s_{y_{1i},y_{2j}}}{s_{y_{1i}} s_{y_{2j}}}\right) \tag{4.15}$$

In order to obtain a smaller neighbourhood for later identification processes, Zhao's algorithm is to test the correlation between the output and any sub-neighbourhood which is constructed by combinations of neighbouring cells from an initial neighbourhood. If the selected neighbourhoods include all the right neighbouring cells, the MI between these neighbourhoods as the input and the output will be close to 1; On the contrary, if not all the right neighbouring cells are in the input, the MI should be close to 0. Thus a new criterion is introduced to rank each candidate neighbourhood, say

$$Cr = I - OE = \sum_{i,j} p(y_{1i}, y_{2j}) \log\left(\frac{p(y_{1i}, y_{2j})}{p(y_{1i})p(y_{2j})}\right) - \frac{2^m}{N} \tag{4.16}$$

where m is the number of cells in the corresponding sub-neighbourhood. $OE = \frac{2^m}{N}$ is the over-estimation of the MI [116]. All the possible neighbourhoods are tested by this criterion and when the Cr reaches the peak value, the corresponding neighbourhood can be selected as the output.

This neighbourhood detection algorithm can narrow down the search range for the correct neighbourhood, which considerably reduces the computational complexity in the later steps of the model determination. However, this approach cannot always provide an exactly correct neighbourhood, especially for the neighbourhood with an asymmetrical space structure. Hence, the output neighbourhood of this algorithm can be used as the initial neighbourhood for the CA-OLS algorithm [69], which can finally obtain the correct neighbourhood and the model structure. More details can be found in [71].

4.3.3 Wei's Identification Method using Mutual Information

Wei and Billings [107] introduced an efficient integrated forward orthogonal search (IFOS) algorithm assisted by ERR and mutual information techniques. This algorithm was designed for nonlinear temporal systems, which have NARX (Nonlinear AutoRegressive with eXogenous inputs) model structures in a linear-in-the-parameters form as Eqn.(2.10). The aim of this algorithm is to solve the model structure determination problem in the identification of nonlinear temporal systems.

Model structure selection is essential to the nonlinear system identification. Generally, a large set of candidate model terms or basis functions may be involved in the model structure selection at the beginning. The main task of the model structure selection is to determine which terms should be included in the model and which ones should be out of the model, and finally produce a parsimonious model. Insignificant model terms might result in a more complex model with a large number of parameters, which may become oversensitive to training data or exhibit poor generalisation properties. One of the most efficient model structure detection methods for these temporal systems is the OLS (Orthogonal Least Squares) or OFR (Orthogonal Forward Regression) algorithm, in which significant model terms can be ranked based on ERR values, and then according to the order, model terms are selected one at a time. However, it has been observed in some cases that the OLS-ERR algorithm may sometimes select incorrect model terms which are redundant to the model because of the training data contaminated by certain noise sequences [117] or the poorly designed input [118]. To avoid selecting incorrect model terms using the OLS-ERR algorithm, a MI-based criterion for measuring the significance of model terms is introduced into the orthogonalisation procedure as a complementary approach to the ERR criterion. The integrated criteria are the core of the new IFOS algorithm. In order to gain an accurate model structure, this new IFOS algorithm then combines with a simple hypothesis test (*t*-test) and a general cross-validation (GCV) criterion, which enhances the capability of detecting any spurious model terms and suggests an appropriate number of model terms respectively.

To solve the identification problem for nonlinear systems, an approach with four-stage trial-and-error experiments was employed here. Firstly, five candidate model term dictionaries were selected. Secondly, the IFOS algorithm was applied over these five term dictionaries respectively, so that different model structures with different model terms could be obtained. The performance of these different identified models was compared and then the best model structure according to a specified criterion was selected. Finally, after a model refinement procedure with the *t*-test, the accurate model could be obtained with correct model terms and re-estimated term parameters. More details can be found in [107].

4.4 The New OFR-MI Algorithm

The new OFR-MI algorithm for the identification of dynamic spatio-temporal systems is introduced in this section, because the classic OFR algorithm shows a poor performance in some spatio-temporal identification cases. Assisted by a model length determination method, this new algorithm using mutual information has a capability of detecting the correct model structure efficiently.

4.4.1 The Performance of the OFR Algorithm

The OFR algorithm with the ERR criterion has been widely applied in model structure selection for dynamic spatio-temporal systems. Owing to its efficiency in most identification cases such as nonlinear function approximation and neural network training [119-121], the OFR algorithm has already become a standard method for the model detection of spatio-temporal systems in the identification process. However, this algorithm cannot be always satisfied, especially when it is applied to some bad training data which usually include uncertainties, or to the case with not very persistently exciting input signal [117, 118].

It has been observed that some model term is nearly always selected as the first term with a very high ERR value which is close to 100%. Consequently, the ERR values of the rest of the model terms are small and thereby are sensitive to the noise [118]. This problem is mainly because of the slowly varying output signal. Assuming that the output signal $y(t)$ is sampled at an oversampling rate, the signal $y(t)$ and the first few linear terms such as $y(t-1), y(t-2), \dots$ will become strongly correlated and cannot be distinguished from each other easily, which means $y(t) \approx y(t-1) \approx y(t-2) \approx \cdots$. Hence, if for example the term $y(t-1)$ is in the initial candidate term set, its ERR values may be big as $ERR \approx 1.0$, so that this term will be firstly selected regardless of whether the term is included in the true model. Thus, there are chances that the OFR algorithm might select incorrect terms. Since the terms of the same cluster are indistinguishable due to the small sampling time, too small sampling time should not be chosen for a practical identification problem [122], which may be true for all identification algorithms.

Noise may be another factor to affect the model structure selection. In some cases, while all correct terms are detected and included in the identified model, some redundant terms may also be selected into the final model even when the training data are sampled with an appropriate sampling rate. This might be caused by using data contaminated by noise. Generally, the identification of nonlinear dynamic systems can be viewed as a structure-unknown problem or a black box problem. Therefore, without any prior knowledge of the true model, the input and output are the only known data which can be used for model term selection.

Owing to this data-oriented characteristic, the aim of all model structure detection algorithms is to find a good approximated model which contains as much information as possible about the observed known data. Hence, if the model is learnt from the training data which is contaminated by noise, chances are that spurious or incorrect terms may be included in the identified model subset. However, a good model structure detection algorithm should try to avoid or reduce the effects of the 'wrong' information from the training data, and the identified model should include all true model terms without any spurious and redundant terms, which thus can capture the main underlying properties involved in the data. Model validation which can provide an independent assessment of the identified model has been often used to assist in the identification.

The above discussion about the performance of the OFR algorithm suggests that it is necessary to improve this classical algorithm so that the correct model structure can be identified even when the training data are not good. Therefore, this motivates the development of the new OFR-MI algorithm assisted by MI criteria, which is described in the following section.

4.4.2 The OFR-MI Algorithm

The Orthogonal Forward Regression using Mutual Information (OFR-MI) algorithm is proposed for the identification of spatio-temporal systems. In this new algorithm, mutual information will be added into the orthogonalisation procedure of the OFR algorithm as a criterion to decide the significance of model terms instead of using the ERRs [107]. According to Eqn.(4.3) the algorithm can be described as follows.

1. Model term selection procedure
   a) *Step 1*. All the model terms $\Phi_1 = \phi_i(t), i = 1, \dots, M$ are candidates for the important term $w_1(t)$. For $i = 1, \dots, M$,

$$w_1^{(i)}(t) = \phi_i(t) \tag{4.17}$$

$$[MI]_1^{(i)}\big(y(t),\phi_i(t)\big)=\sum_{y\in Y}\sum_{\phi_i\in\Phi_1}p(y,\phi_i)\log\left(\frac{p(y,\phi_i)}{p(y)p(\phi_i)}\right)\qquad(4.18)$$

Find the maximum of $[MI]_1^{(i)}$, say, $[MI]_1^{(j)}=\max\left\{\left[[MI]_1^{(i)},\ 1\le i\le M\right]\right\}$. The first significant terms can be $w_1(t)=w_1^{(j)}(t)$, $\phi_j(t)$ is selected with

$$y_1(t)=y(t)-\frac{y(t)w_1(t)}{w_1^2(t)}w_1(t)\qquad(4.19)$$

$$\hat{g}_1=\frac{w_1(t)y(t)}{w_1^2(t)}\qquad(4.20)$$

$$a_{11}=1,MI_1=[MI]_1^{(j)}\qquad(4.21)$$

and the Error-to-Signal Ratio (ESR), which is used as the criterion to terminate the search procedure, is

$$\|r_1\|^2=\frac{\|y_1\|^2}{\|y\|^2}=\frac{\left(\|y\|^2-\frac{(yw_1)^2}{w_1^2}\right)}{\|y\|^2}\qquad(4.22)$$

b) *Step 2*. All the rest of the terms $\Phi_2=\phi_i(t),i=1,\ldots,M,i\ne j$ form the candidate terms for $w_2(t)$. For $i=1,\ldots,M,i\ne j$,

$$w_2^{(i)}(t)=\phi_i(t)-a_{12}^{(\mathrm{i})}w_1(t)\qquad(4.23)$$

where

$$a_{12}^{(i)}=\frac{w_1(t)\phi_i(t)}{w_1^2(t)}\qquad(4.24)$$

$$[MI]_2^{(i)}\big(y_1(t),\phi_i(t)\big)=\sum_{y_1\in Y}\sum_{\phi_i\in\Phi_2}p(y_1,\phi_i)\log\left(\frac{p(y_1,\phi_i)}{p(y_1)p(\phi_i)}\right)\qquad(4.25)$$

Find the maximum of $[MI]_2^{(i)}$, $[MI]_2^{(k)}=\max\left\{[MI]_2^{(i)},\ 1\le i\le M,i\ne j\right\}$. Then the second basis $w_2(t)=w_1^{(k)}(t)$, $\phi_k(t)$ is selected with

$$y_2(t)=y_1(t)-\frac{y_1(t)w_2(t)}{w_2^2(t)}w_2(t)\qquad(4.26)$$

$$\hat{g}_2 = \frac{w_2(t)y(t)}{w_2^2(t)} \tag{4.27}$$

$$a_{22} = 1, a_{12} = a_{12}^{(k)}, MI_2 = [MI]_2^{(k)} \tag{4.28}$$

and the ESR is

$$\|r_2\|^2 = \frac{\|y_2\|^2}{\|y\|^2} = \frac{\left(\|y_1\|^2 - \frac{(y_1 w_2)^2}{w_2^2}\right)}{\|y\|^2} \tag{4.29}$$

c) This procedure is terminated as the M_sth step when either $\left\|r_{M_s}\right\|^2 < \rho$ or $M_s = M$, where ρ is a desired stopping tolerance.

2. Compute the estimated parameters $\hat{\theta}_i$

$$\hat{\theta}_{M_s} = \hat{g}_{M_s} \tag{4.30}$$

$$\hat{\theta}_i = \hat{g}_i - \sum_{k=i+1}^{M_s} a_{ik}\hat{\theta}_k\,, \quad i = M_s - 1, \ldots, 1 \tag{4.31}$$

4.4.3 Model Length Determination

In practice, an identified model from real data can be either overfitting or underfitting, which may cause the model to lack good generalisation properties. Thus, the validation of selected model terms and the final model is important. One of the effective methods to refine the model is cross validation [123-126], a tool that can be used to determine model size. Generalised cross-validation (GCV) is one type of cross validation that is commonly and widely used. The GCV criterion used for linear regression model [127, 128] can be expressed,

$$\mathrm{GCV}(k) = \left(\frac{N}{N-k}\right)^2 \mathrm{MSE}(k) \tag{4.32}$$

where N is the length of the test data set, k is the number of selected model terms and the Mean-Square-Error (MSE) is $\mathrm{MSE}(k) = \|r_k\|^2/N$ corresponding to a model with k terms [107, 129]. GCV will have a minimum value when k is the effective number of model terms [130].

4.5 Simulation Studies

In this section, several identification examples for spatio-temporal systems using both OFR and OFR-MI algorithms will be demonstrated. It is shown that the new OFR-MI algorithm works well for selecting correct terms for spatio-temporal model identification.

4.5.1 CA Example

Consider a one-dimensional 3-site CA model,

$$\begin{aligned} c(j,t+1) = & -2.0c(j-1,t)c(j,t) - 2.0c(j,t)c(j+1,t) + c(j,t) \\ & - 2.0c(j-1,t)c(j+1,t) + c(j-1,t) \\ & + 3.0c(j-1,t)c(j,t)c(j+1,t) + c(j+1,t) \end{aligned} \tag{4.33}$$

This model was simulated with the neighbourhood $\{c(j-1,t), c(j,t), c(j+1,t)\}$ and 100 initial data which are randomly valued 1 or 0. The data for the identification is from the simulation over 100 time steps, so the data length is 100×100. Tables 4.1 and 4.2 show the identification results produced by both OFR and OFR-MI algorithms.

Table 4.1 shows that a constant term is selected by the OFR algorithm with the highest ERR value. However, this term should not be in the model. Table 4.2 shows the results produced by the new OFR-MI algorithm. All the seven selected terms are exactly consistent with the true model terms. In addition, it shows that identified model enables the GCV value to be minimised.

Table 4.1: Identified model structure for the CA model of Eqn.(4.33) using OFR algorithm

Terms	Parameters		ERR(%)	GCV
	True	Estimated		
1	0	-5.4401E-15	35.71	0.6431
$c(j-1,t)c(j,t)$	-2.0	-2.0	8.36	0.5595
$c(j-1,t)$	1.0	1.0	7.6	0.4836
$c(j-1,t)c(j+1,t)$	-2.0	-2.0	2.83	0.4554
$c(j+1,t)$	1.0	1.0	7.0	0.3854
$c(j-1,t)c(j,t)c(j+1,t)$	3.0	3.0	4.82	0.3373
$c(j,t)c(j+1,t)$	-2.0	-2.0	14.18	0.1954
$c(j,t)$	1.0	1.0	19.51	0.0

Table 4.2: Identified model structure for the CA model of Eqn.(4.33) using OFR-MI algorithm

Terms	Parameters		MI	GCV
	True	Estimated		
$c(j-1,t)c(j,t)$	-2.0	-2.0	0.1383	0.3571
$c(j,t)c(j+1,t)$	-2.0	-2.0	0.2114	0.3572
$c(j,t)$	1.0	1.0	0.3108	0.2916
$c(j-1,t)c(j+1,t)$	-2.0	-2.0	1.5207	0.2830
$c(j-1,t)c(j,t)c(j+1,t)$	3.0	3.0	1.6856	0.2696
$c(j-1,t)$	1.0	1.0	0.3008	0.1348
$c(j+1,t)$	1.0	1.0	0.57	1.84E-16

4.5.2 CML Example

A Solé's model which was described in detail in Section 3.3, Chapter 3, was simulated on a lattice of 256×256 with 50 random initial seeds of values between 0.3 and 0.4. With pre-set parameters, this model can be written as

$$x_i(t+1) = 4x_i(t)[1 - x_i(t)]\exp[-5y_i(t)] + 0.001\nabla^2 x_i(t) \qquad (4.34)$$

$$y_i(t+1) = x_i(t)\{1 - \exp[-5y_i(t)]\} + 0.2\nabla^2 y_i(t) \qquad (4.35)$$

The identification was performed using data from eight points at locations (200,192), (200,193), (200,194), (200,195), (200,196), (200,197), (200,198), and (200,199) over 500 time steps. The data length is therefore 8 × 500. The final models identified from the data are detailed in Tables 4.3 and 4.4.

Table 4.3: Identified model structure for the CML model of Eqn.(4.34) and (4.35) using OFR algorithm

Output	Terms	Parameters		ERR	GCV
		True	Estimated	(%)	
$x_i(t)$	$x_i(t-1)[1-x_i(t-1)]$ $\exp[-5y_i(t-1)]$	4.0	4.0	99.999988345	1.1661E-7
	$\nabla^2 x_i$	0.001	0.001	1.1655E-5	0.0
$y_i(t)$	$x_i(t-1)$	1.0	1.0	88.82	0.1119
	$x_i(t-1)\exp[-5y_i(t-1)]$	-1.0	-1.0	10.78	0.004
	$\nabla^2 y_i$	0.2	0.2	0.4	2.2238E-16
	$x_i(t-1)\exp[-5y_i(t-1)]$ $\nabla^2 y_i$	0	3.2513E-14	4.0235E-30	2.2249E-16
	$\exp[-5y_i(t-1)]$	0	6.0457E-17	2.1667E-30	2.2260E-16
	$x_i(t-1)\,\nabla^2 y_i$	0	-1.5438E-14	2.7343E-30	2.2271E-16
	$\exp[-5y_i(t-1)]\nabla^2 y_i$	0	-7.7381E-15	8.4522E-30	2.2283E-16

Table 4.4: Identified model structure for the CML model of Eqn.(4.34) and (4.35) using OFR-MI algorithm

Output	Terms	Parameters		MI	GCV
		True	Estimated		
$x_i(t)$	$x_i(t-1)[1-x_i(t-1)]$ $\exp[-5y_i(t-1)]$	4.0	4.0	7.1866	1.4796E-8
	$\nabla^2 x_i$	0.001	0.001	6.2765	0.0
$y_i(t)$	$x_i(t-1)$	1.0	1.0	2.6799	0.0039
	$x_i(t-1)\exp[-5y_i(t-1)]$	-1.0	-1.0	3.8430	1.3961E-4
	$\nabla^2 y_i$	0.2	0.2	5.4032	8.7180E-18

In Table 4.3, the ERR values for the sub-model of $y_i(t)$ show the first three terms are significant, and GCV reaches a minimum value at the third term indicating that these three terms should be included in the true model. From the selected terms for the sub-model of $x_i(t)$, it is noticed that the first item has a very high ERR value, very close to 1.0, and the other terms therefore have very small ERR values, which suggests that only the first term can describe the true model with a very small and acceptable error. In temporal systems modelling a high initial ERR value is often indicative of an over-sampled data set because adjacent samples then have almost the same amplitude because of the high sampling. This problem exists in the models studied here but reducing the sampling was not found to be an effective solution.

However, the new OFR-MI algorithm can effectively avoid the problem of high initial ERR values. From Table 4.4, the estimated terms are identical to the true model terms.

4.5.3 PDE Example

Computer simulations of a Kawasaki's model expressed by Eqn.(4.36) and (4.37) were applied on the space domain $(0,1)\ \times (0,1)$ and over a lattice with the size of 400 ×400 and no-flux boundary conditions.

$$\frac{\partial v}{\partial t} = \nabla^2 v - wv \tag{4.36}$$

$$\frac{\partial w}{\partial t} = \nabla \cdot \{\alpha v w \nabla w\} + vw \tag{4.37}$$

where $\alpha = \alpha_0(1 + \Delta)$ and α is random and normally distributed with the mean value $\alpha_0 = 4.0$. The initial distribution can be described as below,

$$w_i(0) = w_{i_1,i_2}(0) = 0.71\text{exp}\{-\left({i_1}^2 + {i_2}^2\right)/6.25\}, v_i(0) = 0.35 \tag{4.38}$$

For this model, the identification procedure was applied on the data from the first 1000 successive frames in the simulation. Eight successive points in each frame, located at (200,192), (200,193), (200,194), (200,195), (200,196), (200,197), (200,198), and (200,199), were selected to form the training data set. Therefore, the size of the data set is 8×1000. The results are illustrated in Tables 4.5 and 4.6.

Table 4.5: Identified model structure for the PDE model of Eqn.(4.36) and (4.37) using OFR algorithm

Output	Terms	Parameters		ERR	GCV
		True	Estimated	(%)	
$v(t)$	$v(t-1)$	1.0	1.0	99.96	3.7533E-4
	$w(t-1)v(t-1)$	-0.2	-0.2	2.5720E-2	1.1810E-4
	$\nabla^2 v$	0.2	0.2	1.1804E-2	1.4444E-15
$w(t)$	$w(t-1)$	1.0	1.0	99.99931422	6.8595E-6
	$\alpha w(t-1)v(t-1)$	0	-2.3822E-13	5.2959E-4	1.5627E-6
	$\alpha w(t-1)v(t-1)\nabla^2 w$	2.0	2.0	1.0804E-4	4.8185E-7
	$\nabla(\alpha v w)\nabla w$	2.0	2.0	4.1976E-5	6.1797E-8
	$w(t-1)v(t-1)$	2.0	2.0	6.1735E-6	0.0

Table 4.6: Identified model structure for the PDE model of Eqn.(4.36) and (4.37) using OFR-MI algorithm

Output	Terms	Parameters		MI	GCV
		True	Estimated		
$v(t)$	$\nabla^2 v$	0.2	0.2	3.9504	0.0033
	$v(t-1)$	1.0	1.0	6.2936	1.0237E-6
	$w(t-1)v(t-1)$	-0.2	-0.2	13.3318	4.8288E-19
$w(t)$	$v(t-1)\nabla(\alpha vw)\nabla w$	0	-5.2225E-13	6.388	0.5180
	$\alpha v(t-1)$	0	-1.6653E-16	8.0544	0.5139
	$w(t-1)v(t-1)\nabla^2 w$	0	9.4502E-13	4.3195	0.5094
	$w(t-1)v(t-1)$	2.0	2.0	6.8711	0.4869
	$\alpha w(t-1)v(t-1)\nabla^2 w$	2.0	2.0	3.9573	0.4853
	$w(t-1)$	1.0	1.0	2.6855	2.9212E-8
	$\nabla(\alpha vw)\nabla w$	2.0	2.0	1.2248	0.0

From Table 4.5, the ERRs of the first terms, $v(t-1)$ and $w(t-1)$, for both $v(t)$ and $w(t)$ are close to 1.0 using the OFR algorithm. As noted above this may be caused by the high sampling frequency, so that the output values at the time step $t-1$ are almost identical to the ones at step t. Hence, the terms at $t-1$ time step are selected as the first term every time. If the spurious terms are included in the model, poor estimations may result. However, if the sampling frequency is reduced, the correct models may not be correctly detected. This problem appears to be especially important in spatio-temporal system modelling. The OFR-MI algorithm overcomes these problems and is applicable for spatio-temporal system identification. However, the OFR-MI algorithm cannot always produce better results than the OFR algorithm. For example, the results in Table 4.6, for the model of $v(t)$, all the right terms have been detected. But for the model of $w(t)$, the first three terms are spurious.

4.6 Conclusions

The new OFR-MI algorithm provides an effective model term selection approach for spatio-temporal system identification. In some spatio-temporal system cases spurious terms may be detected using the classical OFR algorithm due to high initial ERR values. This means that the subsequent selection procedure based on ERR values can be affected by the spurious terms. However, by using the new OFR-MI algorithm, this problem can be overcome, because the mutual information is introduced as a criterion for term selection. The contributions of all the regressors to the model are tested using a MI value and regressors with large MI values are selected as significant model terms, this avoids selecting spurious terms successfully. The MI criterion in OFR-MI algorithm works as a replacement of the ERR procedure in the OFR algorithm and the new identification method works well on spatio-temporal models including CA, CML and PDE models. The OFR-MI algorithm is therefore a complementary method for the OFR algorithm, rather than a substitute. This new algorithm has the potential to be developed and to apply to more complicated spatio-temporal models and this and related problems will be studied in later research studies.

Chapter 5

A Probabilistic CA Model Formulation of the Nazim Fatès Model

5.1 Introduction

Chemotaxis, defined as a chemically directed movement, is a common phenomenon for living organisms such as insects, animals (including humans) and microorganisms. Chemotaxis can convey information between species members, which is essential for their survival and reproduction. For example, the male silk moth flies to female moths, because of a chemical called bombykol as a sex attractant for the male, which is produced by the female. Chemotactic behaviours, which are also observed in the aggregation of slime mould, have been widely studied. The movement of single-cell amoebae is directed according to a chemoattractant cAMP in the environment. Unlike the diffusion process, amoebae move up a concentration gradient of cAMP which is secreted from the aggregation centre. In this way, interesting wavelike movements and spatial pattern formation are observed experimentally. Modelling chemotaxis involves employing mathematical expressions to describe chemotactic behaviours and pattern formations.

