# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

**4,800**
Open access books available

**122,000**
International authors and editors

**135M**
Downloads

Our authors are among the

**154**
Countries delivered to

**TOP 1%**
most cited scientists

**12.2%**
Contributors from top 500 universities

**BOOK CITATION INDEX**
CLARIVATE ANALYTICS
INDEXED

**WEB OF SCIENCE™**

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

## Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

# Cellular Automata for Pattern Recognition

Sartra Wongthanavasu and Jetsada Ponkaew

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/52364

## 1. Introduction

Cellular Automata (CA) are spatiotemporal discrete systems (Neumann, 1966) that can model dynamic complex systems. A variety of problem domains have been reported to date in successful CA applications. In this regard, digital image processing is one of those as reported by Wongthanavasu et. al. (Wongthanavasu et al., 2003; 2004; 2007) and Rosin (Rosin, 2006).

Generalized Multiple Attractor CA (GMACA) is introduced for elementary pattern recognition (Ganguly et al., 2002; Maji et al., 2003; 2008). It is a promising pattern classifier using a simple local network of Elementary Cellular Automata (ECA) (Wolfram, 1994), called attractor basin that is a reverse tree-graph. GMACA utilizes a reverse engineering technique and genetic algorithm in ordering the CA rules. This leads to a major drawback of computational complexity, as well as recognition performance. There are reports in successful applications of GMACA in error correcting problem with only one bit noise. It shows the promising results for the restricted one bit noise, but becomes combinatorial explosion in complexity, using associative memory, when a number of bit noises increases.

Due to the drawbacks of complexity and recognition performance stated previously, the binary CA-based classifier, called Two-class Classifier Generalized Multiple Attractor Cellular Automata with artificial point (2C2-GMACA), is presented. In this regard, a pattern recognition of error correcting capability is implemented comprehensively in comparison with GMACA. Following this, the basis on CA for pattern recognition and GMACA's configuration are presented. Then, the 2C2-GMACA model and its performance evaluation in comparison with GMACA are provided. Finally, conclusions and discussions are given.

## 2. Cellular Automata for Pattern Recognition

Elementary Cellular Automata (ECA) (Wolfram, 1994) is generally utilized as a basis on pattern recognition. It is the simplest class of one dimension (1d) CA with $n$ cells, 2 states and 3 neighbors. A state is changed in discrete time and space ( $S_i^t \rightarrow S_i^{t+1}$ ; where $S_i^t$ is the present state and $S_i^{t+1}$ is the next state for the $i^{th}$ cell) by considering it nearest neighbor ( $S_{i-1}^t$, $S_i^t$, $S_{i+1}^t$ ) of the present state.
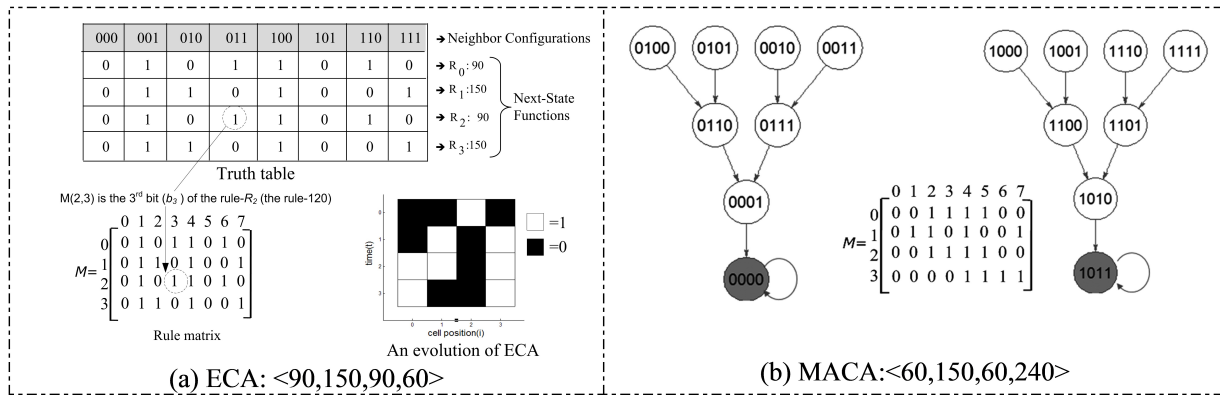


**Figure 1.** Elementary Cellular Automata (ECA) and Generalized Multiple Attractor Cellular Automata (GMACA).

For $n$-cell ECA, the next state function ( $S_i^t \rightarrow S_i^{t+1}$ ) can be represented by a rule matrix ($M$) with size $|n \times 8|$ and the nearest neighbour configuration ( $S_{i-1}^t$, $S_i^t$, $S_{i+1}^t$ ) of the present state. Suppose a $n$-cell ECA ( $S_0^t S_1^t S_2^t \dots S_{n-1}^t$ ) at time 't' is changed in discrete time by a rule vector $<R_0, R_1, \dots, R_{n-1}>$ . A truth table is a simplified form of the rule vector as illustrated in Fig. 1(a). It comprises the possible 3 neighbor values of $S_{i-1}^t S_i^t S_{i+1}^t$ from 000 to 111, and the next states for the rule $R_i$; where $i$=0, 1, 2…, n-1. Each rule is represented in binary numbers ($b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0$). If the binary numbers are decoded into decimal, it must equal to the number $R_i$ such as '01011010' for the rule-90. Simultaneously, A rule matrix ($M$) can also be represented the rule vector.

Let $M(i,j)$ be an element of the matrix at the $i^{th}$ ($i$=0,1,2,...,n-1) row and the $j^{th}$ ($j$=0,1,2,...,7) column. The $M(i,j)$ is contained $b_j$ of the rule-$R_i$. For example, $M(2,3)$ is $b_3$ of the rule $R_2$ (the rule-90) that is '1'. Consequently, the next state ( $S_i^{t+1}$ ) for the $i^{th}$ cell is represented by the $M(i,j)$ as the following:

$$S_i^{t+1} = M(i, j_i) \qquad (1)$$

where;

$S_i^{t+1}$ is the next state of the $i^{th}$ cell.