The first model for the slime mould chemotaxis, which extended the reaction-diffusion model with a convection term, is proposed by Keller and Segel [24, 131]. Based on this model, many models have been developed especially for bacteria colonies [132-135]. These chemotaxis models based on the reaction-diffusion model can be called Reaction-Diffusion-Chemotaxis (RDC) models or Reaction-Diffusion-Advection (RDA) models, which can be capable of investigating the effects of different cell movements on the generation of aggregation patterns and ubiquitous motion waves. The Nazim Fatès model [44] which will be discussed in this chapter is a discrete RDC model and is designed to solve the decentralised gathering problem of slime mould.

The Nazim Fatès model is a spatio-temporal model with discrete time, space and state. It is basically a CA model including several stochastic processes, which can describe the interaction between the reaction-diffusion environment and the action of amoebae. The stochastic processes which are involved in this model make the modelling complicated, especially for the identification. Hence, a simplified Nazim Fatès-type CA model is introduced here to apply to one-dimensional examples with a small number of environmental states. In this way, randomness in the original Nazim Fatès model can be reduced by introducing certain random variables. Inspired from this simplified CA model, a new class of CA models, the probabilistic multi-rule CA model, is proposed and validated with several simulation examples.

5.2 Reaction-Diffusion-Chemotaxis Models

The Reaction-Diffusion-Chemotaxis (RDC) system is one of the most common models for many pattern formation problems in chemistry and biology. These systems are important for computer simulations of chemical and biological processes. In this section, the study of the RDC model mainly focuses on its application to the clustering movement of slime mould at the aggregation stage. A mathematical form of the RDC model will be given, followed by an example of the Nazim Fatès model and simulation studies.

5.2.1 RDC Models

Assuming that there is a concentration gradient of a chemoattractant a_c in the environment, which causes an amoeba movement towards the high concentration, the flux of amoeba cells $\mathcal{J}$ will increase with the density of amoebae u. In this chemotactic case, $\mathcal{J}$ is contributed by both the diffusion flux $\mathcal{J}_\mathrm{d}$ of amoebae themselves and the chemotaxis flux $\mathcal{J}_c$ according to a_c [136]. Hence, $\mathcal{J}$ can be expressed as

$$\mathcal{J} = \mathcal{J}_{\mathrm{d}} + \mathcal{J}_{\mathrm{c}} \tag{5.1}$$

and

$$\mathcal{J}_{\mathrm{d}} = -D\nabla u \tag{5.2}$$

$$\mathcal{J}_{\mathrm{c}} = u\mathrm{X}(a_c)\nabla a_c \tag{5.3}$$

where D is the diffusion coefficient and $\mathrm{X}(a_c)$ represents the chemoattractant concentration. The term $\mathrm{X}(a_c)$ is determined from experiments, so the form of this term varies with different experiments. The general conservation equation for u is

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathcal{J} = h(u) \tag{5.4}$$

Substituting Eqn.(5.1)-(5.3) into Eqn.(5.4) gives

$$\frac{\partial u}{\partial t} + \nabla \cdot (\mathcal{J}_{\mathrm{d}} + \mathcal{J}_{\mathrm{c}}) = h(u)$$

$$\frac{\partial u}{\partial t} + \nabla \cdot (-D\nabla u + u\mathrm{X}(a_c)\nabla a_c) = h(u)$$

$$\frac{\partial u}{\partial t} - \nabla \cdot D\nabla u + \nabla \cdot (u\mathrm{X}(a_c)\nabla a_c) = h(u) \tag{5.5}$$

where $h(u)$ is the growth term for amoebae, Therefore, the general form of the RDC equation can be written as [136],

$$\frac{\partial u}{\partial t} = h(u) - \nabla \cdot (u\mathrm{X}(a_c)\nabla a_c) + D\Delta u \tag{5.6}$$

Here, $D\Delta u$ is the diffusion term which describes random movement of slime mould cells with the positive diffusivity of D. The chemotaxis term $\nabla \cdot (u\mathrm{X}(a_c)\nabla a_c)$ indicates directional movement oriented by the concentration gradient of the chemoattractant a_c. For slime mould, a_c is a chemical which can be generated by amoebae, also has the diffusion capability, so a_c can be commonly written as

$$\frac{\partial a_c}{\partial t} = g(a_c, u) + D_{a_c}\Delta a_c \tag{5.7}$$

where D_{a_c} is the diffusion coefficient of a_c and usually $D_{a_c} > D$. In Keller and Segel's model for slime mould [137], the kinetics term was given by $g(a_c, u) = h_1 u - h_2 a_c$, where $h_1 u$ denotes the production of the chemoattractant which is proportional to the density of amoebae, and $-h_2 a_c$ represents the reduction of a_c. Given there is no production of amoebae during the chemotactic process, so $h(u) = 0$. The term $\mathrm{X}(a_c)$ can be set as a positive constant X_0. Thus the model in a simplified form can be expressed as

$$\frac{\partial u}{\partial t} = D\Delta u - \nabla \cdot (u \mathrm{X}_0 \nabla a_c) \tag{5.8}$$

$$\frac{\partial a_c}{\partial t} = h_1 u - h_2 a_c + D_{a_c}\Delta a_c \tag{5.9}$$

5.2.2 The Nazim Fatès Model

The Nazim Fatès model, which is a bio-inspired model with a RDC-based aggregation scheme, was proposed by Nazim Fatès [44] in 2010. This model was introduced to describe the decentralised gathering phenomena observed in the *Dictyostelium discoideum* cellular slime mould aggregation and to solve a problem where slime mould amoebae are initially randomly distributed on a lattice and then gather to form a compact cluster. According to two main phenomena found at the aggregation phase of slime mould: reaction-diffusion and chemotaxis, the gathering process can be described based on a lattice structure which includes active cells. Owing to the food shortage in the environment, the motion of the single-cell slime mould starts with no prior information of its own position and the positions of others. In order to group up to form a mound, amoebae will communicate with each other by sending the position information or messages which can be relayed by the lattice. During this chemotactic process, amoebae can move to a free and excited lattice cell in the neighbourhood step by step, and the clustering signal is propagated by the state changing of each cell in the lattice. Amoebae can trigger an excitation in the cells on which they are located. The state of cell can also change according to the states of neighbouring cells, which can be viewed as a cellular automata rule if the state is discrete.

Many previous studies have been found to address the similar decentralised gathering problem in different application areas. For example, Sugihara and Suzuki [138] proposed a decentralised algorithm to control a group of mobile robots to form simple shapes such as a circle, a convex polygon or a line segment. Christensen et al. [139] also introduced a simple language of SWARMORPH-script to control morphology growth processes of self-assembling mobile robots. Further theoretical development of the decentralised gathering problem can be found in [140]. Based on some ideas from previous work as mentioned above and inspired by the aggregation mechanism of slime mould, the Nazim Fatès model is designed to present an aggregation scheme where amoebae can cooperate to cluster efficiently and to solve the decentralised gathering problem using fewer states for the message propagation and fewer rules to control the movement of amoebae.

Unlike general continuous RDC models, the Nazim Fatès model is a discrete dynamical system with discrete time, space and state. In this model, Greenberg-Hastings Model (GHM) [47] as a discrete reaction-diffusion model is employed to simulate the environment which relays clustering signals. Meanwhile, using the similar cellular automata approach in [141], the chemotaxis is formulated by assigning a preferred direction to each amoeba according to their neighbouring cells. Therefore, this model consists of three parts: the environment, the amoebae movement and the interaction process [44].

5.2.2.1 The Environment Description

The model is simulated on a two-dimensional discrete lattice $\mathcal{X}^2$, where the state of the cell at position $(i,j) \in \mathcal{X}^2$ and at time step t is denoted as $c(i,j,t)$. Given a set of $\{0,\ldots,N_e\}$ are all the possible states, so $c(i,j,t) \in \{0,\ldots,N_e\}$, the state 0 represents the neutral state, N_e represents the excited state, and other states from

1 to $N_e - 1$ are the refractory states. The neighbourhood of $c(i,j,t)$, which can be in the form of a von Neumann neighbourhood or other exotic neighbourhoods, is denoted as $\mathcal{N}_{c(i,j,t)}$ or $\mathcal{N}_c$. Let $\#\left(\mathcal{N}_{c(i,j,t)}\right)$ be the number of excited cells in the neighbourhood $\mathcal{N}_{c(i,j,t)}$.

The rule for the reaction-diffusion environment can be described as below:

a) A neutral cell (0) changes to an excited (N_e) cell, if one or more than one neighbouring cell is excited, or the cell remains neutral;

b) An excited cell (N_e) reduces its state value by 1 to be a refractory cell with the state of $N_e - 1$ at one step;

c) At each step, the refractory cell decreases its state by 1 till 0 (the neutral state).

The mathematical expression for this GHM-based CA transition rule is

$$c(i,j,t) = \begin{cases} N_e & if\ c(i,j,t-1) = 0\ AND\ \left[\#\left(\mathcal{N}_{c(i,j,t-1)}\right) > 0\ \right]; \\ c(i,j,t-1) - 1 & if\ c(i,j,t-1) \in \{1, \dots, N_e\}; \\ 0 & \text{otherwise.} \end{cases} \tag{5.10}$$

5.2.2.2 The Amoebae Movement

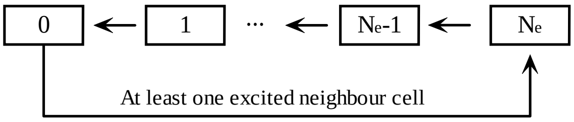
Given a lattice cell can only accommodate a limited number of amoebae at one time, for example the limit is two, so $A_c(i,j,t) \leq 2$, where $A_c(i,j,t)$ is the number of amoebae that a cell $c(i,j,t)$ contains at the time instant t. Consider that the number of amoebae is kept identical at all times, which means no birth or death process is involved in chemotaxis. Amoebae can move randomly to a free excited neighbouring cell which contains fewer amoebae than the limit with a probability p_A, otherwise they stay in the same place.

5.2.2.3 The Interaction Process

Coupling the environment and the amoebae movement is the key step to this RDC model, because it links environmental changes to the motion of amoebae and makes amoebae move as the environment changes. The rule for the interaction is that the excitation of a neutral cell may also come from the amoebae by which the cell is occupied, and the probability of this transition is set as λ. Denote $B(\lambda)$ as the Bernoulli random variable of parameter λ. The interaction transition rule combined to the local environment rule can be represented in Fig.5.1 and can be described mathematically as

$$c(i,j,t)=\begin{cases} N_e & if\ c(i,j,t-1)=0\ AND\ [\#(\mathcal{N}_{c(i,j,t-1)})>0 \\ & \qquad OR\ (A_c(i,j,t-1)\neq 0\ and\ B(\lambda)=1)]; \\ c(i,j,t)-1 & if\ c(i,j,t-1)\in\{1,\dots,N_e\}; \\ 0 & \text{otherwise.} \end{cases}$$

(5.11)

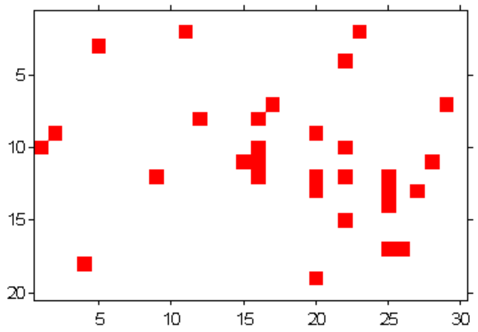


Or with the probability λ for non-empty neutral cells

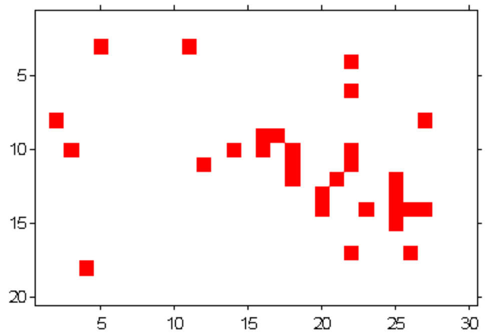
Figure 5.1: The cell state transition rule of the Nazim Fatès model. The arrow expresses the transition from one state to another.

5.2.3 Simulation of the Nazim Fatès Model

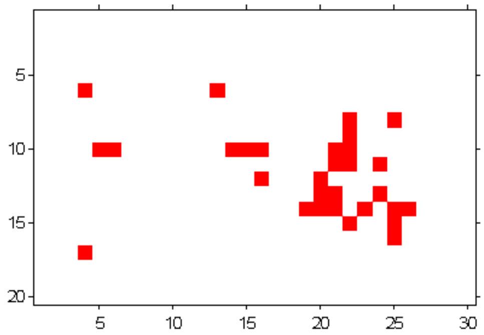
The Nazim Fatès model is simulated on a 20×30 lattice with 30 initially randomly distributed amoebae. The von Neumann neighbourhood and the Moore neighbourhood are applied to two simulation examples respectively with no flux boundaries. Here, let each lattice cell accommodate only one amoeba at a time, so $0\leq A_c(i,j,t)\leq 1$. There are three possible states {0, 1, 2} for the cell, so $N_e=2$. λ is set to be 0.1. The evolutions of the chemotactic gathering are shown in Fig.5.2 and 5.3.



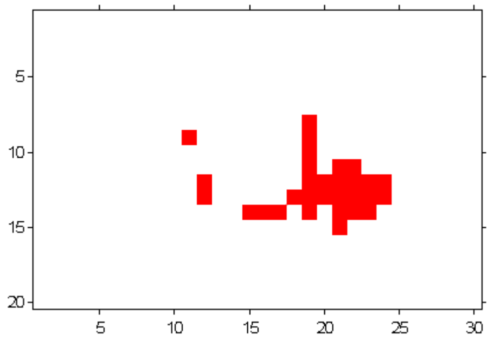
(a) Amoebae t=10



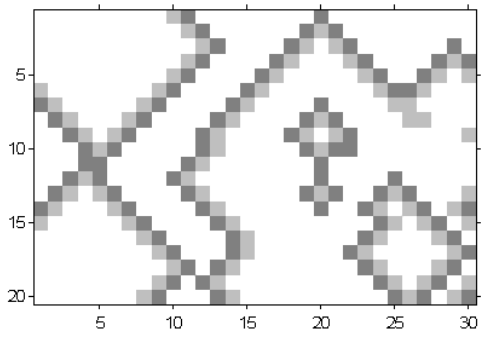
(b) Amoebae t=30



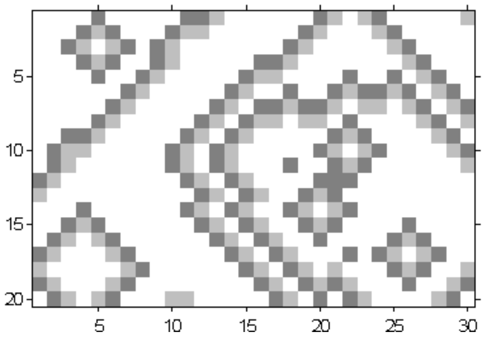
(c) Amoebae t=50



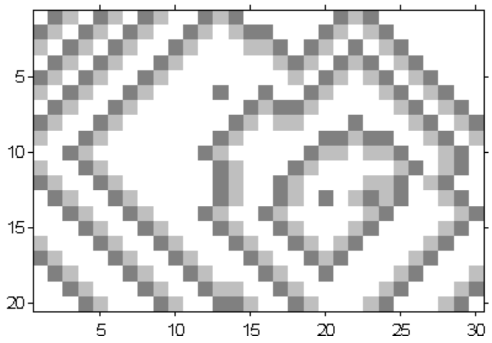
(d) Amoebae t=100



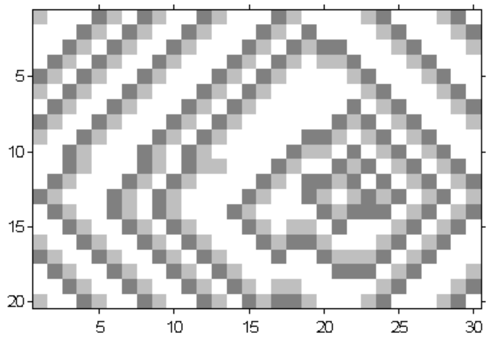
(g) Cells t=10



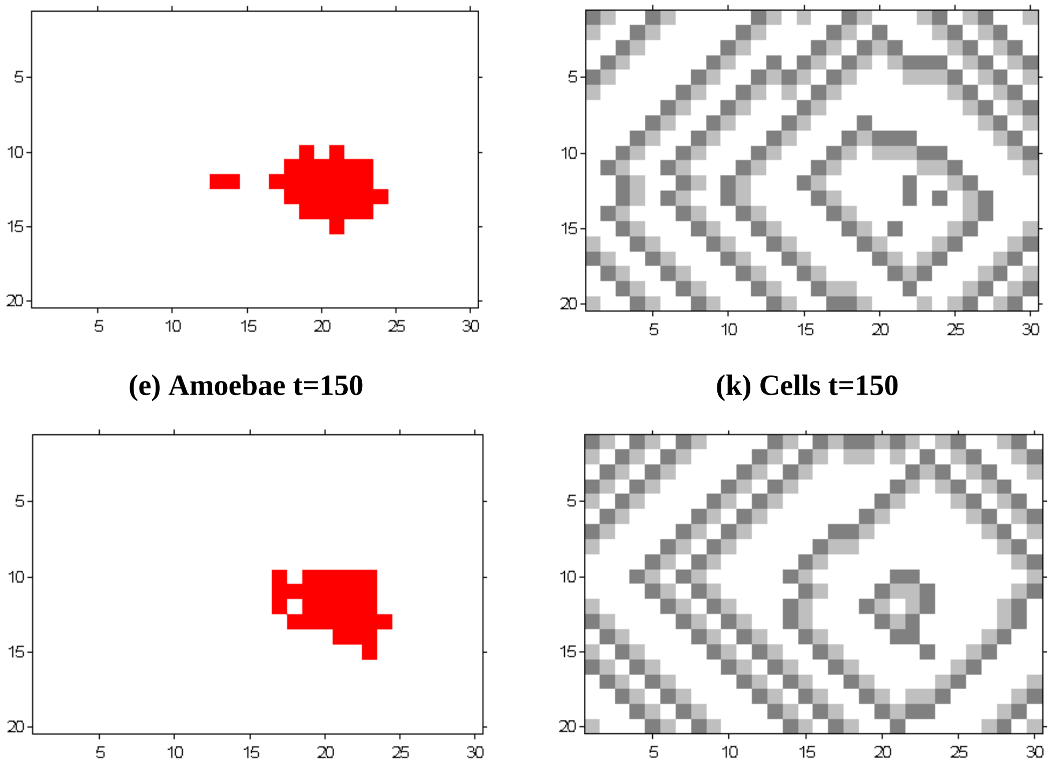
(h) Cells t=30



(i) Cells t=50



(j) Cells t=100



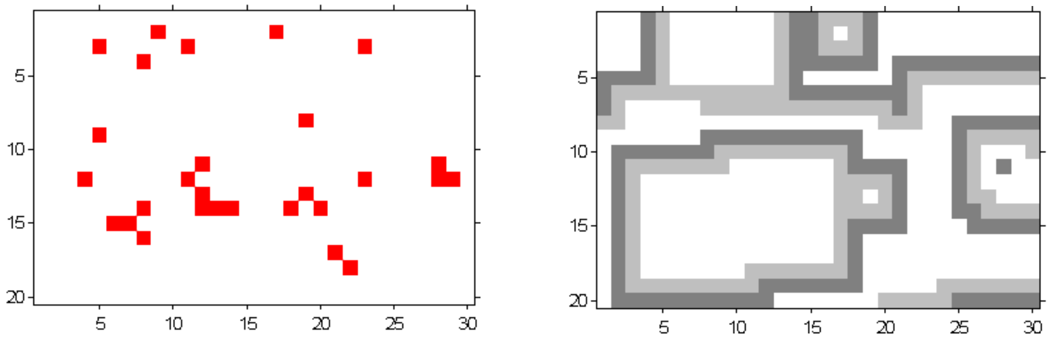
(e) Amoebae t=150

(k) Cells t=150

(f) Amoebae t=250

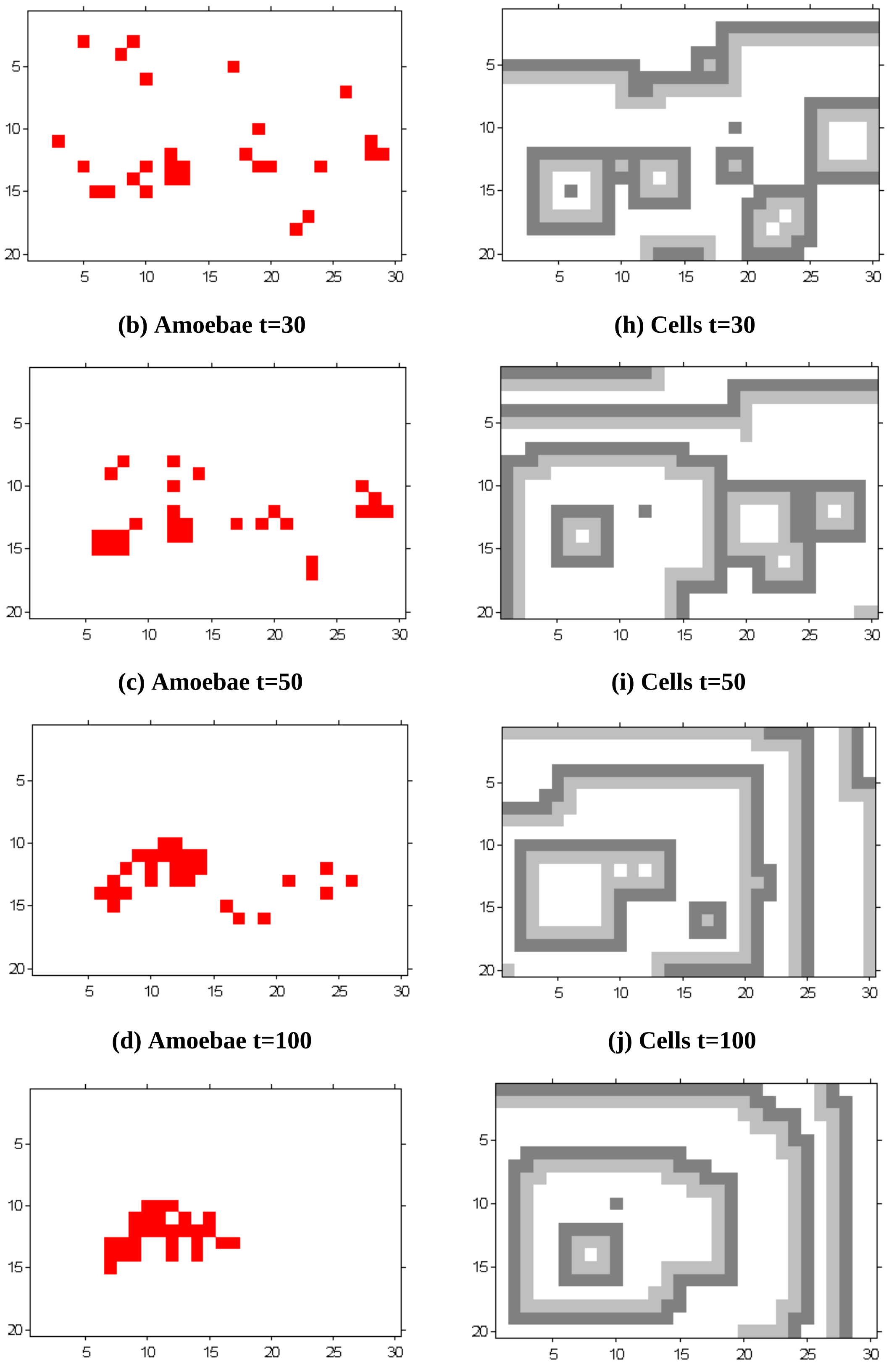
(l) Cells t=250

Figure 5.2: The evolution of the Nazim Fatès model with the 4-site von Neumann neighbourhood. The amoebae distributions at different time steps are shown in (a)-(f), and the corresponding environment states are shown in (g)-(l), where darkest grey cells are excited cells, lighter grey are refractory cells and white are neutral cells.



(a) Amoebae t=10

(g) Cells t=10



(b) Amoebae t=30

(h) Cells t=30

(c) Amoebae t=50

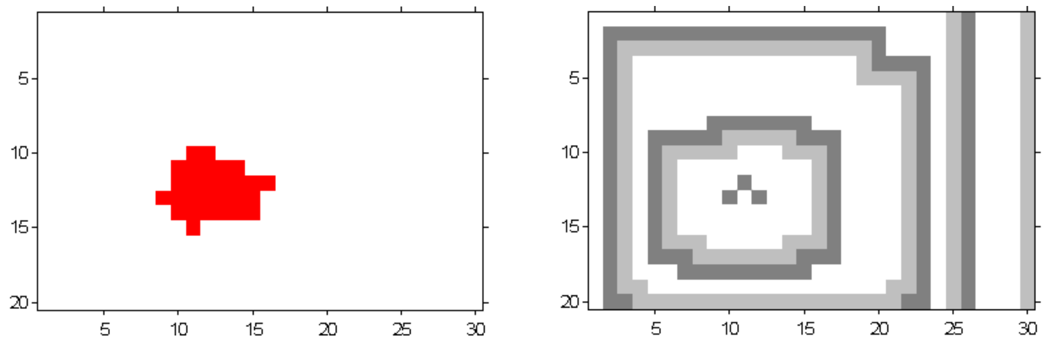
(i) Cells t=50

(d) Amoebae t=100

(j) Cells t=100

(e) Amoebae t=150

(k) Cells t=150



(f) Amoebae t=250 **(l) Cells t=250**

Figure 5.3: The evolution of the Nazim Fatès model with the 8-site Moore neighbourhood. The amoebae distributions at different time steps are shown in (a)-(f), and the corresponding environment states are shown in (g)-(l), where darkest grey cells are excited cells, lighter grey are refractory cells and white are neutral cells.

From Fig.5.2 and 5.3, the simulation results show the gathering process of amoebae which are randomly distributed at first and then gradually group up to form a cluster on a reaction-diffusion lattice.

5.3 A Probabilistic CA Model Formulation

The discrete Nazim Fatès model with a simple rule can simulate the chemotaxis of slime mould. There are two random processes which are involved in the original Nazim Fatès model. One is the current state of a considered cell may change from neutral (0) to excited (N_e) with a probability λ when this cell is occupied by amoebae. The other random process which exists in the movement of the amoebae is that amoebae can move randomly into any free excited neighbouring cell. These random processes can be easy to be achieved in the forward problem, but it makes the reverse problem complicated and difficult to solve. Hence, the identification as the most important part in the reverse problem becomes a challenge, especially for the two-dimensional case. In order to find an effective way to solve this problem, the Nazim Fatès model can be simplified and applied to the one-dimensional case. Therefore, using the Boolean representation a simplified Nazim Fatès model is described in this section. Inspired from this simplified CA model, a new probabilistic multi-rule CA model is introduced with several simulation studies.

5.3.1 A simplified Nazim Fatès-type CA Model

Given there are only two possible states {0, 1} for each environment cell, and 0 represents the neutral state and 1 is the excited. Each lattice cell can accommodate only one amoeba at one time so that $A_c(j,t) \in \{0,1\}$. Therefore, the law for the environment layer of the Nazim Fatès model in one-dimensional examples is simplified as

$$c(j,t) = \begin{cases} 1 & if\ c(j,t-1) = 0\ AND\ [\#\left(\mathcal{N}_{c(j,t-1)}\right) > 0 \\ & \qquad OR\ \ (A_c(j,t-1) = 1\ and\ B(\lambda) = 1)]; \\ & \\ 0 & \text{otherwise.} \end{cases} \tag{5.12}$$

where $c(j,t)$ is the state of the cell at 1-D position j and time step t. A neutral cell becomes an excited cell, if at least one neighbouring cell is excited, or there is a probability (λ) for such transition if the considered cell is occupied. An excited cell turns to the neutral automatically at each time step. This transition rule thus can be described using a Boolean representation as

$$c(j,t) = \left[\delta_{p_1} A_c(j,t-1) \vee c(j-1,t-1) \vee c(j+1,t-1)\right] \wedge \overline{c(j,t-1)} \tag{5.13}$$

where '∨' denotes the *OR* operation and '∧' denotes the *AND* operation. δ_{p_1} is a random variable which obeys a binomial distribution or a Bernoulli distribution as $\delta_{p_1} = 1$ with the probability of λ, and $\delta_{p_1} = 0$ with the probability of $1 - \lambda$. Hence, δ_{p_1} can be expressed as

$$\delta_{p_1} = \begin{cases} 1, & \lambda \\ 0, & 1-\lambda \end{cases} \tag{5.14}$$

For the one-dimensional case, the rule of the amoebae movement can also be written in a Boolean form. Due to only two neighbouring cells of each cell in the one-dimensional lattice, amoebae can randomly move right or left at each time if both neighbouring cells are free and excited, and if only one neighbouring cell is free and excited, amoebae will definitely occupy this one at next time step, otherwise the amoebae will stay at the same place. From the point of view of whether a lattice cell is occupied by amoebae, the rule of the amoebae movement can be described by three processes, that is, a lattice cell changes from occupied to free, from free to occupied, and stays the same state as that at last time step. Hence, this rule can be formulated by two Boolean expressions.

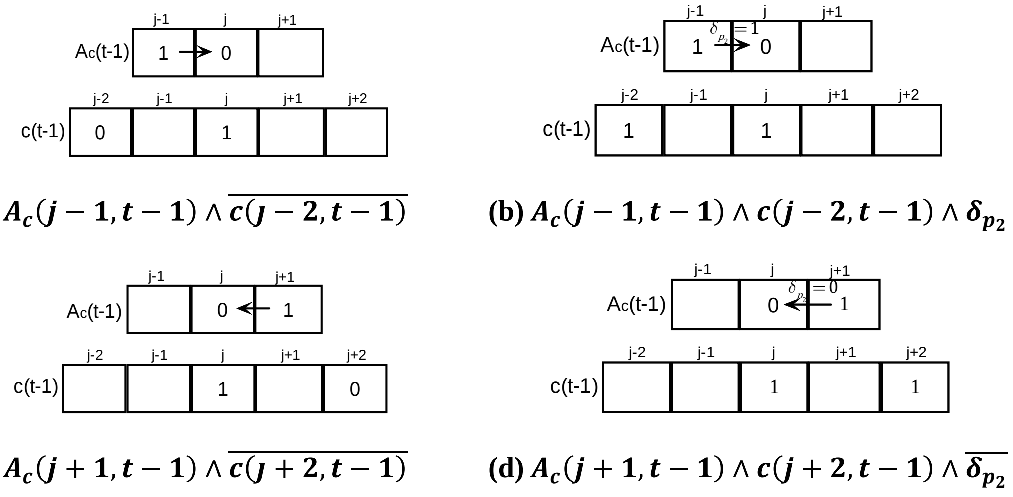
a) An empty and excited cell becomes occupied, or stays free

$$\begin{aligned} A_{c1}(j,t) = \{ & [(A_c(j-1,t-1) \wedge \overline{c(j-2,t-1)}) \vee (A_c(j-1,t-1) \wedge \\ & c(j-2,t-1) \wedge \delta_{p_2})] \vee [(A_c(j+1,t-1) \wedge \overline{c(j+2,t-1)}) \vee \\ & (A_c(j+1,t-1) \wedge c(j+2,t-1) \wedge \overline{\delta_{p_2}})]\} \wedge \overline{A_c(j,t-\iota)} \wedge \\ & c(j,t-1) \end{aligned} \tag{5.15}$$

and

$$\delta_{p_2} = \begin{cases} 1, & p_2 \\ 0, & 1-p_2 \end{cases} \tag{5.16}$$

Here, δ_{p_2} is a random variable which obeys a binomial distribution and can be used to describe the action of amoebae. p_2 is the probability of $\delta_{p_2} = 1$. If $\delta_{p_2} = 1$, amoebae move into the right neighbouring cell, and $\delta_{p_2} = 0$ means amoebae move left. Eqn.(5.15) can also be explained using a diagrammatic representation as shown in Fig.5.4.



(a) $\boldsymbol{A_c(j-1,t-1) \wedge \overline{c(j-2,t-1)}}$ **(b)** $\boldsymbol{A_c(j-1,t-1) \wedge c(j-2,t-1) \wedge \delta_{p_2}}$

(c) $\boldsymbol{A_c(j+1,t-1) \wedge \overline{c(j+2,t-1)}}$ **(d)** $\boldsymbol{A_c(j+1,t-1) \wedge c(j+2,t-1) \wedge \overline{\delta_{p_2}}}$

Figure 5.4: The diagrammatic representation of Eqn.(5.15)

b) An occupied cell becomes free, or keeps occupied

$$A_{c2}(j,t) = \{\overline{[(c(j-1,t-1) \wedge \overline{A_c(j-1,t-1)} \wedge \overline{\delta_{p_2}}) \vee}$$

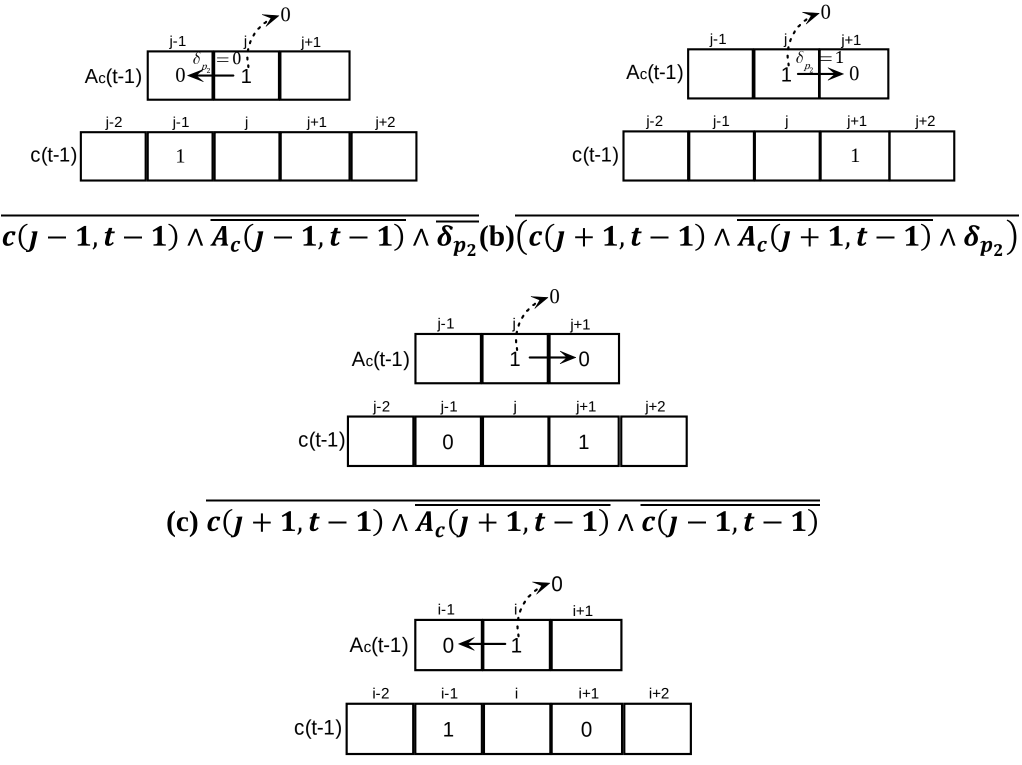
$$\overline{\left(c(j+1,t-1)\wedge\overline{A_c(j+1,t-1)}\wedge\delta_{p_2}\right)}]\vee$$

$$\overline{\left(c(j+1,t-1)\wedge\overline{A_c(j+1,t-1)}\wedge\overline{c(j-1,t-1)}\right)}\vee$$

$$\overline{\left(\overline{c(j+1,t-1)}\wedge c(j-1,t-1)\wedge\overline{A_c(j-1,t-1)}\right)}\}\wedge A_c(j,t-1)$$

(5.17)

The diagrammatic representation for each combination in Eqn.(5.17) is shown in Fig.5.5.



(a)$\overline{\boldsymbol{c(j-1,t-1)}\wedge\overline{\boldsymbol{A_c(j-1,t-1)}}\wedge\overline{\boldsymbol{\delta_{p_2}}}}(b)\overline{\left(\boldsymbol{c(j+1,t-1)}\wedge\overline{\boldsymbol{A_c(j+1,t-1)}}\wedge\boldsymbol{\delta_{p_2}}\right)}$

(c) $\overline{\boldsymbol{c(j+1,t-1)}\wedge\overline{\boldsymbol{A_c(j+1,t-1)}}\wedge\overline{\boldsymbol{c(j-1,t-1)}}}$

(d) $\overline{\overline{\boldsymbol{c(j+1,t-1)}}\wedge\boldsymbol{c(j-1,t-1)}\wedge\overline{\boldsymbol{A_c(j-1,t-1)}}}$

Figure 5.5: The diagrammatic representation of Eqn.(5.17)

c) The distribution of amoebae $A_c(j,t)$ can be described as

$$A_c(j,t)=A_{c_1}(j,t-1)\vee A_{c_2}(j,t-1) \qquad (5.18)$$

Eqn.(5.13) and (5.18) construct the simplified Nazim Fatès model in one dimension, which is a CA model and can be expressed by the Boolean expression. Meanwhile, random processes in the original Nazim Fatès model can be simplified by adding random variables δ_{p_1} and δ_{p_2} in the model. Values of δ_{p_1} and δ_{p_2} may change at every time step according to their probabilities, and different values of δ_{p_1} and δ_{p_2} lead to different mathematical equations for the model. Hence, unlike the classical CA model, the transition rule of this simplified Nazim Fatès model may vary at different time steps, which makes the output at each time step unpredictable. Inspired from the stochastic features of the simplified Nazim Fatès model, a new type of CA models will be created and will be discussed in the following section.

5.3.2 A New Probabilistic Multi-rule Cellular Automata Model

In this section, a new type of CA model, which will be called a Probabilistic Multi-rule Cellular Automata model, is introduced. The probabilistic multi-rule CA model is a CA-based model with random variables, and it can be used to describe spatio-temporal phenomena with random processes. Using classical one-rule CA models, it is difficult to describe random processes, because the classical CA can only output specific and unchanged results at each time step once the transition rule and initialisation are determined. Therefore, if two or more rules with given probabilities respectively can be involved in a CA model and different rules guide the evolution at different time steps, several different outputs can be obtained under the same initial conditions. The probabilities of different outputs depend on the probabilities of different rules in the model. Thus, the evolution can act as a 'random' process using the probabilistic multi-rule CA models. These systems which are different from the deterministic CA with fixed rules can produce more complicated patterns or can be used to describe more complex phenomena.

There are three key elements in the classical CA: a lattice structure, a neighbourhood and a CA transition rule. Generally, the transition function can be described by a finite look-up table. Take a rule of a one-dimensional binary CA with the von Neumann neighbourhood for example. Table 5.1 shows the update outputs corresponding to all the possible states of cells in the neighbourhood. This rule can be defined as $\{r_0, r_1, \dots, r_7\}$, where r_i is the output state of the concerned cell when the states of all its neighbourhood cells are at the i^{th} case as shown in Table 5.1. The rule label can be computed as $\sum_{i=0}^{7} r_i 2^i$, so that this rule is called as *Rule60* = (111100) [142].

Table 5.1: *Rule60* for the one-dimensional 3-site CA.

i	$c(j-1,t-1)$	$c(j,t-1)$	$c(j+1,t-1)$	$c(j,t)$	r_i
0	0	0	0	0	r_0
1	0	0	1	0	r_1
2	0	1	0	1	r_2
3	0	1	1	1	r_3
4	1	0	0	1	r_4
5	1	0	1	1	r_5
6	1	1	0	0	r_6
7	1	1	1	0	r_7

The CA rule in a truth table can also be formulated as a logical function or Boolean function of neighbouring cells [68, 143, 144]. For a one-dimensional CA, the rule can be expressed as

$$c(j,t) = f_b\left(\mathcal{N}_{c(j,t)}|t\right) \tag{5.19}$$

where $\mathcal{N}_{c(j,t)}|t$ includes neighbouring cells at previous times and f_b is a Boolean function. For example, a 3-site CA rule with the neighbourhood set of $\{c(j-1,t-1),\ c(j,t-1),\ c(j+1,t-1)\}$ can be represented by a Boolean equivalent with *AND* (denoted as '$\wedge$') and *XOR* (denoted as '$\oplus$') operators [52] as

$$\begin{aligned} c(j,t) = a_0 \oplus a_1 c(j-1,t-1) \oplus a_2 c(j,t-1) \oplus \ldots \oplus a_7\big(c(j-1,t-1) \wedge \\ c(j,t-1) \wedge c(j+1,t-1)\big) \end{aligned} \tag{5.20}$$

where a_i are binary numbers. The term is included in the model if $a_i = 1$, or it can be removed from the model if $a_i = 0$. Thus the general CA rule with an

n-site neighbourhood $\mathcal{N}_{c(j,t)}|t = \{(c_1|t), (c_2|t), \dots, (c_n|t)\}$ can be written in a Boolean form as

$$c(j,t) = a_0 \oplus a_1(c_1|t) \oplus a_2(c_2|t) \oplus \dots \oplus a_N\big((c_1|t) \wedge \dots \wedge (c_n|t)\big) \qquad (5.21)$$

and $N = 2^n - 1$. $(c|t)$ are states of the cells in the neighbourhood at time steps before t. The rule of Eqn.(5.21) can also be rewritten in a polynomial equation by substituting polynomial representations for logical functions [52], say,

$$\begin{aligned} c(j,t) &= \theta_0 + \theta_1(c_1|t) + \theta_2(c_2|t) + \cdots + \theta_N\big((c_1|t) \cdot \dots \cdot (c_n|t)\big) \\ &= \sum_{i=0}^{N} \theta_i \phi_i(c|t) \end{aligned} \qquad (5.22)$$

where $\phi_i(c|t)$ is the model term from the combination of different variables in the neighbourhood and θ_i is the coefficient for the corresponding model term.

The probabilistic multi-rule CA is defined to share most basic features with the classical CA. The same discrete lattice structure and neighbourhood are included in the probabilistic multi-rule CA model, which, however, has two or more probabilistic transition rules. Using a polynomial expression, the new 1-D probabilistic multi-rule CA model including m rules can be formulated as

$$c_{SCA}(j,t) = \delta_{f^1} f_1 \vee \delta_{f^2} f_2 \vee \cdots \vee \delta_{f^m} f_m \qquad (5.23)$$

here, $c_{SCA}(j,t)$ is the output state. $f_i(i = 1, \dots, m)$ is a basic CA rule, so this model can be rewritten as

$$\begin{aligned} c_{SCA}(j,t) = \delta_{f^1} \sum_{i=0}^{N_1} \theta_{1i}\phi_{1i}(c|t) \vee \delta_{f^2} \sum_{i=0}^{N_2} \theta_{2i}\phi_{2i}(c|t) \vee \dots \\ \vee\, \delta_{f^m} \sum_{i=0}^{N_m} \theta_{mi}\phi_{mi}(c|t) \end{aligned} \qquad (5.24)$$

where $\delta_{f^i} \in \{0,1\}$. According to the probability of each rule p_i and $\sum_{i=1}^{m} p_i = 1$, only one rule is selected out to guide the evolution at each time step, so that in Eqn.(5.24), only one δ is 1 and others are zero at each time step.

$$\sum_{i=1}^{m} \delta_{f^i} = 1 \tag{5.25}$$

N_i is the number of model terms in each rule, which may vary according to different rules. $\phi_i(c|t)$ are model terms of each rule and θ_i are term coefficients.

For example, at a specific time step $t = t_v$ the probabilistic multi-rule CA rule is

$$\begin{aligned} c_{SCA}(j, t_v) = {} & \delta_{f^1} \sum_{i=0}^{N_1} \theta_{1i}\phi_{1i}(c|t_v) + \cdots + \delta_{f^{v-1}} \sum_{i=0}^{N_{v-1}} \theta_{(v-1)i}\phi_{(v-1)i}(c|t_v) \\ & + \delta_{f^v} \sum_{i=0}^{N_v} \theta_{vi}\phi_{vi}(c|t_v) + \delta_{f^{v+1}} \sum_{i=0}^{N_{v+1}} \theta_{(v+1)i}\phi_{(v+1)i}(c|t_v) + \cdots \\ & + \delta_{f^m} \sum_{i=0}^{N_m} \theta_{mi}\phi_{mi}(c|t_v) \end{aligned} \tag{5.26}$$

Given the rule f_v is selected at this time step, so that $\delta_{f^v} = 1$ and $\delta_{f^1} = \cdots = \delta_{f^{v-1}} = \delta_{f^{v+1}} = \cdots = \delta_{f^m} = 0$. Eqn.(5.26) can be rewritten as

$$c_{SCA}(j, t_v) = \sum_{i=0}^{N_v} \theta_{vi}\phi_{vi}(c|t_v) \tag{5.27}$$

According to the probability of each rule, the evolution of a probabilistic multi-rule CA model at every time step may have a different rule. Hence, different output patterns can be produced by the same probabilistic multi-rule CA rule even with the same initialisations. Several examples of simulations will be demonstrated in Section 5.3.3.

5.3.3 Simulation Studies

5.3.3.1 The Probabilistic Multi-rule CA with One-dimensional *Rule27* and *Rule42* on A 3-site Neighbourhood

The spatio-temporal patterns produced by the probabilistic multi-rule CA with one-dimensional *Rule27* and *Rule42* on a 3-site neighbourhood are shown in Fig.5.6. All of them are developed on a 1×100 lattice over 100 time evolution steps from top to bottom, and a periodic boundary condition. These evolutions begin with a same initialisation of a randomly generated binary vector. The neighbourhood of the cell $c(j)$ is $\{c(j-1), c(j), c(j+1)\}$ and the probabilities of *Rule27* and *Rule42* for Fig.5.6 (a)-(d) are {50%, 50%}, {50%, 50%}, {20%, 80%}, and {20%, 80%} respectively.

Although the patterns in Fig.5.6 (a) and (b) are produced from the same probabilistic multi-rule CA rule, the diversity between these two patterns is clear. The same goes for the patterns in Fig.5.6 (c) and (d). It is because that the CA rule at each time step may be different. Fig.5.6 also shows different outputs can be induced by the different probabilities associated with the individual rule in the probabilistic multi-rule CA model.

5.3.3.2 The Probabilistic Multi-rule CA with One-dimensional *Rule110* on Two Different 3-site Neighbourhoods

The spatio-temporal patterns produced by the probabilistic multi-rule CA with one-dimensional *Rule101* on two different 3-site neighbourhoods are shown in Fig.5.7. Here, the probabilistic multi-rule CA rule consists of two rules: *Rule101* on the neighbourhood of $\{c(j-1), c(j), c(j+1)\}$ and *Rule101* on the neighbourhood of $\{c(j-2), c(j-1), c(j)\}$. Fig.5.7 (a) and (b) show the patterns produced from the probabilistic multi-rule CA rule with these two rules with the probabilities of 30% and 70% respectively. The pattern in Fig.5.7 (c) is generated from the same rules with different probabilities as 50% and 50% respectively. Other conditions are the same as that in the examples in Section 5.3.3.1.