$j_i$ is the 3 neighbouring values ( $S_{i-1}^t S_i^t S_{i+1}^t$ ) of the present state at the $i^{th}$ cell decoded in decimal.

The next state $(S^{t+1})$ for $n$-cell ECA calculated is also defined by the rule matrix $M$ as following:

$$\begin{aligned} S^{t+1} &= (S_0^{t+1}, S_1^{t+1}, \ldots, S_{n-1}^{t+1}) \\ &= (M(0, j_0), M(1, j_1) \ldots, M(n-1, j_{n-1})) \\ &= (M, S^t) \end{aligned}$$

(2)

Suppose a system designed with a rule matrix ($M$) comprises a set of solutions $Y = \{ y_i \mid y_i \in \{0,1\}^n \}$ and an input $x$; $x \in \{0,1\}^n$ , where $i = 1, 2\ldots, N$. Consequently, the pattern classifiers based on the evolution of the ECA is defined as following

$$S^{t+1} = \begin{cases} (M, S^t), if\ S^t \notin Y \\ S^t\ and\ stop, otherwise \end{cases}$$

(3)

For an input $x$ , it must be identified a solution from $Y$ using the equation (3). Firstly, the present state ( $S^t$ ) will be set to $x$ . Then, the next state ( $S^{t+1}$ ) will be generated using the rule matrix $M$ until it reaches some solution ( $S^t \in Y$ ). The structure for pattern classification using ECA can be represented by a simple local network called attractor basin. It consists of a cyclic and non-cyclic states. The cyclic state contains a pivotal point which is a solution to classification problem, while the transient states (all possible inputs) are contained in the non-cyclic states. The attractor cycle lengths (height) in the GMACA (Oliveira, et al., 2006; Sipper, 1996) are greater than or equal to one, while Multiple Attractor Cellular Automata (MACA) (Das, et al., 2008; Maji, et al., 2003; Sipper, 1996) is limited to one. In Fig. 1(b), two attractor basins of 4-bit pattern of MACA with null boundary condition are designed with a rule vector <60, 150, 60, 240>. The target solution patterns are 0000 and 1011, respectively. The rule vector is ordered by the evolution of heuristic search using simulated annealing algorithm.

## 3. Generalized Multiple Attractor Cellular Automata

This section gives the detailed configuration of GMACA and its application in ECC. Suppose *an n*-bit pattern is sent in a communication system. Let $X$ be the sender's pattern and $Y$ be the receiver's pattern. Thus, the number of different bits between $X$ and $Y$ is determined by Hamming distance ($r$) defined as follows:

$$r = \sum_{i=0}^{n-1} |x_i - y_i| \tag{4}$$

where $X = x_0 x_1 \ldots x_{n-1}; x_i \in \{0,1\}$ and $Y = y_0 y_1 \ldots y_{n-1}; y_i \in \{0,1\}$.

The number of possible error patterns ($p_r$) for a given $r$ of $n$-bit communication can be expressed as follow:

$$p_r = \binom{n}{r} \tag{5}$$

Then, the number of all possible error patterns ($p_{All}$) for a given $r_{max}$, where $r_{max} \in (0, n)$ is the maximum permissible noise, is given by:

$$p_{All} = \sum_{r=0}^{r_{max}} \binom{n}{r} \tag{6}$$

The maximum permissible noise ($r_{max}$) is the highest value of $r$ allowed to occur in the communication system. The Hamming distance model of a message (pattern) and it errors are also represented by an attractor basin—that is, the messages is a pivotal point while the errors are transient states. Thus, the error correcting codes can be solved by the Generalized Multiple Attractor Cellular Automata (GMACA).
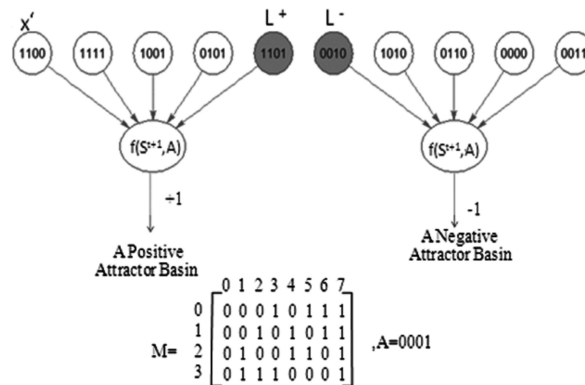


**Figure 2.** Two-Class Classifier GMACA with artificial point (2C2-GMACA): <232,212,178,142 >.

Suppose a communication system comprises $k$ original messages of $n$-bit data and the maximum permissible noise $r_{max}$. If error messages are corrected using the GMACA, thus a satisfied rule vector is required. The rule vector is a result of a reverse engineering technique. Firstly, $k$ attractor basins are randomly constructed with the number of nodes for each at-

tractor basin equals $p_{All}$. Then original messages are randomly mapped into pivotal points while its possible errors are also randomly mapped into transient states at the same attract basin. Finally, the search heuristics, such as simulated annealing (SA) and genetic algorithm (GA) (Holland, 1992; Shuai, et al., 2007; Jie, et al., 2002) have been taken to explore the optimal structure. The search heuristics then iteratively changes directions and height of the attractor basins until the satisfied rule vector is acquired.

As reported in Ganguly, et al., 2002, Maji, et al., 2003 and Maji, et al., 2008, the GMACA provides the best performance of pattern recognition if it is trained with the $r_{max}$ having a value of 1. Although percentage of recognition in testing is high when deals with the $r_{max}$ equals 1, it sharply decreases the recognition performance when the $r_{max}$ is greater than 1.