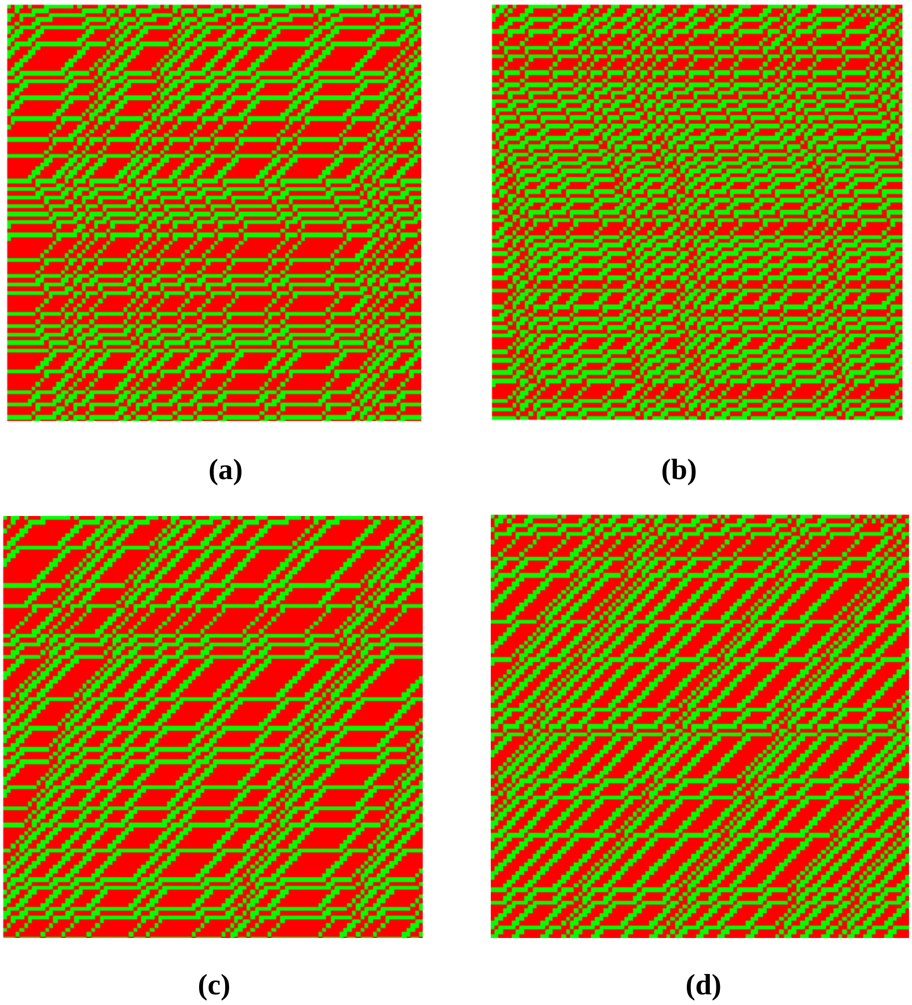


Figure 5.6: Evolutions of the one-dimensional probabilistic multi-rule CA including *Rule27* and *Rule42* with various probabilities of (a){0.5, 0.5}, (b){0.5, 0.5}, (c){0.2, 0.8}, (d){0.2, 0.8} on the same von Neumann neighbourhood.

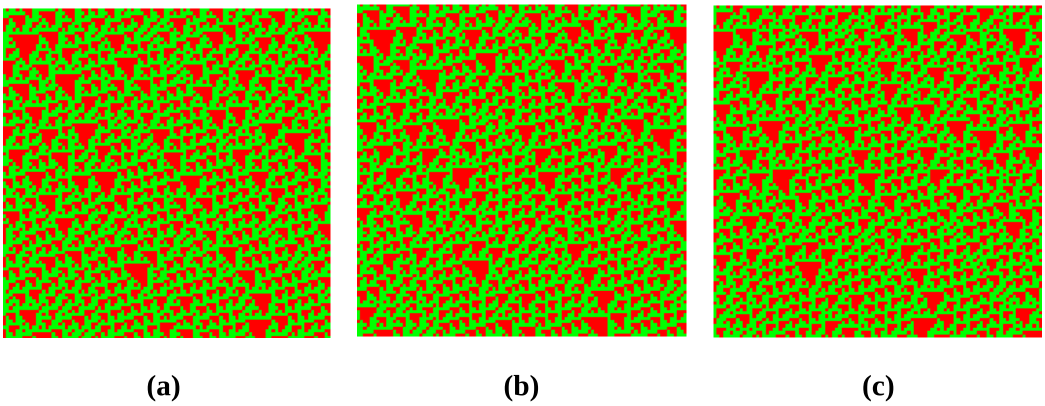


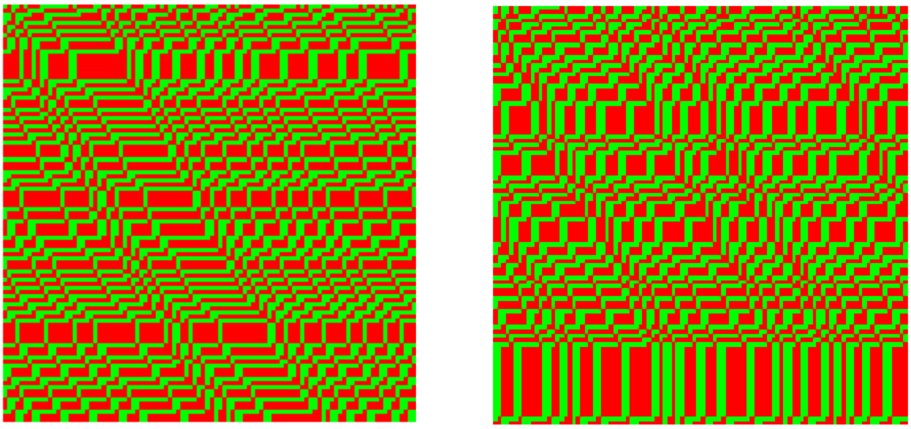
Figure 5.7: Evolutions of the one-dimensional probabilistic multi-rule CA including two sub-rules on two different neighbourhoods respectively. The probabilities of sub-rules are (a){0.3, 0.7}, (b){0.3, 0.7}, (c){0.5, 0.5}.

5.3.3.3 One-dimensional Probabilistic Multi-rule CA with *Rule27* and *Rule42* on Different 3-site Neighbourhoods

Fig.5.8 shows the evolution produced by the one-dimensional probabilistic multi-rule CA with two rules: *Rule27* on the neighbourhood of $\{c(j-1), c(j), c(j+1)\}$ and *Rule42* on the neighbourhood of $\{c(j-2), c(j-1), c(j)\}$. The probabilities of these two sub-rules are set to 40% and 60% respectively. Evolutions in Fig.5.8 (a) and (b) start from the same initial settings as those in the examples in Section 5.3.3.1 and 5.3.3.2.

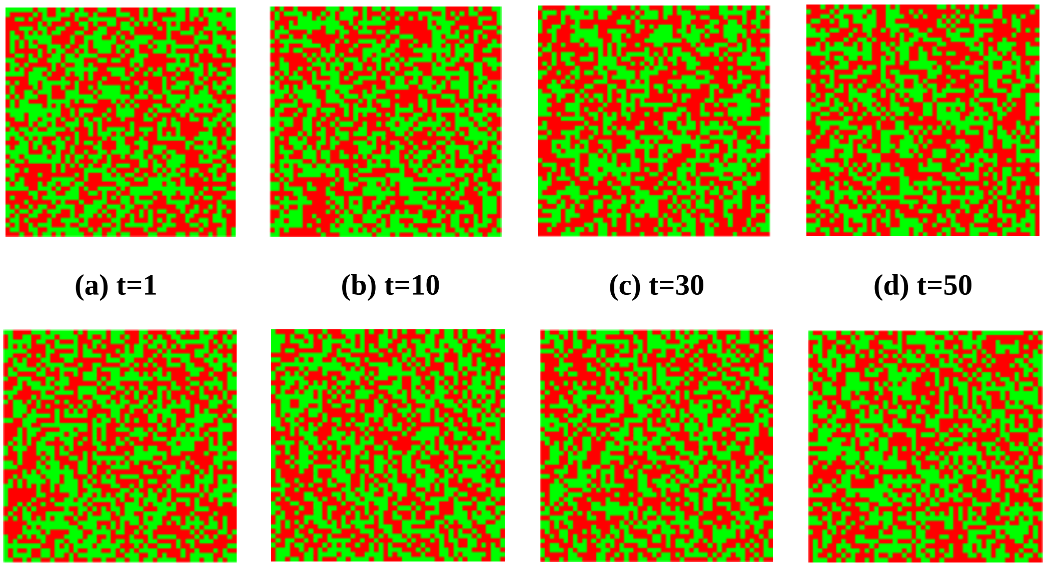
5.3.3.4 Two-dimensional Probabilistic Multi-rule CA

Two-dimensional (2-D) probabilistic multi-rule CA models are more complex than one-dimensional cases, and can produce more complicated patterns. That is because the 2-D probabilistic multi-rule CA model usually includes several 2-D CA rules with 2-D neighbourhoods. There is an example of the 2-D probabilistic multi-rule CA which is developed on the lattice of 50×50 over 100 time steps with a periodic boundary condition. Fig.5.9 (a)-(h) show some snapshots of the evolution at different time steps. Fig.5.9 (a) is the initialisation in which each cell is randomly assigned to 0 or 1. There are two 2-D CA rules in this model. One is *Rule2770298* on the neighbourhood of $\{c(i-1,j), c(i,j-1), c(i,j), c(i,j+1)\}$ with the probability of 40% and the other is *Rule83254* on the neighbourhood of $\{c(i+1,j), c(i,j-1), c(i,j), c(i,j+1)\}$ with the probability of 60%.



(a) (b)

Figure 5.8: Evolutions of the one-dimensional probabilistic multi-rule CA including *Rule27* and *Rule42* on two different neighbourhoods respectively. The probabilities of sub-rules are {0.4, 0.6}.



(a) t=1 (b) t=10 (c) t=30 (d) t=50

(e) t=65 (f) t=80 (g) t=90 (h) t=100

Figure 5.9: Snapshots of the evolution of a 2-D probabilistic multi-rule CA

5.4 Conclusions

The Nazim Fatès model, which is a reaction-diffusion-chemotaxis model with discrete time, space and state, can provide an effective solution to the decentralised gathering problem and has been successfully employed to describe the chemotaxis of slime mould at its aggregation stage. The forward problem of this model can be solved easily, but the reverse problem such as the identification problem is still an open question, because there are some random processes in this spatio-temporal model.

The Nazim Fatès model is then simplified to a one-dimensional CA model which can be represented by Boolean equations. Inspired from this simplified Nazim Fatès-type CA model, a new probabilistic multi-rule CA model is introduced in this chapter. This new type of CA model shares some basic features with the classical CA models but has two or more transition rules with associated probabilities. According these properties, probabilistic multi-rule CA models appear to have the potential to produce many interesting patterns and textures which can be applied to simulate spatio-temporal behaviours.

Chapter 6

Identification of Probabilistic Multi-rule CA Models

6.1 Introduction

Cellular Automata (CA) are spatially extended dynamical systems with discrete space-time and a finite number of states, which are commonly used to describe spatio-temporal dynamics. CA models have been widely studied recently because many spatio-temporal phenomena in the natural world can be described by CA models. Simple CA rules can produce complex dynamical spatio-temporal patterns. Specifically CA’s have been used to model pattern formation such as crystal growth [145] and tumour growth [146]. CA based models have also been used for pattern recognition [147], block ciphers and stream ciphers [148], and for modelling computer networks [149].

Many classes of cellular automata have been proposed and studied, most of which are deterministic such as binary automata, mobile automata and linear hybrid cellular automata [150]. Another class of CAs are probabilistic CA. Probabilistic CAs considered in [69] for example, can be viewed as deterministic CAs corrupted by noise. Other types of CA such as fuzzy CAs, hierarchical CAs, and exotic CAs have been proposed in [151]. Typically the evolution of the CA considered in the literature is described by a single rule at all times.

Most of the literature in this field is concerned with simulation and analysis of different classes of CA models. In contrast, the identification problem has received less attention. The identification of the CA models has been considered in a number of papers [46, 79, 124, 152, 153]. Central to the identification approaches developed in these papers is the orthogonal forward regression algorithm [71, 154], which has been applied successfully for the identification of classical CA models.

A new class of probabilistic multi-rule CA models has been introduced in Chapter 5. This chapter addresses the problem of identifying such probabilistic multi-rule CA models from data. In this case, the existing identification methods developed for the classical CA have to be modified to deal with this class of CAs. This chapter introduces a new method for inferring both the rules and the associated transition probabilities of a probabilistic multi-rule CA model.

6.2 The CA-OLS Algorithm

As shown in [52], a binary CA rule can be formulated as a Boolean function, which in turn can be mapped onto a polynomial equation which has a linear-in-the-parameters model structure. Consider a 2-D n-site neighbourhood CA model for example. Denoting the state of the cell at position (x, y) and time step t by $c(x, y, t)$ and its n-site neighbourhood by $\mathcal{N}\big(c(x, y, t)\big) = \{c_1(t-1), \dots, c_n(t-1)\}$, the polynomial function of $c(x, y, t)$ can be expressed as

$$\begin{aligned} c(x, y, t) &= \theta_0 + \theta_1 c_1(t-1) + \theta_2 c_2(t-1) + \cdots + \theta_N\big(c_1(t-1) \times \\ &\qquad \dots \times c_n(t-1)\big) \\ &= \sum_{i=0}^{N} \theta_i \phi_i\,(t-1) \end{aligned} \tag{6.1}$$

and model terms $\phi_i(t-1)$ can be written as

$$\begin{aligned} &\phi_0(t-1) = 1; \\ &\phi_1(t-1) = c_1(t-1); \\ &\quad \vdots \\ &\phi_N(t-1) = c_1(t-1) \times \dots \times c_n(t-1). \end{aligned} \tag{6.2}$$

where $c(x, y, t) \in \{0,1\}$ and the coefficient $\theta_i \in \mathbb{Z}\,(i = 0, \dots, N, N = 2^n - 1)$. Based on the polynomial form of CA rules, Yang and Billings [52] introduced an identification algorithm called CA-OLS to determine the model structure of the CA model. By applying a modified Gram-Schmidt orthogonalisation procedure, the CA-OLS algorithm can determine which terms are important or which should be included in the CA model. The Error Reduction Ratio (ERR) was used to estimate the contribution which each candidate term makes to the updated output, which means the bigger contribution a term makes, the bigger ERR value it has. The ERR criterion is used to rank the contribution of each candidate model term. The model terms are subsequently then selected in accordance with this ranking. The selection procedure terminates when the mean-square error of the model is less than a pre-specified cut-off value. The cut-off value can be set-to zero for a noise-free CA model.

The algorithm can be summarised as

1) ***Step 1.*** The set of candidate model terms is denoted by $\Phi_1 = \{\phi_i, i = 1,2, \dots, M_C\}$, where M_C is the length of the candidate model term set.

   For $i = 1,2, \dots, M_C$,

$$\mathcal{w}_1^i = \phi_i \tag{6.3}$$

$$\hat{g}_1^i = \frac{\langle c, \mathcal{w}_1^i \rangle}{\langle \mathcal{w}_1^i, \mathcal{w}_1^i \rangle} \tag{6.4}$$

$$[ERR]_1^i = \frac{\langle c, \mathcal{w}_1^i \rangle^2}{\langle c, c \rangle \langle \mathcal{w}_1^i, \mathcal{w}_1^i \rangle} \tag{6.5}$$

   where $\mathcal{w}$ represents orthogonal regressors. c stands for the updated $c(x, y, t)$. $\langle \cdot,\cdot \rangle$ denotes inner product of two vectors. $[ERR]$ denotes the contribution of each term. The term which has the largest ERR value is selected first.

$$l_1 = argmax\{[ERR]_1^i, 0 \le i \le M_C\} \tag{6.6}$$

$$\hat{g}_1 = \hat{g}_1^{l_1} \tag{6.7}$$

$$[ERR]_1 = [ERR]_1^{l_1} \tag{6.8}$$

$$w_1 = w_1^{l_1} \tag{6.9}$$

2) ***Step m***. $m-1$ significant terms have been selected out and have been excluded from the candidate term set at the $(m-1)$th step, so for $i = 1, \dots, M_C, i \neq \{l_1, \dots, l_{m-1}\}$.

$$w_m^i = \phi_i - \sum_{k=1}^{m-1} \left(\frac{\langle \phi_i, w_k \rangle}{\langle w_k, w_k \rangle} w_k \right) \tag{6.10}$$

$$\hat{g}_m^i = \frac{\langle c, w_m^i \rangle}{\langle w_m^i, w_m^i \rangle} \tag{6.11}$$

$$[ERR]_m^i = \frac{\langle c, w_m^i \rangle^2}{\langle c, c \rangle \langle w_m^i, w_m^i \rangle} \tag{6.12}$$

As before, the term with the maximum ERR value is selected by

$$l_m = argmax\{[ERR]_m^i\} \tag{6.13}$$

$$\hat{g}_m = \hat{g}_m^{l_m} \tag{6.14}$$

$$[ERR]_m = [ERR]_m^{l_m} \tag{6.15}$$

The mth orthogonal basis can be selected as

$$w_m = w_m^{l_m} \tag{6.16}$$

and the corresponding significant term $\phi_{l_m}$is selected out from the mth candidate term set.

3) This process is terminated when $1 - \sum_{i=1}^{M_s} [ERR]_i < \rho$ and $M_s < M_c$ or when $M_s = M_c$, where ρ is the cut-off parameter.
4) The estimated parameters are computed by

$$\hat{\theta}_{M_s} = \hat{g}_{M_s} \tag{6.17}$$

$$\hat{\theta}_k = \hat{g}_k - \sum_{i=k+1}^{M_s} \frac{\langle \phi_k, w_i \rangle}{\langle w_i, w_i \rangle} \hat{\theta}_i, \qquad k = M_s - 1, \dots, 1 \tag{6.18}$$

Hence the identified model can be written as

$$c(t) = \sum_{k=1}^{M_s} \theta_k \times \phi_k\,(t-1) \tag{6.19}$$

6.3 A New Identification Algorithm for Probabilistic Multi-rule Cellular Automata

A 2-D probabilistic multi-rule CA model defined in a square lattice with a periodic boundary [155] can be expressed by

$$c_{SCA}(x,y,t) = \delta_{f^1} f_1 \vee \delta_{f^2} f_2 \vee \cdots \vee \delta_{f^m} f_m \qquad (6.20)$$

where $c_{SCA}(x,y,t)$ is the state of the cell at the position (x,y) at time step t. '$\vee$' is the *OR* operator. The m possible rules define the space $\mathbb{F} = \{f_1, \ldots, f_m\}$. It is assumed that the probability of the elementary event $\{f_i(i = 1,2,\ldots,m)\}$ is p_i and that

$$\sum_{i=1}^{m} p_i = 1 \qquad (6.21)$$

The rule f_i is a basic CA rule. Essentially, this means that each iteration step as single rule is selected and that the probability of selecting a particular rule f_i is p_i. In Eqn.(6.20),

$$\delta_f(f) = [\delta_{f^1}(f), \ldots, \delta_{f^m}(f)] \qquad (6.22)$$

denotes a vector random variable associated with each rule f_i such that

$$\delta_f(f_i) = [\delta_{f^1}(f_i), \ldots, \delta_{f^m}(f_i)] \qquad (6.23)$$

and

$$\delta_{f^j}(f_i) = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \qquad (6.24)$$

At each time step the rule is determined by the value of δ_{f^i}, so that if $\delta_f = \delta_f(f_j)$ for $t = t_k$, the state of the cell at t_{k+1} will be given by

$$c_{SCA}(x,y,t_{k+1}) = f_j(t_k) = \sum_{i=0}^{N_k} \theta_{i,j}\phi_{i,j}(t_k) \qquad (6.25)$$

where N_k denotes the number of model terms in the rule $f_j(t_k)$. $\phi_{i,j}$ are model terms for the rule f_j.

For example, if a 1-D probabilistic multi-rule CA model is composed of two CA rules with the probabilities of $p_{Rule\ 1} = 0.6$ and $p_{Rule\ 2} = 0.4$ respectively. The schematic diagram of one possible output over t_n time steps for this probabilistic multi-rule CA model is shown in Figure 6.1.

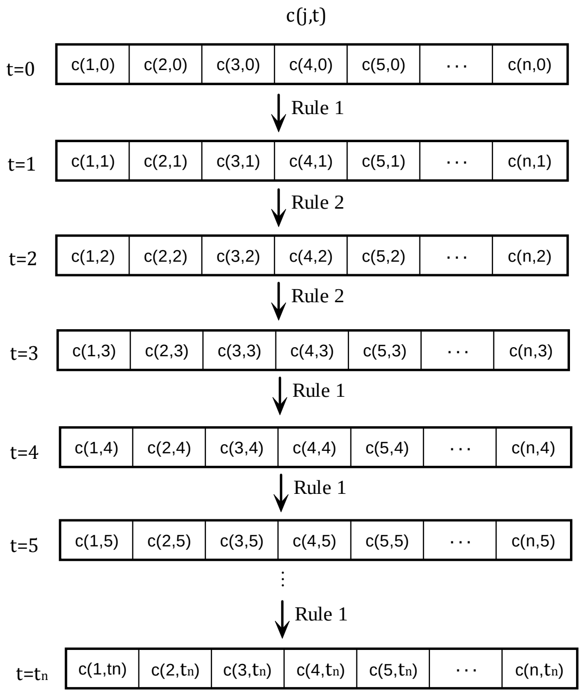

c(j,t)
t=0
c(1,0) c(2,0) c(3,0) c(4,0) c(5,0) ··· c(n,0)
Rule 1
t=1
c(1,1) c(2,1) c(3,1) c(4,1) c(5,1) ··· c(n,1)
Rule 2
t=2
c(1,2) c(2,2) c(3,2) c(4,2) c(5,2) ··· c(n,2)
Rule 2
t=3
c(1,3) c(2,3) c(3,3) c(4,3) c(5,3) ··· c(n,3)
Rule 1
t=4
c(1,4) c(2,4) c(3,4) c(4,4) c(5,4) ··· c(n,4)
Rule 1
t=5
c(1,5) c(2,5) c(3,5) c(4,5) c(5,5) ··· c(n,5)
⋮
Rule 1
t=tn
c(1,tn) c(2,tn) c(3,tn) c(4,tn) c(5,tn) ··· c(n,tn)

Figure 6.1: The schematic diagram of one possible output over t_n time steps for a 1-D probabilistic multi-rule CA model with two CA sub-rules.

The data length at each time step should be long enough to guarantee the correct identified result. Usually 500 iterations are sufficient to estimate the rules and the associated probabilities.

The new identification algorithm for the probabilistic multi-rule CA based on the CA-OLS algorithm can be described as following steps.

a) For any time step $t_k, k = 1, \dots, N - 1$, where N is the number of time samples

Apply the CA-OLS algorithm to determine the rule f_k such that

$$c(x, y, t_{k+1}) = f_k(t_k) = \sum_{i=0}^{N_k} \theta_{ki} \phi_{ki}\,(t_k) \tag{6.26}$$

b) If $k = N$, determine the number m of distinct rules in $\mathbb{F}$ which are involved in the evolution of the probabilistic multi-rule CA model.