## 4. Proposed 2C2-GMACA Model

Due to the drawbacks of recognition performance resulting from the increasing $r_{max}$ and search space complexity in rule ordering, the proposed method, called Two-class Classifier Generalized Multiple Attractor Cellular Automata with artificial point (2C2-GMACA) (Ponkaew, et al., 2011; Ponkaew, et al., 2011), is introduced. The 2C2-GMACA is designed based on two class classifier architecture basis. In this regard, two classes are taken to process at a time and a solution is binary answer +1 or -1, which is a pointer to the class label of solution. There are two kinds of attractor basins: a positive attractor basin that returns the +1 as the result and a negative attractor basin, otherwise.

Suppose a system consists of patterns $(x_i, y_i)$, where $x_i \in \{0,1\}^n$ is the $i^{th}$ pattern , and $y_i \in \{L^+, L^-\}$ is the $i^{th}$ class label and $i=1,2,\dots N$. Let $L^+$ and $L^-$ be a class label of the positive and negative attractor basins, respectively. Given $x \in \{0,1\}^n$ as an input, the x must be assigned a class label which is a solution to the pattern recognition. The 2C2-GMACA begins with setting the present state ($S^t$) to $x$. Then, the $S^t$ will be evolved with the equation (2) to the next state ($S^{t+1}$). Next, the binary decision function will take $S^{t+1}$ and artificial point ($A$) as parameters as the equation (7) to assign the class.

$$f(S^{(t+1)}, A) = sgn(\sum_{i=0}^{n-1} S_i^{t+1}.A_i - \sum_{i=0}^{n-1} \bar{S}_i^{t+1}.A_i) \tag{7}$$

where

$sgn(\_)$ denotes the sign function.

$S_i^{t+1}$ represents the next state for the $i^{th}$ cell.

$A_i$ represents the artificial point for the $i^{th}$ cell.

$\bar{S}_i^{t+1}$ represents a bit complement of the next state for the $i^{th}$ cell.

(.) denotes "AND" logical operator.

Finally, the $x$ is considered to be a member of the positive attractor basin and returns $L^+$ if $f(S^{t+1}, A) = +1$, and returns $L^-$, otherwise.

*Example 1*: Consider two attractor basins of 4-bit recognizer of 2C2-GMACA with periodic boundary condition given in Fig. 2, they are designed by a rule vector <232,212,178,142> representing in a matrix $M$, and an artificial point ($A$) of '0001'. Suppose a class label of the positive ($L^+$) and the negative attractor basins ($L^-$) are '1101' and '0010', respectively. For an input $x' = '1100'$, firstly the present state $(S^t; t=0)$ is set to $x'$ and then evolved with the given rule vector to the next state $(S^{t+1}; t+1=1)$ by the equation (2), resulting

$$S^1 = (S_0^0, S_1^0, S_2^0, S_3^0) = (M(0, j_0), M(1, j_1), M(2, j_2), M(3, j_3)) \tag{8}$$

where $j_i$ is the 3 neighbour values ($S_{i-1}^t S_i^t S_{i+1}^t$) for the $i^{th}$ cell decoded in decimal. That is, $j_0 = (011)_2 = 3$, $j_1 = (110)_2 = 6$, $j_2 = (100)_2 = 4$ and $j_3 = (001)_2 = 1$. Thus, the above equation is replaced with the $j_i$ in decimal as following

$$= (M(0,3), M(1,6), M(2,4), M(3,1)) = 1111 \tag{9}$$

Finally, the binary decision function will process the $S^{t+1}$, which equals "1111" using the artificial point $A=0001$ as co-parameters resulting in the following

$$f(S^{t+1}, A) = sgn\left(\sum_{i=0}^{n-1} S_i^{t+1}.A_i - \sum_{i=0}^{n-1} \bar{S}_i^{t+1}.A_i\right) = sgn\left((1.0 + 1.0 + 1.0 + 1.1) - (\bar{1.0} + \bar{1.0} + \bar{1.0} + \bar{1.1})\right) = +1 \tag{10}$$

The function returns 1 meaning that the input $x'$ is a member of positive attractor basin and then the label '1101' is assigned as the solution.

## 4.1. 2C2-GMACA with Associative and Nonassociative Memories

Given a set of patterns $Y = \{y_1, y_2 \dots, y_k\}$ represents original messages; where $y_i \in \{0,1\}^n$ and $i=1,2\dots,k$. 2C2-GMACA takes two patterns $\{y_i, y_j\}: y_i \neq y_j$ and $y_i, y_j \in Y$ to process at a time. For associative memory learning, all possible transient states of the $y_i$ and $y_j$ are generated using the equation (6) with the maximum permissible noise ($r_{max}$), while all transient states are randomly generated $r \in [0, r_{max}]$ for non-associative memory. Then, all states of $y_i$ and $y_j$ are mapped into the leaf nodes of the positive and negative attractor basins, respectively. After two attractor basins are completely constructed, it will be synthesized by a majority voting technique to arrive at the rule vector. In other word, the rule vector is determined in only one time step which is different from GMACA in that it is iteratively determined through the evolution of heuristic search. In this regard, complexity is the main drawback excluding recognition performance.

According to a binary classifier, 2C2-GMACA conducts multiclass classification by DDAG (Decision Directed Acyclic Graph), One-versus-All, One-versus-One, etc., for example. However, this paper focuses on DDAG approach [28]. Suppose that a set of three patterns $\{y_1, y_2, y_3\}$, where $y_i \in \{0,1\}^n$ and $i$=1, 2, 3, is constructed using the DDAG scheme. Thus, total number of binary classifier is ( $3 \bullet 2/2$ ) = 3. That is, (1 vs 3), (1 vs 2) and (2 vs 3) and the number of levels is $\lceil log_2 3 \rceil$ = 2. A root node is (1 vs 3) contained in the $0^{th}$-level. Then, (1 vs 2) and (2 vs 3) are contained in the $1^{st}$-level. Finally, the solutions {3, 2, 1} are labeled in the leaf nodes of the $2^{nd}$-level. In order to assign a class label for an unknown input $x \in \{0,1\}^n$, it is first evaluated at the root node. The node is exited through the left edge if the binary decision function is -1. On the other hand, it is exited via the right edge if the binary decision function is +1. The $x$ is evaluated until it reaches final level. At this point, a leaf node connecting to the edge of the binary decision function is assigned as the solution.

## 4.2. Design of Rule Vector

A majority voting rule is utilized to synthesize a rule vector for two attractor basins. It is one time step process which is different from a reverse engineering technique (Maji, et al., 2003; Maji, et al., 2008) using in GMACA. Reverse engineering technique continues reconstructing attractor basins randomly until arriving at the rule vector with the lowest collision. In this regard, 2C2-GMACA's time complexity for ordering the rule is simply O(1). However, it must search for an optimal artificial point which applies evolutionary heuristic search. The 2C2-GMACA synthesis scheme comprises three phases as follows.
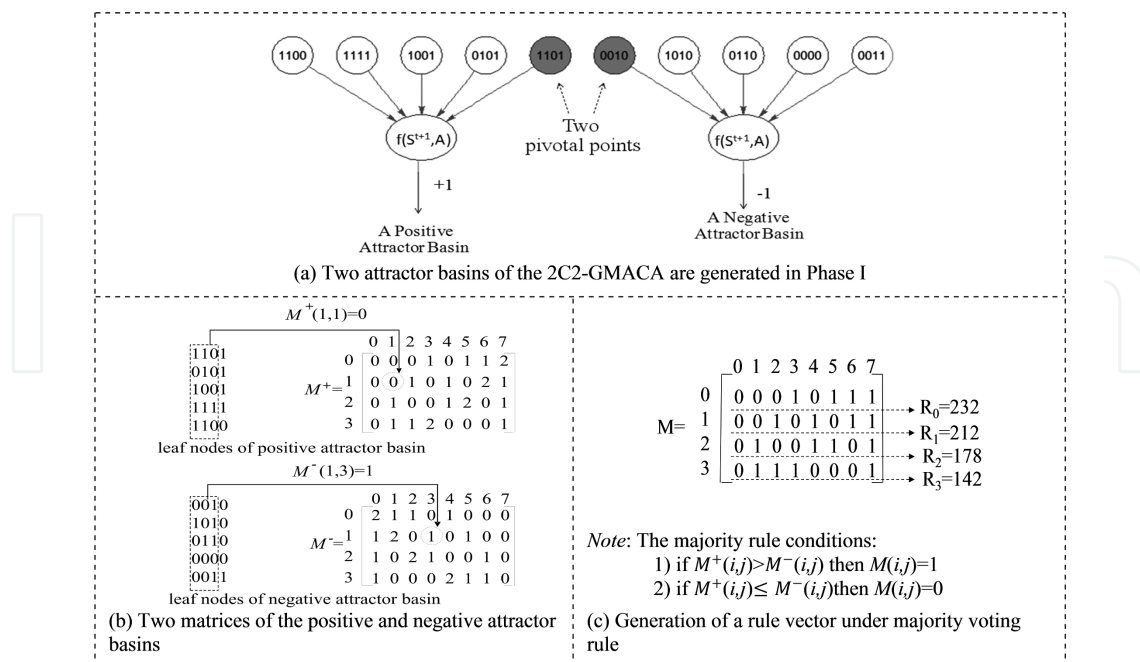


**Figure 3.** GMACA synthesize scheme under the majority voting rule.

*Phase I*--- Two attractor basins, namely, positive and negative attractor basins, are generated. In this phase, two patterns { $y_l$ , $y_m$ }, where $y_l \neq y_m$ and $y_l$, $y_m \in Y$ are chosen from a set of learnt patterns to process according to the multiclass classification scheme [28]. Suppose $y_l$ is assigned to a class label of $L^+$. Thus, the $y_m$ is a class label of $L^-$. Then, transient states of the $y_l$ and $y_m$ are generated into the leaf nodes of the positive and negative attractor basins, respectively.

*Example 1*: Fig. 3(a) represents two attractor basins based on associative memory learning of 4 bit patterns with $r_{max}$=1. Suppose $Y$={1101, 0010} is a set of learnt patterns. The 2C2-GMA-CA takes two patterns {$y_1$=1101, $y_2$=0010} to process according to the multiclass classification algorithm. Let a class label of the positive ($L^+$) and negative ($L^-$) attractor basins be '1101' and '0010', respectively. Then, two sets of noisy patterns with $r_{max}$=1 are generated resulting in {1101, 0101, 1001, 1111, 1100} and {0010, 1010, 0110, 0000, 0011}, respectively. Then, all patterns are mapped into leaf nodes of attractor basins corresponding with its label as shown in Fig. 3(a).

*Phase II*--- Let $M^+$ and $M^-$ be matrices with size |nx8|, and $M^+(i, j)$ and $M^-(i, j)$, where i=0,1,2,...,n-1 and j=0,1,2,...,7, be an element of the matrices $M^+$ and $M^-$, respectively. The $M^+(i, j)$ represents numbers of nodes from the positive attractor basin where the 3 neighbors, ( $S_{i-1}^t S_i^t S_{i+1}^t$ ), for the $i^{th}$ cell is decoded in decimal satisfying the $j^{th}$ column. The negative attractor basin considers the $M^-(i, j)$ under the similar condition with the positive one.