Determine the number of occurrences of each event n_i and the probability of each event

$$p_i \approx \frac{n_i}{N}, \qquad i = 1, \dots, m \tag{6.27}$$

6.4 Simulation Studies

The new identification algorithm for probabilistic multi-rule CA models will be tested on one-dimensional and two-dimensional examples. The probabilistic multi-rule CA rules can include several rules which can share the same neighbourhood but different CA rules, or share the same CA rule with different neighbourhoods, or have different rules in different neighbourhoods. Four simulations will be demonstrated in this section.

6.4.1 One-dimensional Probabilistic Multi-rule CA Model

6.4.1.1 Different Rules in the Same Neighbourhood

To begin with a simple example, the simulation and identification were employed on a one-dimensional 3-site probabilistic multi-rule CA model with two different rules. Given the rules in the model are *Rule22* and *Rule54* which are shown in Table 6.1 and 6.2, the probability of each rule is set to 0.5. Both rules are in the von Neumann neighbourhood.

The simulation with a periodic boundary condition started from 500 random data with the value of 0 or 1. The evolution was applied over 100 time steps, which is shown in Fig.6.2.

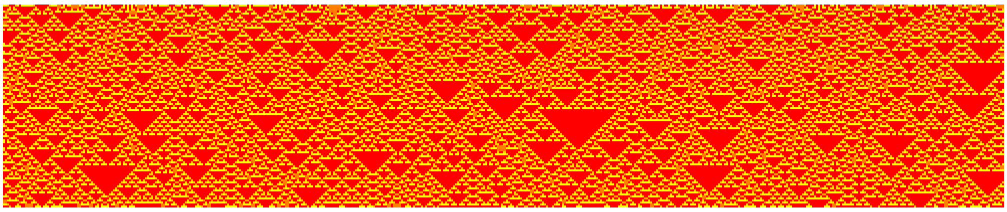


Figure 6.2: 1D probabilistic multi-rule CA model evolution over 100 time steps. The red cells represent 0 and the yellow ones represent 1.

Table 6.1: *Rule 22* for the one-dimensional 3-site probabilistic multi-rule CA example

i	$c(j-1,t-1)$	$c(j,t-1)$	$c(j+1,t-1)$	$c(j,t)$
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

Table 6.2: *Rule 54* for the one-dimensional 3-site probabilistic multi-rule CA example

i	$c(j-1,t-1)$	$c(j,t-1)$	$c(j+1,t-1)$	$c(j,t)$
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	0
7	1	1	1	0

Using the proposed algorithm, two polynomial rules were identified, which are listed in Table 6.3 and 6.4.

Table 6.3: Identified polynomial terms for the *Rule22*

Terms	Parameters	ERR(%)
$c(j-1,t-1)$	1.0	11.98
$c(j-1,t-1)c(j,t-1)$	-2.0	15.24
$c(j,t-1)$	1.0	12.29
$c(j,t-1)c(j+1,t-1)$	-2.0	5.78
$c(j+1,t-1)$	1.0	15.55
$c(j-1,t-1)c(j,t-1)c(j+1,t-1)$	3.0	7.08
$c(j-1,t-1)c(j+1,t-1)$	-2.0	32.08

Table 6.4: Identified polynomial terms for the *Rule54*

Terms	Parameters	ERR(%)
$c(j-1,t-1)$	1.0	22.97
$c(j-1,t-1)c(j,t-1)$	-2.0	25.97
$c(j,t-1)$	1.0	12.45
$c(j,t-1)c(j+1,t-1)$	-2.0	5.79
$c(j+1,t-1)$	1.0	18.06
$c(j-1,t-1)c(j,t-1)c(j+1,t-1)$	2.0	6.44
$c(j-1,t-1)c(j+1,t-1)$	-1.0	8.32

The identified rules can be written as

$$\begin{aligned} f_1(t-1) = {} & c(j-1,t-1) + c(j,t-1) + c(j+1,t-1) \\ & - 2c(j-1,t-1)c(j,t-1) - 2c(j-1,t-1)c(j+1,t-1) \\ & - 2c(j,t-1)c(j+1,t-1) \\ & + 3c(j-1,t-1)c(j,t-1)c(j\ +1,t-1) \end{aligned} \tag{6.28}$$

$$\begin{aligned} f_2(t-1) = {} & c(j-1,t-1) + c(j,t-1) + c(j+1,t-1) \\ & - 2c(j-1,t-1)\,c(j,t-1) - c(j-1,t-1)c(j+1,t-1) \\ & - 2c(j,t-1)c(j+1,t-1) \\ & + 2c(j-1,t-1)c(j,t-1)c(j+1,t-1) \end{aligned} \tag{6.29}$$

The next step is to identify the probabilities of these two distinct rules. By counting how many times each rule occurs over 100 time steps and determining the relative frequency, the probabilities of $Rule22$ in Eqn.(6.28) and $Rule54$ in Eqn.(6.29) are $p_{Rule22} = 0.49$ and $p_{Rule54} = 0.51$. Better estimates for the probabilities of rules can be obtained if the CA evolution is observed over a longer time interval. For example, the same identification algorithm can be employed to over 100×100 time steps.

6.4.1.2 The Same Rules in Different Neighbourhoods

In this example, a simulation was applied on a one-dimensional probabilistic multi-rule CA model with two same rules of *Rule22*, but in different neighbourhoods. One is in the von Neumann neighbourhood and the other is in the exotic 1 neighbourhood. 500 random binary data were used to set as the initialisation with a periodic boundary. The probability of *Rule22* in the von Neumann neighbourhood was set to 0.3 and the other rule was set to 0.7. In Fig.6.3, the pattern was developed from the evolution of this model over 100 time steps.

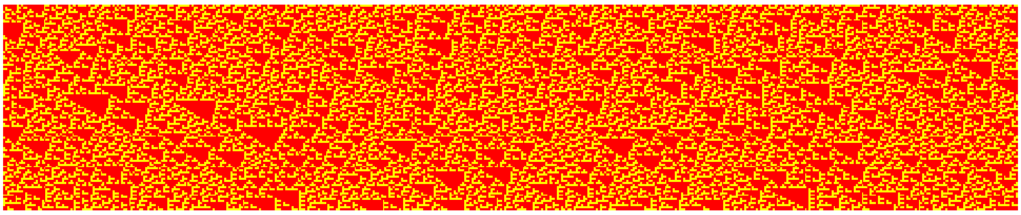


Figure 6.3: The simulation of the 1D probabilistic multi-rule model with two rules in different neighbourhoods over 100 time steps. The red cells represent 0 and the yellow cells represent 1.

Table 6.5: Identified polynomial terms for the *Rule22* in the von Neumann neighbourhood

Terms	Parameters	ERR(%)
$c(j-1,t-1)$	1.0	14.34
$c(j-1,t-1)c(j,t-1)$	-2.0	14.50
$c(j,t-1)$	1.0	9.77
$c(j,t-1)c(j+1,t-1)$	-2.0	6.34
$c(j+1,t-1)$	1.0	17.09
$c(j-1,t-1)c(j,t-1)c(j+1,t-1)$	3.0	6.43
$c(j-1,t-1)c(j+1,t-1)$	-2.0	31.53

Table 6.6: Identified polynomial terms for the *Rule22* in the exotic 1 neighbourhood

Terms	Parameters	ERR(%)
$c(j,t-1)$	1.0	8.56
$c(j-1,t-1)c(j,t-1)$	-2.0	9.28
$c(j-1,t-1)$	1.0	15.08
$c(j-2,t-1)c(j,t-1)$	-2.0	7.87
$c(j-2,t-1)c(j-1,t-1)c(j,t-1)$	3.0	8.54
$c(j-2,t-1)c(j-1,t-1)$	-2.0	16.41
$c(j-2,t-1)$	1.0	34.25

Based on the simulation data in Fig.6.3, two polynomial rules were detected using the proposed identification algorithm, which are given in Table 6.5 and 6.6. Eqn.(6.30) and (6.31) are the polynomial expressions of the rules in Table 6.5 and 6.6 respectively.

$$
\begin{aligned}
f_1(t-1) = {} & c(j-1,t-1) + c(j,t-1) + c(j+1,t-1) \\
& - 2c(j-1,t-1)c(j,t-1) - 2c(j-1,t-1)c(j+1,t-1) \\
& - 2c(j,t-1)c(j+1,t-1) \\
& + 3c(j-1,t-1)c(j,t-1)c(j+1,t-1) \qquad (6.30)
\end{aligned}
$$

$$
\begin{aligned}
f_2(t-1) = {} & c(j-2,t-1) + c(j-1,t-1) + c(j,t-1) \\
& - 2c(j-2,t-1)c(j-1,t-1) - 2c(j-2,t-1)c(j,t-1) \\
& - 2c(j-1,t-1)c(j,t-1) \\
& + 3c(j-2,t-1)c(j-1,t-1)c(j,t-1) \qquad (6.31)
\end{aligned}
$$

As same as the method shown in Section 6.4.1.1, the probability of each rule can be estimated by calculating the frequency of its occurrence over 100 time steps. The estimated probability of *Rule22* in the von Neumann neighbourhood is $p_{von} = 0.28$ and the one of *Rule22* in the exotic 1 neighbourhood is $p_{exotic} = 0.72$.

6.4.1.3 Different Rules in Different Neighbourhoods

The identification algorithm was then tested on a more complicated 1-D model, which contains two different rules in different neighbourhood radiuses respectively. The simulation, shown in Fig.6.4, started from 800 random binary data with the periodic boundary condition. The probability of *Rule22* in the von Neumann neighbourhood is set to 0.4 and *Rule54* in the exotic 1 neighbourhood is 0.6.

In the identification, two rules with different model structures were identified from the data in Fig.6.4. According to the identified polynomial terms in Table 6.7 and 6.8, rules can be described as polynomials in Eqn. (6.32) and (6.33).

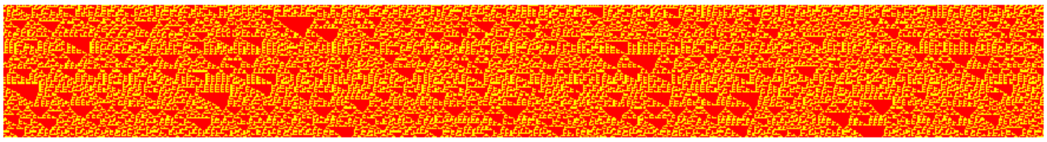


Figure 6.4: The simulation on the 1D CA probabilistic multi-rule model over 100 time steps with two rules: one is *Rule22* in a von Neumann neighbourhood and the other is *Rule54* in an exotic 1 neighbourhood. The red cells represent 0 and the yellow cells represent 1.

Table 6.7: Identified polynomial terms for *Rule22* in the von Neumann neighbourhood

Terms	Parameters	ERR(%)
$c(j+1,t-1)$	1.0	9.08
$c(j,t-1)c(j+1,t-1)$	-2.0	9.08
$c(j,t-1)$	1.0	15.84
$c(j-1,t-1)c(j,t-1)$	-2.0	8.0
$c(j-1,t-1)c(j,t-1)c(j+1,t-1)$	3.0	8.0
$c(j-1,t-1)c(j+1,t-1)$	-2.0	15.93
$c(j-1,t-1)$	1.0	34.08

Table 6.8: Identified polynomial terms for *Rule54* in the exotic 1 neighbourhood

Terms	Parameters	ERR(%)
$c(j-2,t-1)$	1.0	24.63
$c(j-2,t-1)c(j-1,t-1)$	-2.0	24.63
$c(j-1,t-1)$	1.0	12.19
$c(j-1,t-1)c(j,t-1)$	-2.0	6.16
$c(j,t-1)$	1.0	17.88
$c(j-2,t-1)c(j-1,t-1)c(j,t-1)$	2.0	6.16
$c(j-2,t-1)c(j,t-1)$	-1.0	8.36

$$
\begin{aligned}
f_1(t-1) = {} & c(j-1,t-1) + c(j,t-1) + c(j+1,t-1) \\
& - 2c(j-1,t-1)c(j,t-1) - 2c(j-1,t-1)c(j+1,t-1) \\
& - 2c(j,t-1)c(j+1,t-1) \\
& + 3c(j-1,t-1)c(j,t-1)c(j+1,t-1) \qquad (6.32)
\end{aligned}
$$

$$
\begin{aligned}
f_2(t-1) = {} & c(j-2,t-1) + c(j-1,t-1) + c(j,t-1) \\
& - 2c(j-2,t-1)c(j\ -1,t-1) - c(j-2,t-1)c(j,t-1) \\
& - 2c(j-1,t-1)c(j,t-1) \\
& + 2c(j-2,t-1)c(j-1,t-1)c(j,t-1) \qquad (6.33)
\end{aligned}
$$

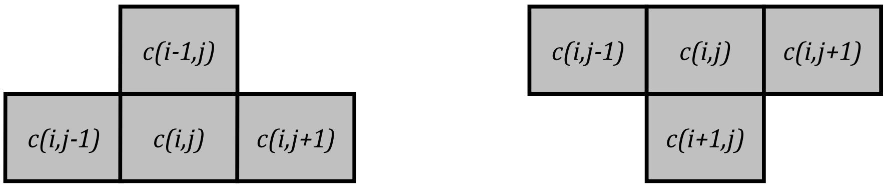
By computing the frequency of its occurrence over 100 time steps, the estimated probability of *Rule22* in the von Neumann neighbourhood is $p_{R_{22^*}} = 0.4$ and the one of *Rule54* in the exotic 1 neighbourhood is $p_{R_{54^*}} = 0.6$.

6.4.2 Two-dimensional Probabilistic Multi-rule CA Model

A 2-D probabilistic multi-rule CA model was simulated on a lattice of 50×50 with a periodic boundary condition. The simulation developed over 100 time steps with the initialisation of 50×50 random binary data. Two rules are included in this model, one is *Rule338314618* with the probability of 0.75 and the neighbourhood shown in Fig.6.5(a) and the other is *Rule1091913014* with the probability of 0.25 and the neighbourhood of Fig.6.5(b).

Two rules were identified from the 100 time step simulation data. The identified polynomial terms are shown in Table 6.9 and 6.10. The models can be written in integer-parameterized polynomials as Eqn. (6.34) and (6.35).

Based on the evolution data over 100 time steps, the estimated probabilities of the rules which are included in this 2D probabilistic multi-rule CA model are $p_{R_{338314618}} = 0.76$ for the *Rule338314618* and $p_{R_{1091913014}} = 0.24$ for the *Rule1091913014.*



(a) **(b)**

Figure 6.5: The neighbourhoods of rules in the 2D probabilistic multi-rule CA model. (a) is for *Rule338314618* and (b) is for *Rule1091913014*

Table 6.9: Identified polynomial terms for *Rule338314618* in the neighbourhood of Fig.6.5(a)

Terms (at the $t-1$ time step)	Parameters	ERR(%)
$c(i,j-1)$	1.0	27.36
$c(i-1,j)c(i,j-1)c(i,j+1)$	2.0	8.82
$c(i,j+1)$	1.0	7.05
$c(i,j-1)c(i,j)c(i,j+1)$	-1.0	6.60
$c(i-1,j)c(i,j+1)$	-2.0	6.81
$c(i-1,j)$	1.0	12.14
$c(i-1,j)c(i,j-1)$	-2.0	11.29
$c(i-1,j)c(i,j-1)c(i,j)$	1.0	13.07
$c(i,j-1)c(i,j+1)$	-1.0	6.84

Table 6.10: Identified polynomial terms for *Rule1091913014* in the neighbourhood of Fig.6.5(b)

Terms (at the $t-1$ time step)	Parameters	ERR(%)
$c(i,j-1)$	1.0	16.10
$c(i,j-1)c(i+1,j)$	-2.0	16.44
$c(i+1,j)$	1.0	13.49
$c(i,j+1)c(i+1,j)$	-2.0	7.10
$c(i,j+1)$	1.0	10.46
$c(i,j-1)c(i,j+1)c(i+1,j)$	2.0	7.25
$c(i,j)c(i,j+1)$	-2.0	2.05
$c(i,j)c(i,j+1)c(i+1,j)$	2.0	2.05
$c(i,j-1)c(i,j)c(i,j+1)$	3.0	1.18
$c(i,j-1)c(i,j)$	-2.0	2.55
$c(i,j)$	1.0	9.28
$c(i,j-1)c(i,j+1)$	-1.0	1.87
$c(i,j-1)c(i,j)\ c(i,j+1)c(i+1,j)$	-3.0	1.80
$c(i,j-1)c(i,j)\ c(i+1,j)$	2.0	3.74
$c(i,j)\ c(i+1,j)$	-1.0	4.64

$$\begin{aligned} f_1(t-1) \ &= c(i,j-1,t-1) + c(i,j+1,t-1) + c(i-1,j,t-1) \\ &\quad - 2c(i-1,j,t-1)c(i,j+1,t-1) \\ &\quad - 2c(i-1,j,t-1)c(i,j-1,t-1) \\ &\quad - c(i,j-1,t-1)c(i,j+1,t-1) \\ &\quad + 2c(i-1,j,t-1)c(i,j-1,t-1)c(i,j+1,t-1) \\ &\quad - c(i,j-1,t-1)c(i,j,t-1)c(i,j+1,t-1) \\ &\quad + c(i-1,j,t-1)c(i,j-1,t-1)c(i,j,t-1) \end{aligned} \tag{6.34}$$

$$\begin{aligned}f_2(t-1) = {} & c(i,j-1,t-1) + c(i+1,j,t-1) + c(i,j+1,t-1) \\ & + c(i,j,t-1) - 2c(i,j-1,t-1)c(i+1,j,t-1) \\ & - 2c(i,j+1,t-1)c(i+1,j,t-1) \\ & - 2c(i,j,t-1)c(i,j+1,t-1) - 2c(i,j-1,t-1)c(i,j,t-1) \\ & - c(i,j-1,t-1)c(i,j+1,t-1) \\ & - c(i,j,t-1)c(i+1,j,t-1) \\ & + 2c(i,j)c(i,j+1,t-1)\ c(i+1,j,t-1) \\ & + 2c(i,j-1,t-1)c(i,j+1,t-1)c(i+1,j,t-1) \\ & + 3c(i,j-1,t-1)c(i,j,t-1)c(i,j+1,t-1) \\ & + 2c(i,j-1,t-1)c(i,j,t-1)\ c(i+1,j,t-1) \\ & - 3c(i,j-1,t-1)c(i,j,t-1)c(i,j+1,t-1)c(i+1,j,t-1)\end{aligned} \tag{6.35}$$

6.5 Conclusions

With the development of CA, CA models have wider and wider applications so that the identification of CA becomes more and more important. The identification is a key component in the CA modelling, but it is still a hard and challenging topic which few investigators have studied.

This chapter proposed a new algorithm for the identification of a probabilistic multi-rule CA based only on observations. This new method, which is based on the CA-OLS algorithm can be used to determine the model structure as well as estimate the probabilities associated with each individual rules. This new algorithm has been demonstrated through numerical simulations involving examples of one- and two-dimensional CAs.

Chapter 7

Probabilistic Multi-rule CA Models with Noise

7.1 Introduction

Noise is an unwanted perturbation for dynamical systems in signal processing. In practice, seldom processes or systems can be free of noise. In the study of modelling natural phenomena, noise cannot be ignored, because it may change, influence or perturb the pattern formation process or may affect the system identification process. In an ecological system for example, noise can be caused by inherent uncertainties such as inhomogeneous distribution of food, or can be caused by human activities including pollution, resource exploitation and land-use change [156]. Noise can disturb the dynamics of an ecosystem making future predictions difficult. It is very important that the effect of noise is taken into account when modelling a dynamical system.

When modelling a discrete-state spatio-temporal system, such as a Cellular Automata (CA), two categories of random perturbations can be considered, static noise and dynamic noise. Generally, CA systems disturbed by noise can be regarded as probabilistic cellular automata (PCA). The forward (simulation) problem of PCAs has been investigated in [151, 157, 158]. The problem of identifying binary PCA from noisy data was addressed in [69].

In this chapter, the probabilistic multi-rule cellular automata corrupted by static and dynamic noise will be discussed respectively. The definitions of these two categories of perturbations are given in Section 7.2. Also, the effect of static and dynamic noise on the probabilistic multi-rule CA pattern formation and identification are compared. Section 7.3 proposes an extension to the identification method for probabilistic multi-rule CA models which were proposed in Chapter 5. Section 7.4 includes the conclusions to this chapter.

7.2 Types of Noise Involved in Probabilistic Multi-rule CA models

A number of quantitative measures will be proposed to characterise the effect of static and dynamic noise on the evolution and identification of probabilistic multi-rule CA systems. Several numerical examples will be used to investigate to what extent static and dynamic noise affect the evolution of probabilistic multi-rule CAs in this section.

7.2.1 Static Noise

In pattern formation, static noise as an external perturbation can be added to the model pattern after the evolution is over, which means static noise is not involved in the evolution of CA patterns. Thus, for a binary CA, the pattern corrupted by static noise can be obtained by randomly flipping a limited number of states in the noise-free pattern according to a specified probability $p_s = n_1/N$, where n_1 is the number of states to be flipped and *N* is the number of cells in the lattice.

A CA system which is corrupted by static noise can be expressed as

$$y(t) = \mathcal{F}(y(t-1), \ldots, y(t-n_y), x(t-1), \ldots, x(t-n_x)) + e_s(t) \quad (7.1)$$

where $\mathcal{F}$ is a CA rule. $y(t)$,$x(t)$ and $e_s(t)$ are system output, input and static noise sequences respectively. A simulation example is given in Fig.7.1, where the transition rule is a probabilistic two-rule CA with one-dimensional *Rule27* and *Rule42* on a 3-site von Neumann neighbourhood. *Rule27* and *Rule42* are shown in Table 7.1and 7.2 respectively.

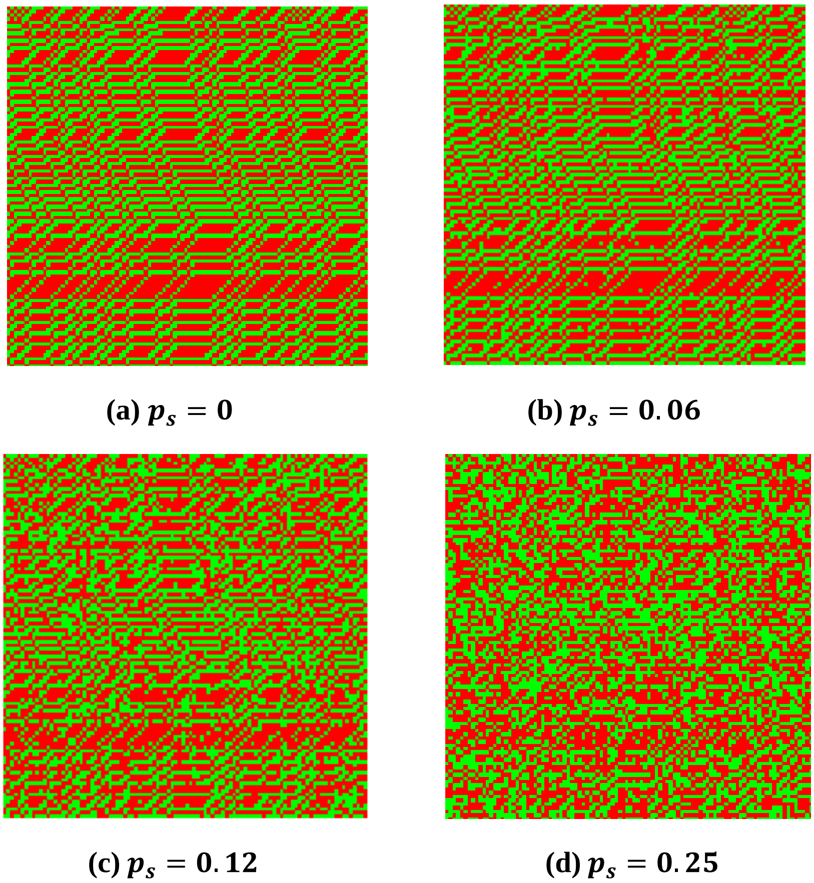
Table 7.1: *Rule 27* for the 1-D 3-site probabilistic multi-rule CA

000	001	010	011	100	101	110	111
1	1	0	1	1	0	0	0

Table 7.2: *Rule 42* for the 1-D 3-site probabilistic multi-rule CA

000	001	010	011	100	101	110	111
0	1	0	1	0	1	0	0

Fig.7.1(a) shows the original noise-free pattern ($p_s = 0$), and Fig.7.1(b)-(d) show the patterns contaminated by static noise with probabilities $p_s = 0.06$, $p_s = 0.12$, and $p_s = 0.25$ respectively. All the patterns in Fig.7.1 were developed on a lattice of 1×100 over 100 time steps with a periodic boundary condition. The system was simulated using the same initial condition of a randomly generated binary vector and was based on the same probabilistic two-rule CA model in which probabilities of *Rule27* and *Rule42* are 50% and 50% respectively.