*Example 2*: As shown in Fig. 3(b), two matrices $M^+$ and $M^-$ are constructed with size |4x8|, each element of which is represented the numbers of nodes from corresponding attractor basin. For example, $M^+(1, 1)$ represents an element of matrix $M^+$ at the $1^{st}$ row and the $1^{st}$ column; it is a total number of leaf nodes from the positive attractor basin where 3 neighbors ( $S_0^t S_1^t S_2^t$ ) of the $1^{st}$ cell decoded in decimal equal to 1, i.e. j=1=$001_2$=( $S_{i-1}^t S_i^t S_{i+1}^t$ )$_2$ where i=1.

*Phase III*--- Rule matrix $M$ is determined. The matrix $M$ with size |nx8| is the simplified form of the rule vector (RV), while an element $M(i, j)$ represents the next state for the $i^{th}$ cell, where the 3 neighbor ($S_{i-1}^t S_i^t S_{i+1}^t$ ) of the cell decoded in decimal equal to $j$. The $M$ is designed by comparing between $M^+(i, j)$ and $M^-(i, j)$, where i=0,1,2,...,n-1 and j=0,1,2,...,7, due to the following conditions:

1) if $M^+(i, j) > M^-(i, j)$ then $M(i, j)$=1

2) if $M^+(i, j) \leq M^-(i, j)$ then $M(i, j)$=0

Fig. 3(c) shows that a rule vector <232, 212, 178, 142> is obtained by the majority voting technique. The rule vector (matrix rule) is utilized to evolve the given pattern in one time step to the pattern at the next time step which becomes one of parameters of the binary decision function.

### 4.3. Design of Artificial Point

An artificial point (A) takes a major role in the binary decision function. It interprets the next state ( $S^{t+1}$ ) in features space to be a pointer identifying the class label of solution. In this respect, Genetic Algorithm (GA) (Holland, 1992; Buhmann, et al., 1989) is implemented to determine the optimal artificial point. A chromosome with n genes in GA represents an n-bit artificial point as follows:

$$chromosome = \begin{bmatrix} b_0\ b_1\ b_2\ \dots\ b_{n-1} \end{bmatrix} \tag{11}$$

Selection is done by using a random pairing approach and a traditional single point cross-over is also performed by random at the same point of the $n$ element array of the selected two parents. Mutation makes a small change in the bits in the list of a chromosome with a small percentage. The fitness function is calculated as a cost for each chromosome. It is created from a true positive (TP) and a false positive (FP) of the confusion matrix (Simon, et al., 2010) calculated by the below equation (8). The fitness function is given as following

$$Fitness = 1 - \frac{TP}{TP + FP} \tag{12}$$

The search space complexity for rule ordering of the 2C2-GMACA is the all possible patterns of the artificial point, *000…000 to 111….111*, which is *2 ⁿ*, i.e. O(*2 ⁿ*).

## 5. Performance Evaluation

This section reports performance evaluation of the proposed method in comparison with GMACA on a set of measured matrices consisting of search space and classification complexities, recognition percentage, evolution time for rule ordering, and effects of the number of pivotal point, permissible noises, p-parameter, pattern size on error correcting problem.

### 5.1. Reduction of Search Space

Given a set of learnt patterns   $Y = \{y_1,\ y_2 \dots,\ y_k\}$ , where $y_i \in \{0,1\}^n$ and $i$=1,2…,$k$, is original messages. The 2C2-GMACA and GMACA based associative memory learning will generate all transient states using the equation (6) with the maximum permissible noise ($r_{max}$). Then, the transient states are constructed to be attractor basins.

*Theorem 1:* In training phase, a search space complexity of the GMACA ( $S_{GMACA}$ ) depends on parameters of bit patterns (*n*), the maximum permissible noise (*r $_{max}$*) and the maximum permissible height (*h $_{max}$*), while the search space complexity of 2C2-GMACA ( $S_{2C2\text{-}GMACA}$ ) depends only on a parameter *n*.

*Proof:* From the set $Y = \{y_1, y_2 \ldots, y_k\}$, GMACA constructs $k$ attractor basins randomly until a satisfied rule vector is acquired. Thus, the search space of the GMACA ($S_{GMACA}$) is all possible patterns of $k$ attractor basins defined by

$$S_{GMACA} = G^k \tag{13}$$

where $G$ is the number of learnt patterns in each attractor basin previously defined by Cayley 's formula (Maji, et al., 2003) as follows:

$$G = p^{p-2} \tag{14}$$

where $p$ is the number of possible transient states calculated from (6). Therefore, the above equation is defined following

$$S_{GMACA} = \left[ \sum_{r=0}^{r_{max}} \binom{n}{r} \right]^{\sum_{r=0}^{r_{max}} \binom{n}{r} - 2} \right]^k \in O(n!) \tag{15}$$

It shows that search space complexity of GMACA is factorial growth $O(n!)$, which depends on parameters $n$ and $r_{max}$. In real world application, it must face a severe search space in which the search heuristics cannot reach the optimal solution if $n$ or $r_{max}$ is considered at a high number. In this regard, GMACA tries to examine the optimal values of the $r_{max}$ and $h_{max}$. GMACA shows that the search space complexity can be reduced to $O(n^n)$ if the $r_{max}=1$ as shown following

$$S_{GMACA} = (n+1)^{kn-k} \in O(n^n) \tag{16}$$

The search space complexity in Maji, et al., 2003 and Maji, et al., 2008 is examined under the $h_{max}=2$ and the $r_{max}=1$ as described below.

$$S_{GMACA} = n^k \in O\left(n^c\right); c > 1 \tag{17}$$

For the proposed 2C2-GMACA, the search space is the number of possible patterns ($G$) of artificial point: *000…000 to 111….111*—that is; $2^n$. Due to DDAG approach for multiclass classification algorithm, the machine consists of $k(k-1)/2$ binary classifier. Thus, the search space complexity of the 2C2-GMACA ($S_{2C2\text{-}GMACA}$) is:

$$\begin{aligned} S_{2C2-GMACA} &= \frac{k(k-1)}{2} G \\ &\cong k^2(2^n) \in O(2^n) \end{aligned} \tag{18}$$

When comparing the search space complexity between GMACA and 2C2-GMACA, we found that GMACA can only be implemented if it is considered at the $h_{max}=2$ and $r_{max}=1$, while 2C2-GMACA can be implemented whatsoever with the exact solution through heuristic search. This corresponds to the reports in Maji, et al., 2003 and Maji, et al., 2008, the GMACA provides the best performance of pattern recognition when it is trained with the $r_{max}=1$ and $h_{max}=2$. However, the percentage of recognition in testing is also high if the Hamming distance of patterns is less than or equal to 1 and it is decreased sharply when the Hamming distance is greater than 1.

## 5.2. Reduction of Classification Complexity

*Theorem 2:* In worst case scenario of learning based on associative memory model, the classification complexity of *n*-bit pattern for GMACA is O($n^2$), while 2C2-GMACA is O($n$).

*Proof:* In general, time spent in classifying *n* nodes of GMACA depends on an arrangement of nodes in attractor basins. At worst, the attractor basin is a linear tree. Thus, time for classifying *n* nodes is the summation of the number of traversal paths from each node to a pivotal point. For example, the number of traversal paths of a pivotal point is 0 while the $n^{th}$-node is ($n$-1). This can be solved by arithmetic series ( $S_n$ ). Given the common different *d* is 1 and an initial term ($a_1$) is 0, the equation in determining the summation is given as follows.

$$S_n = \frac{n}{2}[2a_1 + (n-1)d]$$
$$= \frac{n}{2}[2(0) + (n-1)(1)] \tag{19}$$
$$= (\frac{n^2}{2} - \frac{n}{2}) \in O(n^2)$$

As being designed the height of attractor basis of 2C2-GMACA is limited to 1, the time of classifying *n* nodes is *n* , *ie*. $O(n)$.

## 5.3. Performance Analysis of 2C2-GMACA on Associative Memory

Pattern classifiers based on an associative memory is independent from the number of patterns to be learnt, because all possible distorted patterns are generated into learning system. Suppose a set of pivotal points $Y = \{y_1, y_2 \ldots, y_k\}$, where $y_i \in \{0,1\}^n$ and *i*=1, 2…, *k*, is original messages. 2C2-GMACA takes two pivotal points { $y_l$ , $y_m$ }, where $y_l$, $y_m \in Y$ , $y_l \neq y_m$ and *l*, *m*=1, 2…,*k*, to process at a time using the DDAG scheme. Thus, the number of classifiers of the 2C2-GMACA is $k \bullet (k-1)/2$, while GMACA takes all pivotal points to process at once.

### 5.3.1. Recognition and Evolution Time

This section reports recognition rate and evolution time for rule ordering between 2C2-GMACA and GMACA based on associative memory. Table 1 presents the recognition rate at different sizes of bit patterns (*n*) and the number of attractor basins (*k*). It generates pat-

terns with maximum permissible noise in training phase ($r_{max}$) and testing with different sizes of noise $r$; $r \in (1, r_{max})$. Table 2 presents the evolution time in second for the genetic algorithm in determining the well-fitting attractor basins and artificial point with different values of $n$ and $k$. The results show that 2C2-GMACA is superior to GMACA both recognition performance and times spent in rule ordering. This corresponds the previous mention that search space is the major problem of GMACA for ordering the rules when deals with high number of $r_{max}$.

### 5.3.2. Effects of Number of Pivotal Points and Pattern Size

A pivotal point in 2C2-GMACA represents an original message in communication systems. Fig. 4 shows the effects of the number of pivotal points ($k$) in the recognition performance of the proposed 2C2-GMACA based on associative memory learning at a particular $r_{max}$ and bit pattern. It shows that if is trained by $r_{max}$= 3 the recognition rate is almost 100% when the number of bit noises ($r$) is not greater than 5 no matter of the number of classes ($k$), and declined sharply when the number of bit noises increases. The less the number of classes, the better the recognition performance. Fig. 5 shows the effects of the number of bit pattern in recognition performance of the 2C2-GMACA based on associative memory learning by fixing $r_{max}$ and the number of classes ($k$). In this regard, when the number of bit noises in testing increases, the recognition of different number of bit patterns decreases in distinguishable manner. The more the number of bit patterns, the less the recognition performance.

## 5.4. Performance Analysis of 2C2-GMACA on Non-Associative Memory

The memory capacity becomes a serious problem of pattern classifier based on an associative memory learning if the classfier deal with the high values of $n$, $r_{max}$ and $k$. It generates a large number of transient states. In ordet to solve this problem, the 2C2-GMACA based on non-associcitve memory is presented. The transient states will be generated by randomly choosing bit noise $r \in (0, r_{max})$, the number of which is limited into some number $p; p \in I^+$.

### 5.4.1. Effects of Maximum Permissible Noise and P-Parameter

In order to examine the effects of the maximum permissible noise $r_{max}$ on the error correcting problem of 2C2-GMACA based non-associative memory, two pivotal points are randomly generated and then the number of transient states is limited to some number $p; p \in I^+$. Thus, the transient states are randomly generated from the equation (6) using $r \in (0, r_{max})$ until the number of states equals to $p$. This method is called uniform distribution learning. Fig. 6 shows the effects of the $r_{max}$ at $1/4 \bullet n$, $2/4 \bullet n$ and $3/4 \bullet n$; where $n$=100 and $n$ is bits pattern. The number of pivotal points ($k$) and transient states ($p$) is fixed to 2 and 2000, respectively. Results are plotted in the inverted bell curve. It shows that the 2C2-GMACA has the lowest capability in range of $r \in (0, 1/2 \bullet n)$ if it is trained by the $r_{max} \approx 3/4 \bullet n$, which opposed to the $r_{max}$= $1/2 \bullet n$. However, overall average percentage of the $r_{max}$ $\approx 3/4 \bullet n$ is the highest value.