(a) $p_s = 0$

(b) $p_s = 0.06$

(c) $p_s = 0.12$

(d) $p_s = 0.25$

Figure 7.1: Probabilistic two-rule CA patterns contaminated by various levels of static noise

In Fig.7.1, the red cell represents the state of 0 and the green one is the state of 1, which is the same in all other patterns in following sections.

7.2.2 Dynamic Noise

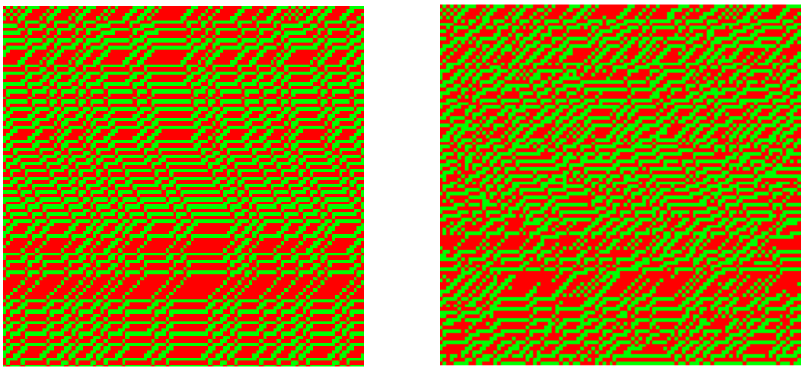
Unlike static noise, dynamic noise is directly involved in the evolution of the pattern formation and is commonly introduced by internal factors, so that dynamic noise needs to be added into the evolution of the pattern formation process. Thus, the state of each cell at current time step is determined by the noisy data at previous time steps. Dynamic noise can be measured in the same way as static noise.

A CA system which is corrupted by static noise can be expressed as

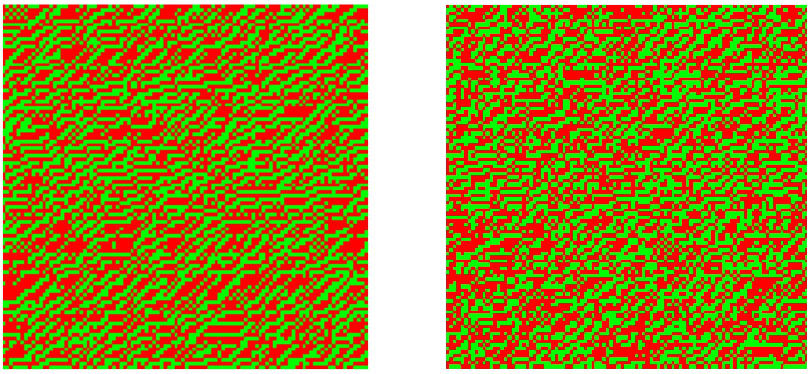
$$y(t) = \mathcal{F}\left(y(t-1), \ldots, y\left(t-n_y\right), x(t-1), \ldots, x(t-n_x), e_d(t-1), \ldots, e_d(t-n_x)\right) \tag{7.2}$$

where $e_d(t)$ are dymanic noise sequences.

Fig.7.2 gives evolutions of the probabilistic 1-D two-rule CA patterns corrupted with several levels of dynamic noise. In order to facilitate comparison of dynamic and static noise, the original noise-free evolution which is shown in Fig.7.2(a) is the same as Fig.7.1(a), and all other initial settings of the evolutions in Fig.7.2 are the same as the example in Fig.7.1 including the lattice size, time steps, periodic boundary condition and initialisation. The same probabilistic CA rule with probabilistic *Rule27* and *Rule42* is also employed here.



(a) $p_d = 0$ **(b) $p_d = 0.06$**



(c) $p_d = 0.12$ **(d) $p_d = 0.25$**

Figure 7.2: Probabilistic two-rule CA patterns contaminated by various levels of dynamic noise

7.2.3 Effects of Noise

For traditional dynamical systems, the signal-to-noise ratio is one of the commonly used measurements for the effect of noise on a designed signal. However, the signal-to-noise cannot be applied to qualify the noise in CA systems, because the magnitude of the noise in the CA system is equal to that of the designed signal, so that the ratio of the signal power to the noise power is always 1, that is, the signal-to-noise ratio is 1. An alternative measure of the noise effect is defined as

$$J_{e1} = \frac{1}{N} \sum_{c \in \mathcal{X}} (c_n - c_o)^2 \tag{7.3}$$

where N is the total number of cells in the lattice $\mathcal{X}$ on which the pattern evolves; c_n and c_o represent states of cells in the noisy and noise-free patterns respectively.

Consider the examples of Fig.7.1(b) and Fig.7.2(b) in Section 7.2.1 and 7.2.2. The errors between the noisy and noise-free patterns are shown in Fig.7.3.

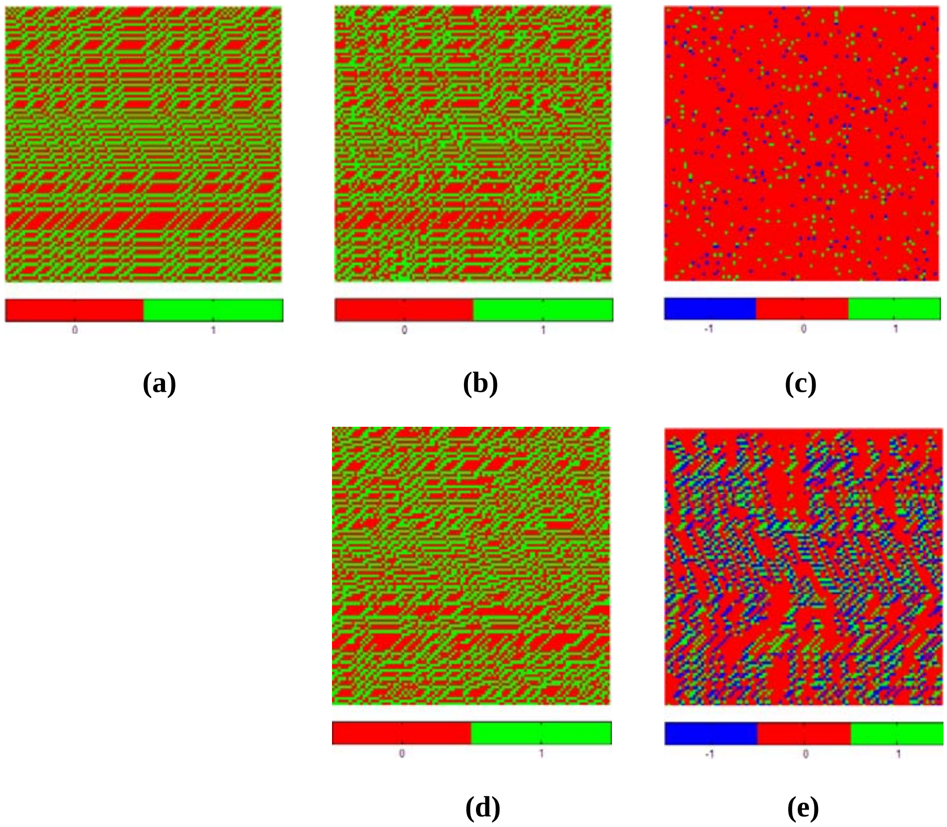


Figure 7.3: Effect measurements of static and dynamic noise in a 1-D probabilistic two-rule CA pattern. (a) the original noise-free pattern (b) the pattern with 6% static noise (c) the error pattern between the statically noisy and noise-free patterns (d) the pattern with 6% dynamic noise (e) the error pattern between the dynamically noisy and noise-free patterns.

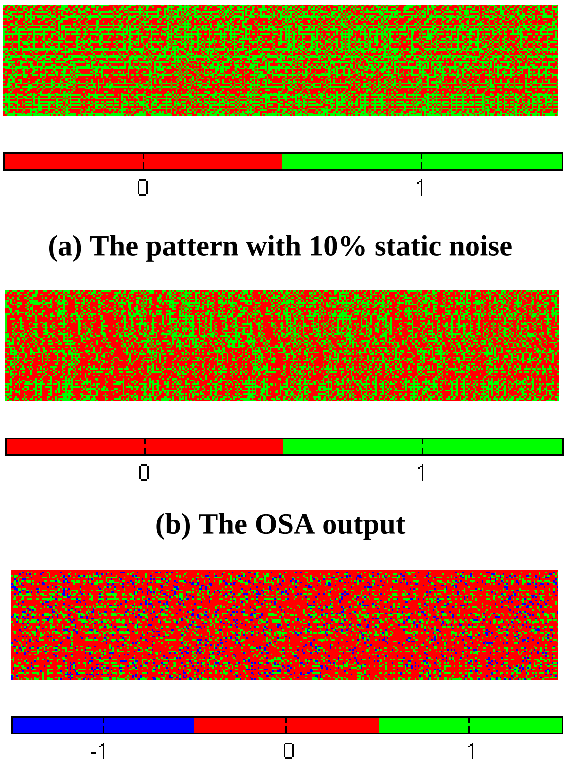
Fig.7.3 shows that the effect of dynamic noise on the evolution of a 1-D probabilistic two-rule CA system is much stronger than that of static noise. In order to quantify the difference between the effects of these two categories of perturbations in Fig.7.3, the number of cells disturbed by noise in the evolution on a 1×100 lattice over 100 time steps can be calculated, that is, 600 for the 6% static noise and 4201 for 6% dynamic noise.

However, negative effect of static noise on the identification process of the underlying rules can be more significant than that of dynamic noise.

The effect of the dynamic noise can be characterised by the following function:

$$J_{e2} = \frac{1}{N}\sum_{c\in\mathcal{X}}(c_n - c_{osa})^2 \qquad (7.4)$$

where N is the total number of cells in the lattice $\mathcal{X}$. c_n and c_{osa} represent states of cells in the noisy pattern and the one-step-head (OSA) output respectively. In the OSA prediction, all the cells are updated by strictly complying with the underlying rule according to the data from the noise corrupted pattern. Take a 1-D probabilistic two-rule CA pattern corrupted by noise for example. The pattern corrupted by 10% static noise, its corresponding OSA output and the error pattern are shown in Fig.7.4. Another example of the pattern corrupted by 10% dynamic noise is shown in Fig.7.5. All the patterns are evolved on a 1×100 lattice over 500 iteration steps from a same original initial vector and periodic boundary condition.

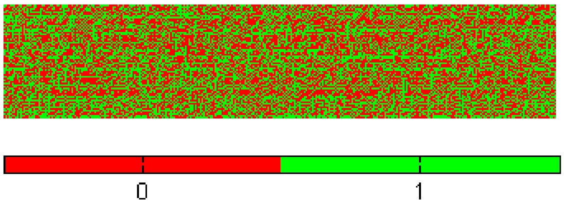


(a) The pattern with 10% static noise

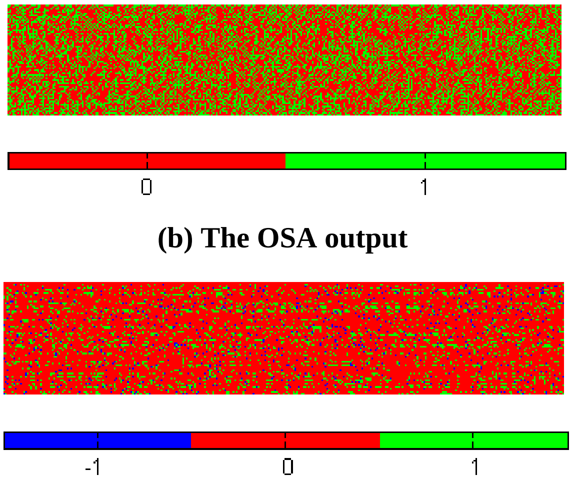
(b) The OSA output

(c) The error pattern between (a) and (b)

Figure 7.4: Effects of 10% static noise on the identification of a probabilistic two-rule CA system.



(a) The pattern with 10% dynamic noise



(b) The OSA output

(c) The error pattern between (a) and (b)

Figure 7.5: Effects of 10% dynamic noise on the identification of a probabilistic two-rule CA system.

Table 7.3: Effect Measurements of dynamic and static noise on the identification

	Number of cells disturbed by noise in the identification	J_{e2}
10% static noise pattern	13066	0.2613
10% dynamic noise pattern	8533	0.1707

The results in Fig.7.4, Fig.7.5 and Table 7.3 show that adding the same amount of static and dynamic noise separately into the noise-free pattern can have different effects on the identification process. Static noise has greater influence on the identification process than dynamic noise. This means compared with dynamic noise, static noise makes the identification of a probabilistic multi-rule CA system more difficult and challenging, because the cells affected by dynamic noise can continue to follow the transition rule instead of providing worse training data to the identification as static noise dose.

7.3 Identification of Probabilistic Multi-rule CA Models with Noise

In Chapter 6, a new identification method for probabilistic multi-rule CA systems was introduced. Based on the CA-OLS algorithm, this new method can effectively determine the rules which are included in the probabilistic multi-rule CA model. Simulation studies in Chapter 6 showed that the method had a good performance on the identification using noise-free data. In this section, this method will be tested on the patterns corrupted by static and dynamic noise respectively.

7.3.1 Identification of CA Patterns Corrupted by Static Noise

In the noise-free case, the identified CA rules in a probabilistic multi-rule CA model can be described by several polynomials, in which the parameter of each model term is integer. Owing to effects of noise, estimated term parameters may be biased and cannot necessary integers. This is because the discrete noise in the CA model may not have expectation zero, or may not be uncorrelated and homoscedastic, which may cause biased parameter estimation using the ordinary least squares (OLS) algorithm according to Gauss-Markov theorem. To deal with this problem, the parameters are rounded up to the nearest integers. However, this method may not be the best, because this method cannot always work fine for any level of noise. Hence, there could be a better solution for mapping the real coefficients to integers in the future work.

In this section, several examples based on a single CA model or a 1-D probabilistic two-rule CA model will demonstrate the identification of different CA models contaminated by static noise.

For the single rule example, a 1-D CA is simulated with a random initialisation and the periodic boundary condition on a 1×500 lattice over 10 time steps. The CA rule is *Rule 22* with the von Neumann neighbourhood, which is shown in Table 7.4. Fig.7.6 shows the patterns corrupted by 0%, 3% and 5% static noise.

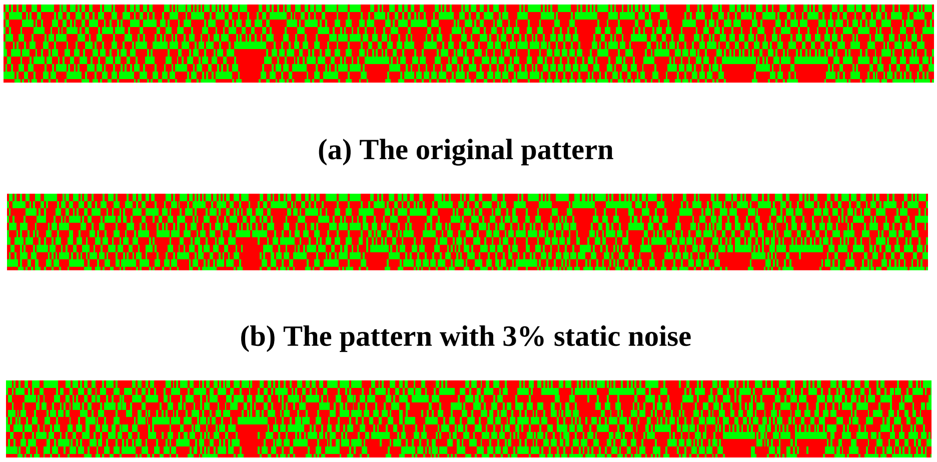
Using the CA-OLS algorithm, identified model terms based on the noisy data in Fig.7.6 are shown in Table 7.6 and 7.7. Table 7.5 shows the list of identified model terms. After the rounding-up process, the identified model can be express by Eqn.(7.5) for the case of 3% noise and Eqn.(7.6) for the 5% noise.

$$\begin{aligned} c(j,t) = {} & c(j-1,t-1) + c(j,t-1) + c(j+1,t-1) \\ & - 2c(j-1,t-1)c(j,t-1) - c(j-1,t-1)c(j+1,t-1) \\ & - 2c(j,t-1)c(j+1,t-1) \\ & + 2c(j-1,t-1)c(j,t-1)c(j+1,t-1) \end{aligned} \tag{7.5}$$

$$\begin{aligned} c(j,t) = {} & c(j-1,t-1) + c(j,t-1) + c(j+1,t-1) \\ & - 2c(j-1,t-1)c(j,t-1) - c(j-1,t-1)c(j+1,t-1) \\ & - 2c(j,t-1)c(j+1,t-1) \\ & + c(j-1,t-1)c(j,t-1)c(j+1,t-1) \end{aligned} \tag{7.6}$$

Table 7.4: *Rule 54* for the 1-D 3-site CA model

000	001	010	011	100	101	110	111
0	1	1	0	1	1	0	0



(a) The original pattern

(b) The pattern with 3% static noise

(c) The pattern with 5% static noise

Figure 7.6: The evolution of 1-D CA pattern corrupted by static noise

Table 7.5: Identified model terms in the identification result

$\boldsymbol{\phi}$	Identified Model Terms
$\boldsymbol{\phi_1}$	$\boldsymbol{c(j-1,t-1)}$
$\boldsymbol{\phi_2}$	$\boldsymbol{c(j,t-1)}$
$\boldsymbol{\phi_3}$	$\boldsymbol{c(j+1,t-1)}$
$\boldsymbol{\phi_4}$	$\boldsymbol{c(j-1,t-1)c(j,t-1)}$
$\boldsymbol{\phi_5}$	$\boldsymbol{c(j-1,t-1)c(j+1,t-1)}$
$\boldsymbol{\phi_6}$	$\boldsymbol{c(j,t-1)\ c(j+1,t-1)}$
$\boldsymbol{\phi_7}$	$\boldsymbol{c(j-1,t-1)c(j,t-1)c(j+1,t-1)}$

Table 7.6: Identified model terms for the 1-D CA corrupted by 3% static noise

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
ϕ	0.9319	0.8727	0.8865	-1.7214	-0.9055	-1.6739	1.6562
Rule54	1	1	1	-2	-1	-2	2

Table 7.7: Identified model terms for the 1-D CA corrupted by 5% static noise

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
ϕ	0.8598	0.8008	0.8642	-1.5389	-0.8771	-1.5290	1.4927
Rule54	1	1	1	-2	-1	-2	2

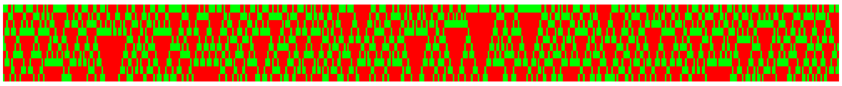
The identified result of Eqn.(7.6) shows when the static noise reaches 5%, it is challenging for CA-OLS algorithm to identify the correct model.

For a 1-D two-rule CA model, the new CA-OLS based identification method proposed in chapter 6 is tested on the statically noisy data. This probabilistic two-rule CA model includes *Rule 22* and *Rule 54* with probabilities of 70% and 30% separately on a 3-site von Neumann neighbourhood. *Rule 22* and *Rule 54* are shown in Table 7.8 and 7.4.

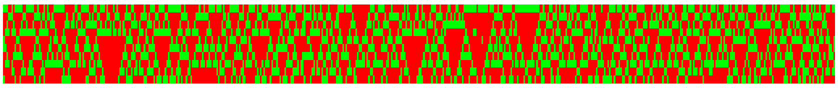
Table 7.8: *Rule 22* for the 1-D 3-site probabilistic two-rule CA

000	001	010	011	100	101	110	111
0	1	1	0	1	0	0	0

With the 1.5% static noise, the simulation is evolved on a 1×500 lattice over 10 iteration steps with a random initial vector and the periodic boundary condition. The evolution results are shown in Fig.7.7, and all rules using at these 10 time steps are shown in Table 7.9.



(a) The original pattern



(b) The pattern with 1.5% static noise

Figure 7.7: The evolution of a probabilistic two-rule CA pattern corrupted by 1.5% static noise

Each rule in this probabilistic two-rule CA model is a basic CA rule which can be expressed as a linear combination of model terms, that is, $f_0(c) \ = \sum_{i=0}^{N} \theta_i \phi_i(c)$, where c is the state of the cell in the lattice, $\phi_i(c)$ are model terms which are constructed by the cells in a neighbourhood, and θ_i are the parameters of terms to be estimated. For this one-dimensional case, the neighbourhood is set as $\{c(j-1,t-1), c(j,t-1), c(j+1,t-1)\}$, where $c(j,t-1)$ represents the state of the cell at the position j at time instant $t-1$. Table 7.10 shows the identification result based on the noisy pattern of Fig.7.7(b).