The effects of the number of transient states ($p$ ; $p \in I^+$) for two attractor basins ($k$=2) are examined and shown in Fig. 7. During the training phase, the number of bit pattern ($n$) is set to 100, while the maximum permissible noise ($r_{max}$) is set nearly to $3/4 \bullet n \approx 75$. Then, the percentage of recognition is observed at different numbers of $p$---that is 2000, 4000 and 10000. The results show that the average percentage of recognition is highest if it is trained with the highest number of $p$. However, it is memory consumptions as already mentioned.
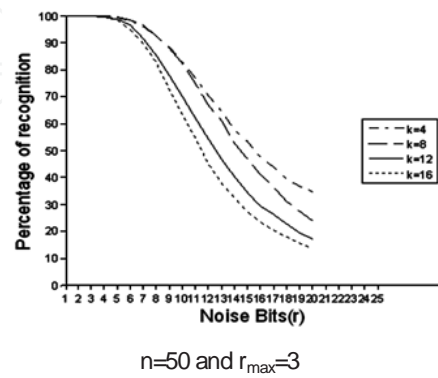


n=50 and $r_{max}$=3

**Figure 4.** The effect of $k$-parameter on the percentage of recognition of 2C2-GMACA based on associative memory.



k=15 and $r_{max}$=2
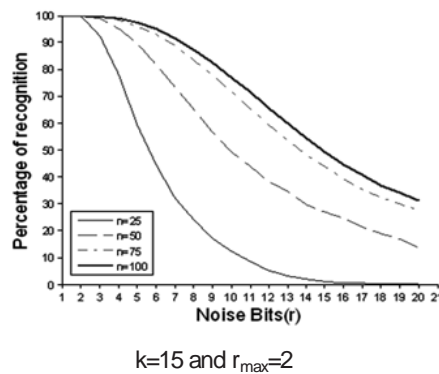
**Figure 5.** The effect of $n$-parameter on the percentage of recognition of 2C2-GMACA based on associative memory.



n=100, k=2 and p=2000

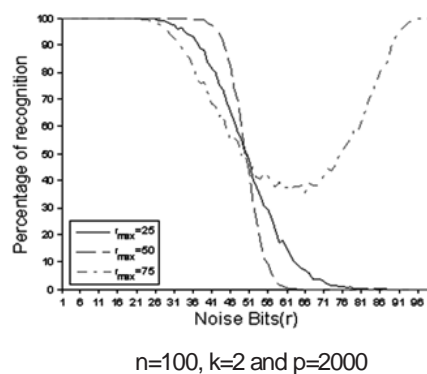**Figure 6.** The effect of $r_{max}$ parameter on the percentage of recognition of 2C2-GMACA based on non-associative memory.
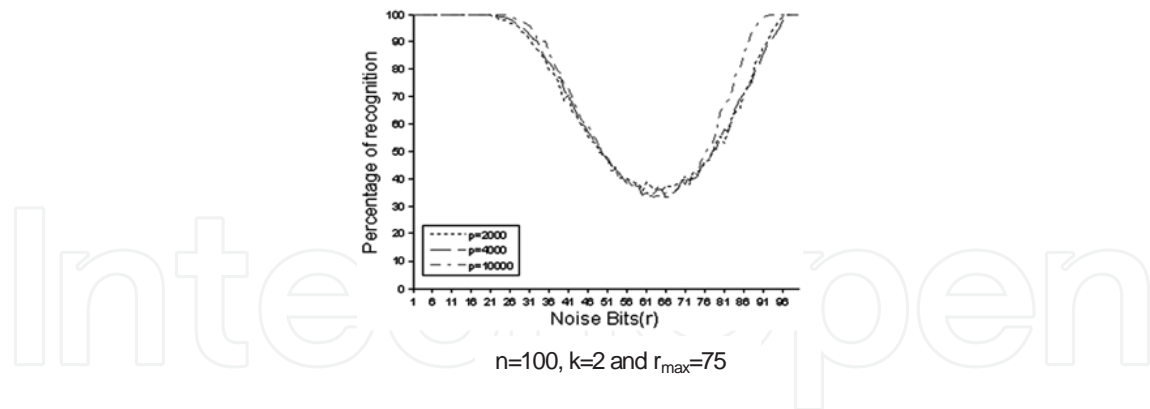
n=100, k=2 and r$_{max}$=75

**Figure 7.** The effect of *p*-parameter on the percentage of recognition of 2C2-GMACA based on non-associative memory.

## 6. Conclusions and Discussions

This chapter presents a non-uniform cellular automata-based algorithm with binary classifier, called Two-class Classifier Generalized Multiple Attractor Cellular Automata with artificial point (2C2-GMACA), for pattern recognition. The 2C2-GMACA is built around the simple structure of evolving non-uniform cellular automata called attractor basin, and classify the patterns on the basis of two-class classifier architecture similar to support vector machines. To reduce computational time complexity in ordering the rules, 2C2-GMACA is limited the height of attractor basin to 1, while GMACA can have its height to n, where n is a number of bit pattern. Genetic algorithm is utilized to determine the CA's best rules for classification. In this regard, GMACA designs one chromosome consists of k-genes, where k is a number of classes (target patterns) to be classified. This leads to abundant state spaces and combinatorial explosion in computation, especially when a number of bit noises increases. For the design of 2C2-GMACA, a chromosome represents an artificial point which is consists of n-bit pattern. Consequently, the state space is minimal and feasible in computation in general pattern recognition problem. The 2C2-GMACA reduces search space for ordering a rule vector from GMACA which is $O(n^n)$ to $O(1)+O(2^n)$. In addition, multiple errors correcting problem is empirically experimented in comparison between the proposed method and GMACA based on associative and non-associative memories for performance evaluation. The results show that the proposed method provides the 99.98% recognition rate superior to GMACA which reports 72.50% when used associative memory, and 95.00% and 64.30% when used non-associative memory, respectively. For computational times in ordering the rules through genetic algorithm, the proposed method provides 7 to 14 times faster than GMACA. These results suggests the extension of 2C2-GMACA to other pattern recognition tasks. In this respect, we are improving and extending the 2C2-GMACA to cope with complicated patterns in which state of the art methods, SVM, ANN, etc., for example, poorly report the classification performance, and hope to report our findings soon.

## Author details

Sartra Wongthanavasu[1*] and Jetsada Ponkaew[2]

*Address all correspondence to: wongsar@kku.ac.th

1 Machine Learning and Intelligent Systems (MLIS) Laboratory, Department of Computer Science, Faculty of Science, Khon Kaen University, Khon Kaen, Thailand

2 Cellular Automata and Knowledge Engineering (CAKE) Laboratory, Department of Computer Science, Faculty of Science, Khon Kaen University, Khon Kaen, Thailand

## References

[1] Neumann, J.V. (1966). Theory of Self-Reproducing Automata. University of Illinois Press.

[2] Wongthanavasu, S., & Sadananda, R. (2003). A CA-based edge operator and its performance evaluation. *Journal of Visual Communication and Image Representation*, 14, 2, 83-96.

[3] Wongthanavasu, S., & Lursinsap, C. (2004). A 3D CA-based edge operator for 3D images. *in Image Processing*, ICIP'04. 2004 International Conference on, 2004, 235-238.

[4] Wongthanavasu, S., & Tangvoraphonkchai, V. (2007). Cellular Automata-Based Algorithm and its Application in Medical Image Processing. *Image Processing*, ICIP 2007. IEEE International Conference on, 2007, 41-44.

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

[6] Wolfram, S. (1994). Cellular automata and complexity: collected papers. Addison-Wesley Pub. Co.

[7] Delgado, O.G., Encinas, L.H., White, S.H., Rey, A.M.d. and Sanchez, G.R.,. (2005). Characterization of the reversibility of Wolfram cellular automata with rule number 150 and periodic boundary conditions. *INFORMATION*, 8(4), 483-492.

[8] Das, S., & Chowdhury, D. R. (2008). An Efficient n x n Boolean Mapping Using Additive Cellular Automata. *Proceedings of the 8th international conference on Cellular Automata for Reseach and Industry*, Yokohama, Japan Springer-Verlag,, 168-173.

[9] Maji, P., Shaw, C., Ganguly, N., Sikdar, B.K., & Chaudhuri, P.P. (2003). Theory and application of cellular automata for pattern classification. *Fundam. Inf.*, 58(3-4), 321-354.

[10] Chady, M., Chady, a. M., & Poli, R. (1997). Evolution of Cellular-automaton-based Associative Memories. *Technical Report CSRP-97-15*, University of Birmingham, School of Computer Science, May.

[11] Ganguly, N., Maji, P., Sikdar, B.K., & Chaudhuri, P.P. (2002). Generalized Multiple Attractor Cellular Automata (GMACA) For Associative Memory. *International Journal of Pattern Recognition and Artificial Intelligence, Special Issue: Computational Intelligence for Pattern Recognition*, 16(7), 781-795.

[12] Maji, P., Ganguly, N., & Chaudhuri, P.P. (2003). Error correcting capability of cellular automata based associative memory. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 33(4), 466-480.

[13] Maji, P., & Chaudhuri, P.P. (2008). Non-uniform cellular automata based associative memory: Evolutionary design and basins of attraction. *Information Sciences*, 178(10), 2315-2336.

[14] Niloy, G., Maji, P., Sikdar, B.K., & Chaudhuri, P.P. (2004). Design and characterization of cellular automata based associative memory for pattern recognition. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 34(1), 672-678.

[15] Tan, S.K., & Guan, S.U. (2007). Evolving Cellular Automata to Generate Nonlinear Sequences with Desirable Properties. *Applied Soft Computing*, 7(3), 1131-1134.

[16] Shannon, C.E., & Weaver, W. (1949). The Mathematical Theory of Communication. *Univ. of Illinois Press Urbana*.

[17] de Oliveira, P.P.B., Bortot, J.C., & Oliveira, G.M.B. (2006). The best currently known class of dynamically equivalent cellular automata rules for density classification. *Neurocomputing*, 70(1-3), 35-43.

[18] Sipper, M. (1996). Co-evolving Non-Uniform Cellular Automata to Perform Computations.

[19] Holland, J.H. (1992). Adaptation in natural and artificial systems. MIT Press.

[20] Oh, H. (2001). Application of a Heuristic Search to the Deadlock Avoidance Algorithm that Achieves the High Concurrency. *INFORMATION*, 4(4).

[21] Shuai, D., Dong, Y., & Shuai, Q. (2007). A new data clustering. approach Generalized cellular automata. *Information Systems*, 32(7), 968-977.

[22] Jie, Y., & Binggang, C. (2002). Study on an Improved Genetic Algorithm. *INFORMATION*, 5(4).

[23] Ponkaew, J., Wongthanavasu, S., & Lursinsap, C. (2011). A nonlinear classifier using an evolution of Cellular Automata,. *Intelligent Signal Processing and Communications Systems (ISPACS)*, 2011 International Symposium on, 1-5.

[24] Ponkaew, J., Wongthanavasu, S., & Lursinsap, C. (2011). Two-class classifier cellular automata. *Industrial Electronics and Applications (ISIEA)*, 2011 IEEE Symposium on, 354-359.

[25] Buhmann, J., Divko, R., & Schulten, K. (1989). Associative memory with high information content. *Physical Review A*, 39(5), 2689-2692.

[26] Simon, D., & Simon, D.L. (2010). Analytic Confusion Matrix Bounds for Fault Detection and Isolation Using a Sum-of-Squared-Residuals Approach. *Reliability, IEEE Transactions on*, 59(2), 287-296.