Table 7.9: CA Rules involved in the evolution of the statically noisy pattern

Time steps	Rules
t=1	*Rule54*
t=2	*Rule22*
t=3	*Rule54*
t=4	*Rule22*
t=5	*Rule22*
t=6	*Rule22*
t=7	*Rule54*
t=8	*Rule22*
t=9	*Rule22*
t=10	*Rule22*

Table 7.10: The identification result based on the pattern with 1.5% static noise

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
t=1	1.0	1.0	0.9857	-1.9848	-1.0061	-1.9857	2.0458
t=2	0.9474	0.9184	0.9737	-1.8273	-1.6811	-1.8344	2.5140
t=3	0.9231	0.9714	1.0	-1.8756	-0.9231	-1.9337	1.8682
t=4	0.9398	0.9394	0.9157	-1.8488	-1.7304	-1.8248	2.6274
t=5	0.8939	0.9211	0.9394	-1.8150	-1.7281	-1.8392	2.6278
t=6	0.9559	0.9167	0.9559	-1.8185	-1.8529	-1.8455	2.6885
t=7	0.9683	0.9706	0.9524	-1.8972	-0.9206	-1.8813	1.8079
t=8	0.9535	0.8923	0.9651	-1.7458	-1.8075	-1.8241	2.5665
t=9	0.9492	0.8696	0.9661	-1.7610	-1.8127	-1.7780	2.5891
t=10	0.9683	0.9444	0.9206	-1.8719	-1.7525	-1.8243	2.6759
Rule22	1	1	1	-2	-2	-2	3
Rule54	1	1	1	-2	-1	-2	2

It is found in Table 7.10 that most estimated parameters are not integers because not all the data in the statically noisy pattern obey the underlying transition rule, which introduces error in the identification result. After rounding all the parameters to the nearest integers, the identified rules can be written as

$$\begin{aligned} c(j,t) = {} & c(j-1,t-1) + c(j,t-1) + c(j+1,t-1) \\ & - 2c(j-1,t-1)c(j,t-1) - 2c(j-1,t-1)c(j+1,t-1) \\ & - 2c(j,t-1)\,c(j+1,t-1) \\ & + 3c(j-1,t-1)c(j,t-1)c(j+1,t-1) \end{aligned}$$

$$\begin{aligned} c(j,t) = {} & c(j-1,t-1) + c(j,t-1) + c(j+1,t-1) \\ & - 2c(j-1,t-1)c(j,t-1) - c(j-1,t-1)c(j+1,t-1) \\ & - 2c(j,t-1)c(j+1,t-1) \\ & + 2c(j-1,t-1)c(j,t-1)c(j+1,t-1) \end{aligned} \tag{7.7}$$

Table 7.11: The identification result based on the pattern with 3% static noise

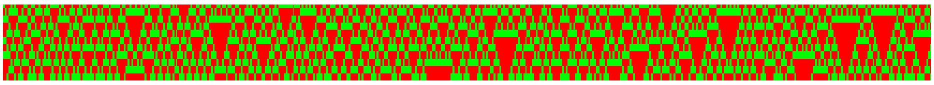
	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
t=1	0.9714	1.0	0.9857	-1.9411	-0.9571	-1.9857	1.9268
t=2	0.9351	1.0	0.9221	-1.8973	-1.6667	-1.8089	2.5592
t=3	0.8281	0.7838	0.8438	-1.5378	-0.7459	-1.5535	1.3977
t=4	0.9024	0.9194	0.9146	-1.7107	-1.6921	-1.7784	2.4448
t=5	0.8571	0.9118	0.8730	-1.7272	-1.5723	-1.7014	2.4215
t=6	0.7826	0.7917	0.8696	-1.4204	-1.3188	-1.5843	1.8797
t=7	0.9394	0.7857	0.8939	-1.6164	-0.9697	-1.6144	1.6467
t=8	0.9186	0.9365	0.9419	-1.7162	-1.5528	-1.6839	2.1559
t=9	0.8621	0.8974	0.9310	-1.6868	-1.7098	-1.7194	2.4879
t=10	0.7812	0.7381	0.8594	-1.3917	-1.4406	-1.5336	2.0517
Rule22	1	1	1	-2	-2	-2	3
Rule54	1	1	1	-2	-1	-2	2

Table 7.11 shows the identification result based on the pattern contaminated by 3% static noise. The rules at time steps t=3,4,5,6,8,9,10 cannot be identified correctly, even after rounding. This shows that increasing static noise will eventually lead to wrong models being identified. Compared to the single CA rule example, the identification algorithm for the 1-D probabilistic two-rule CA is more sensitive to static noise. In order to solve this problem, some special denoising algorithms can be explored to employ on the noisy data before it is used to the identification in further studies.

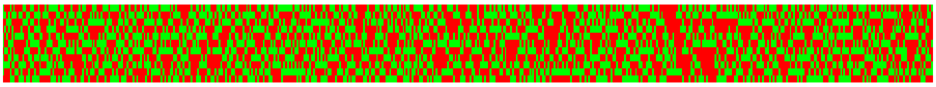
7.3.2 Identification of CA Patterns Corrupted by Dynamic Noise

This section investigates the identification of CA patterns corrupted by dynamic noise. The first two examples are the identification based on 1-D *Rule 54* CA model corrupted by 15% and 20% dynamic noise respectively. The rule of other examples is a 1-D probabilistic two-rule CA model, where the probabilities associated with rules *Rule 22* and *Rule 54* are 40% and 60% respectively. Other settings are the same as those in the examples in Section 7.3.1, including the lattice size, iteration steps, neighbourhood, and boundary conditions.

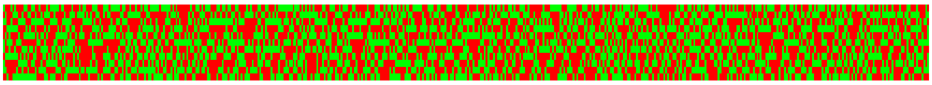
Fig.7.8 shows the patterns which are evolved from the 1-D *Rule 54* CA model with no noise and dynamic noise. Table 7.12 and 7.13 shows the identified model terms based on the dynamically noisy data in Fig.7.8.



(a) The original pattern



(b) The pattern with 15% dynamic noise



(c) The pattern with 20% dynamic noise

Figure 7.8: The evolution of 1-D CA pattern corrupted by dynamic noise

Table 7.12: Identified model terms for the 1-D CA corrupted by 15% dynamic noise

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
ϕ	0.8660	0.8339	0.8468	-1.5810	-0.8759	-1.5392	1.5718
Rule54	1	1	1	-2	-1	-2	2

Table 7.13: Identified model terms for the 1-D CA corrupted by 20% dynamic noise

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
ϕ	0.8121	0.8144	0.8343	-1.4445	-0.8159	-1.4459	1.4379
Rule54	1	1	1	-2	-1	-2	2

The results in Table 7.12 and 7.13 shows CA-OLS algorithm works fine for the single CA rule model with the dynamic noise up to 15%.

Using the same identification method which was employed to the two-rule CA models with static noise in Section 7.3.1, the two-rule CA model corrupted by different levels of dynamic noise can be identified and results are shown in Table 7.15-7.17.

From the evolution data, all the rules for 10 iteration steps can be known and are listed in Table 7.14 for making them comparable with identification results.

Table 7.14: CA Rules involved in the evolution of the dynamically noisy pattern

Time steps	Rules
t=1	*Rule22*
t=2	*Rule54*
t=3	*Rule22*
t=4	*Rule54*
t=5	*Rule54*
t=6	*Rule54*
t=7	*Rule22*
t=8	*Rule22*
t=9	*Rule54*
t=10	*Rule54*

Table 7.15: The identification result based on the pattern with 3% dynamic noise

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
t=1	0.9063	0.9630	0.9844	-1.8375	-1.8906	-1.9315	2.8476
t=2	0.9531	1.0	0.9844	-1.9318	-0.9375	-1.9631	1.8949
t=3	0.9620	1.0	0.9747	-1.9350	-1.9367	-1.9747	2.9360
t=4	0.9464	0.9737	0.9464	-1.8181	-0.8929	-1.8997	1.7441
t=5	0.9600	0.9400	0.9733	-1.9	-1.0	-1.8633	2.0011
t=6	0.9875	0.9474	1.0	-1.9349	-0.9875	-1.9066	1.9248
t=7	0.9429	1.0	0.9857	-1.9044	-1.8941	-1.9473	2.8279
t=8	1.0	0.9565	0.9839	-1.9181	-1.9839	-1.9404	2.9769
t=9	1.0	0.9200	0.9437	-1.8973	-0.9437	-1.8409	1.8426
t=10	0.9730	0.9815	0.9459	-1.9306	-0.9189	-1.8798	1.8540
Rule22	1	1	1	-2	-2	-2	3
Rule54	1	1	1	-2	-1	-2	2

Table 7.16: The identification result based on the pattern with 10% dynamic noise

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
t=1	0.9310	0.8909	0.9655	-1.7324	-1.7716	-1.7818	2.6339
t=2	0.9200	0.8824	0.9333	-1.7062	-0.8890	-1.7195	1.6648
t=3	0.9028	0.8621	0.8889	-1.7072	-1.6601	-1.6356	2.5157
t=4	0.8657	0.9608	0.9254	-1.7688	-0.8466	-1.8092	1.8156
t=5	0.8933	0.9385	0.8667	-1.6829	-0.8141	-1.6987	1.5842
t=6	0.8696	0.8571	0.8551	-1.6553	-0.7802	-1.6586	1.5830
t=7	0.9420	0.8704	0.8986	-1.6948	-1.7572	-1.7101	2.5640
t=8	0.8308	0.9388	0.9538	-1.6495	-1.6964	-1.8726	2.5518
t=9	0.8767	0.8814	0.9041	-1.6444	-0.9142	-1.6718	1.6735
t=10	0.8714	0.9444	0.8714	-1.6677	-0.8218	-1.7233	1.6034
Rule22	1	1	1	-2	-2	-2	3
Rule54	1	1	1	-2	-1	-2	2

The identification result based on the pattern with 3% dynamic noise in Table 7.15 shows that all parameters of selected model terms are consistent with these in the original model after rounding. Compared with the system which is corrupted with 3% static noise, the dynamic noise at the same level has no significant influence on the identification result. In order to evaluate the level of dynamic noise that will affect the identification process, simulations were carried out where the level of dynamic noise was 6%, 8%, 10% and 12%. Simulation results show that the identification based on the data disturbed by no more than 10% dynamic noise can achieve good identification results. Here, only the identification results from the pattern with 10% and 12% dynamic noise are shown in Table 7.16 and 7.17. Examples with 6% and 8% noise lead to the same results with the 10% one. It is also found that the dynamic noise makes the system identification more difficult than that for the single CA rule model.

Table 7.17: The identification result based on the pattern with 12% dynamic noise

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
t=1	0.9474	0.8276	0.8596	-1.6438	-1.6619	-1.5889	2.3652
t=2	0.9333	0.8718	0.8267	-1.7218	-0.8626	-1.6429	1.7306
t=3	0.8235	0.8824	0.8824	-1.6211	-1.5392	-1.6122	2.2752
t=4	0.8356	0.8689	0.9452	-1.5908	-0.8746	-1.6777	1.6330
t=5	0.9155	0.8983	0.9014	-1.7096	-0.8725	-1.6955	1.7106
t=6	0.9275	0.9	0.8551	-1.6875	-0.9116	-1.5951	1.5838
t=7	0.8	0.8889	0.8769	-1.5735	-1.5207	-1.5927	2.1687
t=8	0.9028	0.8333	0.9444	-1.6045	-1.6934	-1.6725	2.3565
t=9	0.8182	0.9184	0.8485	-1.6765	-0.6970	-1.6269	1.5497
t=10	0.8788	0.9388	0.8939	-1.5723	-0.9116	-1.6818	1.5465
Rule22	1	1	1	-2	-2	-2	3
Rule54	1	1	1	-2	-1	-2	2

7.4 Conclusions

This Chapter investigated the effect of static noise and dynamic noise on the identification of probabilistic multi-rule CA. Simulation results showed that while the number of states perturbed by noise is significantly smaller for static noise than for dynamic noise, in identification even relatively modest levels of static noise can prevent the identification of a correct model from data. In contrast, the identification algorithms are more robust to dynamic noise.

The new identification method for the probabilistic multi-rule CA system, which was introduced in Chapter 6, was modified by rounding-off the estimated parameters and tested on examples with static and dynamic noise respectively. Simulation results illustrated that this identification method has a good performance on the data with no more than 1.5% static noise or 10% dynamic noise. These identification results also gave a further explanation of different effects of static and dynamic noise on the identification process.

Chapter 8

Conclusions

The ever-changing world always gives us loads of surprises such as beautiful pattern formation and interesting natural processes which follow the underlying laws of nature. Most phenomena can be viewed as spatio-temporal systems, which represent an essential class of complex dynamical systems with both space and time information. These spatio-temporal phenomena may orientate humans towards the study and understanding of their essence aided by modelling. There are limits to employ universal models to simulate some patterns because of their diversity. Therefore, system identification becomes more and more important in the study of spatio-temporal systems. The identification can provide an effective and potential method to find the intrinsic mechanism behind the complex dynamics from the observed spatio-temporal data.

The aim of the work presented in this thesis is to have a deeper insight into spatio-temporal systems and explore more analysis methods for them. In the mean time, applications of the spatio-temporal system have been extended to model the behaviours of slime mould and the pattern formation during its aggregation process.

8.1 Main Contributions in this thesis

The work of this thesis begins with a brief review of spatio-temporal systems, and then aggregative patterns of slime mould have been simulated using three typical types of spatio-temporal systems. A new model term selection technique based on mutual information (MI) for the identification problem has been developed. In addition, a new type of spatio-temporal system, which is inspired from a probabilistic model for slime mould movement, has been introduced. The corresponding identification method has also been proposed and tested on both noise-free and noisy data. The main contributions of this thesis can be summarised as below.

Slime mould is a ‘smart’ organism and can respond to changing environments intelligently. Hence, interesting patterns such as spirals and concentric circles can often be observed at the aggregation stage in the life cycle of slime mould. Numerical simulations of pattern formation of slime mould have been achieved using three typical spatio-temporal systems, Cellular Automata (CA), Coupled Map Lattices (CML) and Partial Differential Equations (PDE). The Greenberg-Hasting Model (GHM) as a CA model and Solé’s model as a CML model, which were used to describe excitable media and the dynamics in ecosystems respectively, have first been extended to simulate pattern formation and evolutions of slime mould. It has been found that these three models have their respective pros and cons. For the CA model, numerical simulations can easily be realised due to the relatively simplicity of the rule. However, some important information may not be easily included in the model. For example, GHM can produce similar biological patterns of slime mould with no chemoattractant information which is an essential element in experiments. Thus these models may have limits to describe more complex dynamics. For CML and PDE models with continuous states, they can provide better physical explanations corresponding to real systems, but they may make the computer simulation more complicated. Simulation results have proven that CA, CML and PDE models all have capabilities of modelling slime mould dynamics in the aggregation phase.

The reverse problem of pattern formation or system identification is one of the key concerns in the study of spatio-temporal systems, which is still an open question. The aim of spatio-temporal system identification is to determine a mathematical model from observed experimental data. The orthogonal least square (OLS) or the orthogonal forward regression (OFR) has been proved to be one of the most effective algorithms for spatio-temporal system identification. However, using the classical OFR algorithm applied to spatio-temporal systems spurious terms may be detected due to high initial Error Reduction Ratio (ERR) values of these terms in some cases. In order to solve this problem, a new Orthogonal Forward Regression using Mutual Information (OFR-MI) algorithm based on the MI for spatio-temporal system identification was first proposed in this thesis. The MI was introduced as a criterion for measuring the contributions of all the regressors in the orthogonal process where the criterion is the ERR in the OFR algorithm. This new method can effectively avoid selecting spurious terms and work as a complementary method for the OFR algorithm. The method has been successfully tested on various spatio-temporal models including CA, CML and PDE models.

Following a study on a discrete reaction-diffusion-chemotaxis model-the Nazim Fatès model which can be used to describe the chemotactic behaviour of slime mould in its aggregation period, a new class of CA models which are called probabilistic multi-rule CA models was introduced in this thesis. This new CA model is different from the classical one-rule CA in including two or more basic CA transition rules with associated probabilities. Random variables in the probabilistic multi-rule CA model make it possible to simulate many interesting patterns and textures which are produced by some random processes and transitional phenomena in nature.

The forward problem of the probabilistic multi-rule CA system can be solved easily, but the identification of this new CA model directly from an observed pattern is a challenge. That is because most classical identification methods for the one-rule CA model such as the CA-OLS algorithm cannot be directly applied to the probabilistic multi-rule CA model with random variables and a multi-rule structure. Hence, a new identification algorithm for this new CA model was proposed in this thesis. Based on the CA-OLS algorithm and statistical methods, the new method is able to identify the model structure and the probability of each sub-rule which is involved in the probabilistic multi-rule CA model. Simulation studies show that this new identification algorithm worked well on one-dimension and two-dimension examples.

In this thesis, noise in the probabilistic multi-rule CA system was studied. Two types of spatio-temporal noise, static and dynamic noise, were discussed from the perspective of the description and effects on the probabilistic multi-rule CA model. Unlike the signal-to-noise-ratio in the traditional temporal system, the noise level in the probabilistic multi-rule CA system can be described as the proportion of cells to be flipped in the evolution lattice due to noise. Two methods for measuring noise effects were designed. From the simulation result, an important conclusion was obtained, that is, the effect of static noise on the evolution process is less than that of dynamic noise at the same noise-level, but from the point of view of the identification static noise makes the identification process more challenged than dynamic noise.

The new identification method for the probabilistic multi-rule CA model was, for the first time, tested on the patterns corrupted by static and dynamic noise respectively. The simulation result showed that this identification method has a good performance on both noise-free and noisy data.

8.2 Discussion and Suggestions for Future Research

Although there has been an increasing interest devoted to the study of spatio-temporal systems, it is appropriate to say that the current research in this area is still in its infancy because of the diversity and complexity of real applications related to spatio-temporal systems. Based on the works which have been done in this thesis, the following topics may be worth considering in future research.

a) Most observed natural phenomena have high complex spatio-temporal dynamics. Take slime mould for example, after the aggregation phase of its life cycle, slime mould will pile up to form a mound structure and then become a slug which moves chemotactically. One-dimensional and two-dimensional spatio-temporal models cannot be competent for the task of modelling the dynamics which is observed in the mound and slug stage of slime mould. Hence, 3D spatio-temporal systems can be considered to capture the underlying dynamics of these processes. The identification of 3D spatio-temporal systems may become more difficult because more complex spatio-temporal neighbourhood and more highly correlated model terms will be involved in the identification process. Therefore, it is a challenge to develop an effective identification method for the spatio-temporal systems in three dimensions or higher dimensions.

b) Why the OLS algorithm fails when it applied to spatio-temporal systems in some cases also needs further studies. In Chapter 4, the new OFR-MI algorithm is a complementary method for the classical OFR algorithm, rather than a substitute. This is because the new identification algorithm cannot always have correct identified results as the classical algorithm does. Hence, the method can be modified by adding more criterions for selecting model terms or be assisted by some effective neighbourhood detection algorithms.

c) The identification of the Nazim Fatès model is still an open problem. Although a new probabilistic multi-rule CA model which can be used to describe random dynamics in some spatio-temporal systems was introduced in this thesis, this is far from satisfactory for solving the identification problem of the Nazim Fatès model. It is because random processes which are included in the Nazim Fatès model increase the uncertainty in the data for the identification and make the underlying transition rule more complicated to be identified. To solve this problem, a more powerful identification algorithm needs to be designed.

d) As mentioned in Chapter 7, the estimated parameters in the identified probabilistic multi-rule CA model may be biased because of spatio-temporal noise, which may not be uncorrelated and homoscedastic. Hence, a more complex algorithm for estimating the parameters should be developed to fit for the identification from spatio-temporal noisy patterns.

References

[1] Gilbert, S.F., *Developmental Biology, 6th edition*. 2000: Sinauer Associates.

[2] Nagano, S., *Modeling the model organism Dictyostelium discoideum.* Development, Growth & Differentiation, 2000. **42**(6): p. 541-550.

[3] *Life Cycle of Dictyostelium Discoideum*. [cited; Available from: http://comp.uark.edu/~mlehmann/simple_systems.pdf.

[4] Höfer, T. and P.K. Maini, *Streaming Instability of Slime Mold Amoebae: An Analytical Model.* Physical Review E, 1997. **56**(2): p. 2074-2080.

[5] Dormann, D., B. Vasiev, and C.J. Weijer, *Becoming Multicellular by Aggregation; The Morphogenesis of the Social Amoebae Dicyostelium discoideum.* Journal of Biological Physics, 2002. **28**(4): p. 765-780.

[6] Rotstein, H.G., et al., *Canard phenomenon and localization of oscillations in the Belousov-Zhabotinsky reaction with global feedback.* Journal of Chemical Physics, 2003. **119**(17): p. 8824-8832.

[7] Parent, C.A. and P.N. Devreotes, *Molecular Genetics of Signal Transduction in Dictyostelium.* Annual Review of Biochemistry, 1996. **65**(1): p. 411-440.

[8] Martiel, J.-L. and A. Goldbeter, *A Model Based on Receptor Desensitization for Cyclic AMP Signaling in Dictyostelium Cells.* Biophysical Journal, 1987. **52**(5): p. 807-828.

[9] Devreotes, P., *Cell-cell interactions in Dictyostelium development.* Trends in Genetics, 1989. **5**: p. 242-245.

[10] Tyson, J.J. and J.D. Murray, *CYCLIC-AMP WAVES DURING AGGREGATION OF DICTYOSTELIUM AMEBAS.* Development, 1989. **106**(3): p. 421-426.

[11] Dormann, D. and C.J. Weijer, *Chemotactic cell movement during development.* Current Opinion in Genetics & Development, 2003. **13**(4): p. 358-364.

[12] Varnum-Finney, B., N.A. Schroeder, and D.R. Soll, *Adaptation in the motility response to cAMP in Dictyostelium discoideum.* Cell motility and the cytoskeleton, 1988. **9**(1): p. 9-16.

[13] Tomchik, K.J. and P.N. Devreotes, *Adenosine 3',5'-monophosphate waves in Dictyostelium discoideum: a demonstration by isotope dilution--fluorography.* Science, 1981. **212**(4493): p. 443-446.

[14] Savill, N.J. and P. Hogeweg, *Modelling Morphogenesis: From Single Cells to Crawling Slugs.* Journal of Theoretical Biology, 1997. **184**(3): p. 229-235.

[15] Hanson, K.L., et al., *Fungi Use Efficient Algorithms for the Exploration of Microfluidic Networks.* Small, 2006. **2**(10): p. 1212-1220.

[16] Nakagaki, T., H. Yamada, and A. Toth, *Intelligence: Maze-solving by an amoeboid organism.* Nature, 2000. **407**(6803): p. 470-470.

[17] Nakagaki, T., H. Yamada, and Á. Tóth, *Path finding by tube morphogenesis in an amoeboid organism.* Biophysical Chemistry, 2001. **92**(1-2): p. 47-52.

[18] Nakagaki, T., et al., *Obtaining multiple separate food sources: behavioural intelligence in the Physarum plasmodium.* Proceedings of the Royal Society of London. Series B: Biological Sciences, 2004. **271**(1554): p. 2305-2310.

[19] Nakagaki, T., H. Yamada, and M. Hara, *Smart network solutions in an amoeboid organism.* Biophysical Chemistry, 2004. **107**(1): p. 1-5.

[20] Adamatzky, A., *Physarum machines: encapsulating reaction–diffusion to compute spanning tree.* Naturwissenschaften, 2007. **94**(12): p. 975-980.

[21] Bebber, D.P., et al., *Biological solutions to transport network design.* Proceedings of the Royal Society B: Biological Sciences, 2007. **274**(1623): p. 2307-2315.

[22] Tsuda, S., K.-P. Zauner, and Y.-P. Gunji, *Robot Control with Biological Cells,* in *Proceedings of the Sixth International Workshop on Information Processing in Cells and Tissues*. 2005: St. William's College, York. p. 202-216.

[23] Aono, M., et al., *A Model of Amoeba-Based Neurocomputer.* Journal of Computer Chemistry, Japan, 2010. **9**(3): p. 143-156.

[24] Keller, E.F. and L.A. Segel, *Initiation of slime mold aggregation viewed as an instability*. Journal of Theoretical Biology, 1970. **26**(3): p. 399-415.

[25] Turing, A.M., *The Chemical Basis of Morphogenesis.* Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences, 1952. **237**(641): p. 37-72.

[26] Parnas, H. and L.A. Segel, *Computer evidence concerning the chemotactic signal in Dictyostelium discoideum.* J Cell Sci, 1977. **25**(1): p. 191-204.

[27] MacKay, S.A., *Computer simulation of aggregation in Dictyostelium discoideum.* J Cell Sci, 1978. **33**(1): p. 1-16.

[28] Kessler, D.A. and H. Levine, *Pattern formation in Dictyostelium via the dynamics of cooperative biological entities.* Physical Review E, 1993. **48**(6): p. 4801.

[29] Vasiev, B.N., P. Hogeweg, and A.V. Panfilov, *Simulation of Dictyostelium Discoideum Aggregation via Reaction-Diffusion Model.* Physical Review Letters, 1994. **73**(23): p. 3173.

[30] Höfer, T., J.A. Sherratt, and P.K. Maini, *Dictyostelium discoideum: Cellular Self-Organization in an Excitable Biological Medium.* Proceedings of the Royal Society of London. Series B: Biological Sciences, 1995. **259**(1356): p. 249-257.

[31] van Oss, C., et al., *Spatial Pattern Formation During Aggregation of the Slime Mould Dictyostelium discoideum.* Journal of Theoretical Biology, 1996. **181**(3): p. 203-213.

[32] Meinhardt, H., *A model for the prestalk/prespore patterning in the slug of the slime mold Dictyostelium discoideum.* Differentiation, 1983. **24**(1-3): p. 191-202.

[33] Pate, E.F. and H.G. Othmer, *Differentiation, cell sorting and proportion regulation in the slug stage of Dictyostelium discoideum.* Journal of Theoretical Biology, 1986. **118**(3): p. 301-319.

[34] Sekimura, T. and Y. Kobuchi, *A spatial pattern formation model for Dictyostelium discoideum.* Journal of Theoretical Biology, 1986. **122**(3): p. 325-338.

[35] Marée, A.F.M., A.V. Panfilov, and P. Hogeweg, *Migration and Thermotaxis of Dictyostelium discoideum Slugs, a Model Study.* Journal of Theoretical Biology, 1999. **199**(3): p. 297-309.

[36] Marée, A.F.M., A.V. Panfilov, and P. Hogeweg, *Phototaxis during the slug stage of Dictyostelium discoideum: a model study.* Proceedings of the Royal Society of London. Series B: Biological Sciences, 1999. **266**(1426): p. 1351-1360.

[37] Vasiev, B. and C.J. Weijer, *Modelling of Dictyostelium discoideum slug migration.* Journal of Theoretical Biology, 2003. **223**(3): p. 347-359.

[38] Glazier, J.A., et al., *Simulation of the differential adhesion driven rearrangement of biological cells.* Physical Review E, 1993. **47**(3): p. 2128.

[39] Goude, K. and S. O'Keefe, *A Cellular Automata model for Dictyostelium Discoideum.* Science, 2005: p. 1-69.

[40] Graner, F., et al., *Simulation of biological cell sorting using a two-dimensional extended Potts model.* Physical Review Letters, 1992. **69**(13): p. 2013.

[41] Palsson, E. and H.G. Othmer, *A model for individual and collective cell movement in Dictyostelium discoideum.* Proceedings of the National Academy of Sciences of the United States of America, 2000. **97**(19): p. 10448-53.

[42] Dallon, J.C. and H.G. Othmer, *A discrete cell model with adaptive signalling for aggregation of Dictyostelium discoideum.* Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences, 1997. **352**(1351): p. 391-417.

[43] Nanjundiah, V., *Chemotaxis, signal relaying and aggregation morphology.* Journal of Theoretical Biology, 1973. **42**(1): p. 63-105.

[44] Fatès, N., *Solving the decentralised gathering problem with a reaction–diffusion–chemotaxis scheme.* Swarm Intelligence, 2010. **4**(2): p. 91-115.

[45] Billings, S.A., S. Chen, and M.J. Korenberg, *Identification of MIMO non-linear systems using a forward-regression orthogonal estimator.* International Journal of Control, 1989. **49**(6): p. 2157-2189.

[46] Chen, S., S.A. Billings, and W. Luo, *Orthogonal least squares methods and their application to non-linear system identification.* International Journal of Control, 1989. **50**(5): p. 1873-1896.

[47] Greenberg, J.M., B.D. Hassard, and S.P. Hastings, *Pattern formation and periodic structures in systems modeled by reaction-diffusion equations.* Bulletin of The American Mathematical Society, 1978. **84**: p. 1296-1328.

[48] Solé, R., V., J. Valls, and J. Bascompte, *Spiral waves, chaos and multiple attractors in lattice models of interacting populations.* Physics Letters A, 1992. **166**(2): p. 123-128.

[49] Kawasaki, K., et al., *Modeling Spatio-Temporal Patterns Generated byBacillus subtilis.* Journal of Theoretical Biology, 1997. **188**(2): p. 177-185.

[50] von Neumann, J., *The general and logical theory of automata. In: Cerebral mechanisms in behavior:The Hixon symposium*, ed. Jeffries. 1951: New York: John Wiley.

[51] Pagnutti, C., M. Anand, and M. Azzouz, *Lattice geometry, gap formation and scale invariance in forests.* Journal of Theoretical Biology, 2005. **236**(1): p. 79-87.

[52] Billings, S.A. and Y. Yang, *Identification of the neighborhood and CA rules from spatio-temporal CA patterns.* IEEE Transactions on Systems, Man, and Cybernetics, 2003. **33**(2): p. 332-339.

[53] Alarcon, T., H.M. Byrne, and P.K. Maini, *A cellular automaton model for tumour growth in inhomogeneous environment.* Journal of Theoretical Biology, 2003. **225**: p. 257-274.

[54] Zhu, M.F. and C.P. Hong, *A Modified Cellular Automaton Model for the Simulation of Dendritic Growth in Solidification of Alloys.* ISIJ Int, 2001. **41**(5): p. 436-445.

[55] Yeh, A.G.O. and X. Li, *A cellular automata model to simulate development density for urban planning.* Environment and Planning B: Planning and Design, 2002. **29**(3): p. 431-450.

[56] Rosin, P.L., *Training cellular automata for image processing.* Image Processing, IEEE Transactions on, 2006. **15**(7): p. 2076-2087.

[57] Kaneko, K., *Spatiotemporal Intermittency in Coupled Map Lattices.* Progress of Theoretical Physics, 1985. **74**(5): p. 1033-1044.

[58] Jahnke, W., W.E. Skaggs, and A.T. Winfree, *Chemical vortex dynamics in the Belousov-Zhabotinsky reaction and in the two-variable oregonator model.* The Journal of Physical Chemistry, 1989. **93**(2): p. 740-749.

[59] Tabuchi, E., et al., *Spatio-temporal dynamics of brain activated regions during drinking behavior in rats.* Brain Research, 2002. **951**(2): p. 270-279.

[60] White, S.M. and K.A.J. White, *Relating coupled map lattices to integro-difference equations: dispersal-driven instabilities in coupled map lattices.* Journal of Theoretical Biology, 2005. **235**(4): p. 463-475.

[61] Sakaguchi, et al., *A coupled map lattice model for oscillatory growth in electrodeposition*. Vol. 75. 2006, Tokyo, JAPON: Physical Society of Japan.

[62] Reiter, C.A., *A local cellular model for snow crystal growth.* Chaos, Solitons & Fractals, 2005. **23**(4): p. 1111-1119.

[63] Holden, A.V., et al., *Coupled map lattices as computational systems.* Chaos: An Interdisciplinary Journal of Nonlinear Science, 1992. **2**(3): p. 367-376.

[64] FitzHugh, R., *Mathematical models of threshold phenomena in the nerve membrane.* Bulletin of Mathematical Biology, 1955. **17**(4): p. 257-278.

[65] Koros, E. and R.M. Noyes, *Oscillations in chemical systems. II. Thorough analysis of temporal oscillation in the bromate-cerium-malonic acid*

system. Journal of The American Chemical Society, 1972. **94**(25): p. 8649-8664.

[66] Leontaritis, I.J. and S.A. Billings, *Input-output parametric models for non-linear systems Part II: stochastic non-linear systems.* International Journal of Control, 1985. **41**(2): p. 329-344.

[67] Fred, C.R., P.M. Thomas, and H.P. Norman, *Extracting cellular automaton rules directly from experimental data*, in *Cellular automata*, G. Howard, Editor. 1990, MIT Press. p. 189-202.

[68] Yang, Y. and S.A. Billings, *Extracting Boolean rules from CA patterns.* Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 2000. **30**(4): p. 573-580.

[69] Billings, S.A. and Y. Yang, *Identification of probabilistic cellular automata.* Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 2003. **33**(2): p. 225-236.

[70] Billings, S.A. and S.S. Mei, *A new fast cellular automata orthogonal least-squares identification method.* International Journal of Systems Science, 2005. **36**(8): p. 491-499.

[71] Zhao, Y. and S.A. Billings, *Neighborhood detection using mutual information for the identification of cellular automata.* Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 2006. **36**(2): p. 473-479.

[72] Guo, Y., S.A. Billings, and D. Coca, *Identification of n-State Spatio-Temporal Dynamical Systems Using A Polynomial Model.* International Journal of Bifurcation and Chaos, 2008. **18**(7): p. 2049-2057.

[73] Packard, N.H., et al., *Geometry from a Time Series.* Physical Review Letters, 1980. **45**(9): p. 712.

[74] Parlitz, U. and C. Merkwirth, *Prediction of Spatiotemporal Time Series Based on Reconstructed Local States.* Physical Review Letters, 2000. **84**(9): p. 1890.

[75] Mandelj, S., I. Grabec, and E. Govekar, *Statistical approach to modelling of spatiotemporal dynamics.* International Journal of Bifurcation and Chaos, 2001. **11**(11): p. 2731-2738.

[76] Coca, D. and S.A. Billings, *Identification of coupled map lattice models of complex spatio-temporal patterns.* Physics Letters A, 2001. **287**(1-2): p. 65-73.

[77] Coca, D. and S.A. Billings, *Analysis and reconstruction of stochastic coupled map lattice models.* Physics Letters A, 2003. **315**(1-2): p. 61-75.

[78] Guo, L.Z. and S.A. Billings, *Identification of coupled map lattice models of stochastic spatio-temporal dynamics using wavelets.* Dynamical Systems, 2004. **19**(3): p. 265-278.

[79] Pan, Y. and S.A. Billings, *Neighborhood Detection for the Identification of Spatiotemporal Systems.* Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 2008. **38**(3): p. 846-854.

[80] Guo, L.Z., S.S. Mei, and S.A. Billings, *Neighbourhood detection and identification of spatio-temporal dynamical systems using a coarse-to-fine approach.* Intern. J. Syst. Sci., 2007. **38**(1): p. 1-15.

[81] Voss, H., et al., *Identification of continuous, spatiotemporal systems.* Physical Review E, 1998. **57**(3): p. 2820.

[82] Porcu, E., et al., *Modelling spatio-temporal data: A new variogram and covariance structure proposal.* Statistics & Probability Letters, 2007. **77**(1): p. 83-89.

[83] Xu, K. and C.K. Wikle, *Estimation of Parameterized Spatio-Temporal Dynamic Models*. 2005.

[84] González-García, R., R. Rico-Martínez, and I.G. Kevrekidis, *Identification of distributed parameter systems: A neural net based approach.* Computers & Chemical Engineering, 1998. **22**(Supplement 1): p. S965-S968.

[85] Coca, D. and S.A. Billings, *Direct parameter identification of distributed parameter systems.* International Journal of Systems Science, 2000. **31**(1): p. 11-17.

[86] Coca, D. and S.A. Billings, *Identification of finite dimensional models of infinite dimensional dynamical systems.* Automatica, 2002. **38**(11): p. 1851-1865.

[87] Qi, C., H.-T. Zhang, and H.-X. Li, *A multi-channel spatio-temporal Hammerstein modeling approach for nonlinear distributed parameter processes.* Journal of Process Control, 2009. **19**(1): p. 85-99.

[88] Guo, L. and S.A. Billings, *Identification of Partial Differential Equation Models for Continuous Spatio-Temporal Dynamical Systems.* Circuits and Systems II: Express Briefs, IEEE Transactions on, 2006. **53**(8): p. 657-661.

[89] Guo, L.Z., S.A. Billings, and D. Coca, *Consistent recursive parameter estimation of partial differential equation models.* International Journal of Control, 2009. **82**(10): p. 1946-1954.

[90] Woltering, M. and M. Markus, *Oscillations and turbulence induced by an activating agent in an active medium.* Physical Review E, 2001. **64**(4): p. 045601.

[91] Davidenko, J.M., et al., *Stationary and drifting spiral waves of excitation in isolated cardiac muscle.* Nature, 1992. **355**(6358): p. 349-351.

[92] Berridge, M.J., P. Lipp, and M.D. Bootman, *The versatility and universality of calcium signalling.* Nat Rev Mol Cell Biol, 2000. **1**(1): p. 11-21.

[93] Wiener, N. and A. Rosenblueth, *The mathematical formulation of the problem of conduction of impulses in a network of connected excitable elements, specifically in cardiac muscle.* Arch Inst Cardiol Mex, 1946. **16**(3): p. 205-265.

[94] Durston, A.J., *Dictyostelium discoideum aggregation fields as excitable media.* Journal of Theoretical Biology, 1973. **42**(3): p. 483-504.

[95] Fisch, R., J. Gravner, and D. Griffeath, *Metastability in the Greenberg-Hastings Model.* The Annals of Applied Probability, 1993. **3**(4): p. 935-967.

[96] Fisch, R., J. Gravner, and D. Griffeath, *Threshold-range scaling of excitable cellular automata.* Statistics and Computing, 1991. **1**(1): p. 23-39.

[97] Pethel, S.D., N.J. Corron, and E. Bollt, *Deconstructing Spatiotemporal Chaos Using Local Symbolic Dynamics.* Physical Review Letters, 2007. **99**(21): p. 214101.

[98] Chee, M.-N., R. Kapral, and S.G. Whittington, *Phase resetting dynamics for a discrete reaction-diffusion model*. Vol. 92. 1990: AIP. 7315-7322.

[99] Jiang, Y., *Phase transitions in two-variable coupled map lattices.* Physical Review E, 1997. **56**(3): p. 2672.

[100] Nakagaki, T., H. Yamada, and T. Ueda, *Interaction between cell shape and contraction pattern in the Physarum plasmodium.* Biophysical Chemistry, 2000. **84**(3): p. 195-204.

[101] Shannon, C.E., *A Mathematical Theory of Communication.* Bell System Technical Journal, 1984. **27**: p. 379-423.

[102] Darbellay, G.A. and I. Vajda, *Estimation of the information by an adaptive partitioning of the observation space.* Information Theory, IEEE Transactions on, 1999. **45**(4): p. 1315-1321.

[103] Moddemeijer, R., *A statistic to estimate the variance of the histogram-based mutual information estimator based on dependent pairs of observations.* Signal Processing, 1999. **75**(1): p. 51-63.

[104] Moon, Y.-I., B. Rajagopalan, and U. Lall, *Estimation of mutual information using kernel density estimators.* Physical Review E, 1995. **52**(3): p. 2318-2321.

[105] Endres, D. and P. Foldiak, *Bayesian bin distribution inference and mutual information.* Information Theory, IEEE Transactions on, 2005. **51**(11): p. 3766-3779.

[106] Wen, P., et al., *Estimation of Mutual Information: A Survey*, in *Rough Sets and Knowledge Technology*. 2009, Springer Berlin / Heidelberg. p. 389-396.

[107] Wei, H.-L. and S.A. Billings, *Model structure selection using an integrated forward orthogonal search algorithm assisted by squared correlation and mutual information.* International Journal of Modelling, Identification and Control, 2008. **3**(4): p. 341-356.

[108] Billings, S.A., M.J. Korenberg, and S. Chen, *Identification of non-linear output-affine systems using an orthogonal least-squares algorithm.* International Journal of Systems Science, 1988. **19**: p. 1559-1568.

[109] Korenberg, M., et al., *Orthogonal parameter estimation algorithm for non-linear stochastic systems.* International Journal of Control, 1988. **48**(1): p. 193-210.

[110] Billings, S.A. and Q.M. Zhu, *A structure detection algorithm for nonlinear dynamic rational models.* International Journal of Control, 1994. **59**(6): p. 1439-1463.

[111] Chen, S., C.F.N. Cowan, and P.M. Grant, *Orthogonal least squares learning algorithm for radial basis function networks.* Neural Networks, IEEE Transactions on, 1991. **2**(2): p. 302-309.

[112] Eng-Siong, C., H.H. Yang, and S. Bos, *Orthogonal least-squares learning algorithm with local adaptation process for the radial basis function networks.* Signal Processing Letters, IEEE, 1996. **3**(8): p. 253-255.

[113] Yifan, P., A.B. Stephen, and Z. Yifan, *The Identification of Coupled Map Lattice Models for Autonomous Cellular Neural Network Patterns.* I. J. Bifurcation and Chaos, 2008. **18**(4): p. 985-996.

[114] Zhao, Y., S.A. Billings, and A.F. Routh, *Identification of the Belousov-zhabotinskii Reaction Using Cellular Automata Models.* I. J. Bifurcation and Chaos, 2007: p. 1687-1701.

[115] Zhao, Y. and S.A. Billings, *The Identification of Cellular Automata.* Journal of Cellular Automata, 2007. **2**: p. 47-65.

[116] Li, W., *Mutual information functions versus correlation functions.* Journal of Statistical Physics, 1990. **60**(5-6): p. 823-837.

[117] Mao, K.Z. and S.A. Billings, *Algorithms for minimal model structure detection in nonlinear dynamic system identification.* International Journal of Control, 1997. **68**(2): p. 311-330.

[118] Piroddi, L. and W. Spinelli, *An identification algorithm for polynomial NARX models based on simulation error minimization.* International Journal of Control, 2003. **76**(17): p. 1767-1781.

[119] Haykin, S., *Neural Networks: A Comprehensive Foundation (2nd Edition).* 1998: Prentice Hall.

[120] Nelles, O., *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*. 2001: Springer.

[121] Harris, C., X. Hong, and Q. Gan, *Adaptive modelling, estimation, and fusion from data: a neurofuzzy approach*. 2002: Springer.

[122] Billings, S.A. and L.A. Aguirre, *Effects of the sampling time on the dynamics and identification of nonlinear models.* International Journal of Bifurcation and Chaos, 1995. **5**: p. 1541-1556.

[123] Stoica, P., et al., *Model-structure selection by cross-validation.* International Journal of Control, 1986. **43**(6): p. 1841-1878.

[124] Billings, S.A. and Q.M. Zhu, *Model validation tests for multivariable nonlinear models including neural networks.* International Journal of Control, 1995. **62**(4): p. 749-766.

[125] Aguirre, L.A. and S.A. Billings, *Validating identified nonlinear models with chaotic dynamics.* International Journal of Bifurcation and Chaos, 1994. **4**(1): p. 109-125.

[126] Aguirre, L.A. and S.A. Billings, *Dynamical effects of overparametrization in nonlinear models.* Physica D: Nonlinear Phenomena, 1995. **80**(1-2): p. 26-40.

[127] Miller, A.J., *Subset selection in regression*. 1990: Chapman and Hall.

[128] Mark, J.L.O., *Regularization in the selection of radial basis function centers.* Neural Comput., 1995. **7**(3): p. 606-623.

[129] Billings, S.A. and W. Hua-Liang, *Sparse Model Identification Using a Forward Orthogonal Regression Algorithm Aided by Mutual Information.* Neural Networks, IEEE Transactions on, 2007. **18**(1): p. 306-310.

[130] Moody, J.E., *The Effective Number of Parameters: An Analysis of Generalization and Regularization in Nonlinear Learning Systems.* Neural Information Processing Systems, 1991: p. 847-854.

[131] Keller, E.F. and L.A. Segel, *Model for chemotaxis.* Journal of Theoretical Biology, 1971. **30**(2): p. 225-234.

[132] Ben-Jacob, E., et al., *Modeling branching and chiral colonial patterning of lubricating bacteria.* in Mathematical Models for Biological Pattern Formation, K. M. Philip and G. O. Hans, Eds. The IMA Volumes in Mathematics and Its Applications, 2001. **121**: p. 211-254.

[133] Tyson, R., S.R. Lubkin, and J.D. Murray, *Model and analysis of chemotactic bacterial patterns in a liquid medium.* Journal of Mathematical Biology, 1999. **38**(4): p. 359-375.

[134] Budrene, E.O. and H.C. Berg, *Complex patterns formed by motile cells of Escherichia coli.* Nature, 1991. **349**(6310): p. 630-633.

[135] Budrien, E.O., A.A. Polezhaev, and M.O. Ptitsyn, *Mathematical modelling of intercellular regulation causing the formation of spatial structures in bacterial colonies.* Journal of Theoretical Biology, 1988. **135**(3): p. 323-341.

[136] Murray, J.D., *Mathematical biology: An introduction*. 2002: Springer.

[137] Keller, E.F. and L.A. Segel, *Traveling bands of chemotactic bacteria: A theoretical analysis.* Journal of Theoretical Biology, 1971. **30**(2): p. 235-248.

[138] Sugihara, K. and I. Suzuki. *Distributed motion coordination of multiple mobile robots*. in *Intelligent Control, 1990. Proceedings., 5th IEEE International Symposium on*. 1990.

[139] Christensen, A.L., R. O'Grady, and M. Dorigo, *SWARMORPH-script: a language for arbitrary morphology generation in self-assembling robots.* Swarm Intelligence, 2008. **2**(2-4): p. 143-165.

[140] Prencipe, G., *Impossibility of gathering by a set of autonomous mobile robots.* Theoretical Computer Science, 2007. **384**(2-3): p. 222-231.

[141] Simpson, M.J., et al., *Simulating invasion with cellular automata: Connecting cell-scale and population-scale properties.* Physical Review E, 2007. **76**(2): p. 021918.

[142] Ilachinski, A., *Cellular Automata: A Discrete Universe*. 2001, London: World Scientific.

[143] Burton, H.V., *Computational analysis of one-dimensional cellular automata*. 1996: World Scientific.

[144] Wolfram, S., *Cellular automata and complexity: collected papers*. 1994: Addison-Wesley Pub. Co.

[145] Zhao, Y., S.A.Billings, and D.Coca, *Cellular automata modelling of dendritic crystal growth based on Moore and von Neumann neighbourhoods.* International Journal of Modelling, Identification and Control 2009. **6**(2): p. 119-125.

[146] Piotrowska, M.J. and S.D. Angus, *A quantitative cellular automaton model of in vitro multicellular spheroid tumour growth.* Journal of Theoretical Biology, 2009. **258**(2): p. 165-178.

[147] Niloy, G., et al., *Design and characterization of cellular automata based associative memory for pattern recognition.* Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 2004. **34**(1): p. 672-678.

[148] Nandi, S., B.K. Kar, and P.P. Chaudhuri, *Theory and Applications of Cellular Automata in Cryptography.* IEEE Trans. Comput., 1994. **43**(12): p. 1346-1357.

[149] Mardiris, V., et al., *A CAD System for Modeling and Simulation of Computer Networks Using Cellular Automata.* Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 2008. **38**(2): p. 253-264.

[150] Cattell, K., et al., *2-by-n hybrid cellular automata with regular configuration: theory and application.* Computers, IEEE Transactions on, 1999. **48**(3): p. 285-295.

[151] Adamatzky, A., *Identification of Cellular Automata*. 1994: Taylor&Francis.

[152] Aguirre, L.A. and S.A. Billings, *Dynamical effects of overparametrization in nonlinear models.* Physica D: Nonlinear Phenomena, Jan.1995. **80**(1-2): p. 26-40.

[153] Billings, S.A., S. Chen, and M.J. Korenberg, *Identification of mimo nonlinear systems using a forward-regression orthogonal estimator.* International Journal of Control, 1989. **49**(6): p. 2157-2189.

[154] Billings, S.A. and Y. Yang, *Identification of the neighborhood and CA rules from spatio-temporal CA patterns.* Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 2003. **33**(2): p. 332-339.

[155] Rapaport, D.C., *The art of molecular dynamics simulation.* 2004: Cambridge University Press.

[156] Folke, C., et al., *Regime Shifts, Resilience, and Biodiversity In Ecosystem Management.* Annual Review of Ecology, Evolution, and Systematics, 2004. **35**(1): p. 557-581.

[157] Bandman, O., et al., *Simulating Spatial Dynamics by Probabilistic Cellular Automata Cellular Automata*. 2002, Springer Berlin / Heidelberg. p. 10-19.

[158] Nobe, A. and et al., *From cellular automaton to difference equation: a general transformation method which preserves time evolution patterns.* Journal of Physics A: Mathematical and General, 2001. **34**(25): p. L371-L379.