Partitioning Study of Complex System

HE YUE, LIN CHUN-MEI
Department of Mathematics
Shaoxing College of Arts and Sciences
Shaoxing Zhejiang 312000
CHINA
Department of Computer
Shaoxing College of Arts and Sciences
Shaoxing Zhejiang 312000
CHINA

Abstract: - Whether the design of knowledge base or the modeling of complex systems, when systems are characterized as complex systems with high dimension and a variety of variables and factors, to reduce complexity are necessary. Fuzzy cognitive maps (FCM) are a soft computing method for simulation and analysis of complex system, which combines the fuzzy logic with theories of neural networks. It is flexible in system design, model and control, the comprehensive operation and the abstractive representation of behavior for complex systems. When the complexity of the system, the application and maintenance of FCM become more difficult, in particular, the inference is difficult to achieve and not even gets the results. Therefore we present to partition the complex fuzzy cognitive map into smaller chunks based on genetic algorithm in this paper. We construct partitioning rules and criticize rules. Finally, an illustrative example is provided, and its results suggest that the method is capable of partitioning fuzzy cognitive map.

Key-Words: - fuzzy cognitive maps, complex system, genetic algorithm, partitioning, modeling

1 Introduction

There is growing interest in building large knowledge bases and complex system model. Whether the design of knowledge base or the modeling of complex systems, when systems are characterized as complex systems with high dimension and a variety of variables and factors, to reduce complexity are necessary. The design of the knowledge base forms the most crucial part from the performance viewpoint of the expert system. Further, the design of the knowledge base may be divided into three sub-activities [1]: knowledge integration, knowledge base verification and knowledge base partitioning. The total knowledge base can be partitioned into smaller chunks. As each partition forms a portion of the total knowledge base, it is obviously less complex than that of the entire system. This reduction in complexity may aid the verification, validation, and maintenance of the expert system. Further, partitioning helps in developing the modules of an expert system in parallel.

Fuzzy cognitive maps (FCM) are a soft computing method for simulation and analysis of complex system, which combines the fuzzy logic with theories of neural networks. Kosko introduced them as an extension of cognitive maps in 1986 [2]. They were introduced as an extension of Cognitive Maps, and originally applied to problems concerning political science. Their main advantages are flexibility and adaptability to a given domain [3]. In addition, FCM come with a convenient graph representation, in which concepts are represented as nodes of the graph and weighted casual edges represent knowledge about associations between the concepts. This technique is particularly suitable to model qualitative rather than quantitative systems. Quantitative modeling often is not suitable to describe complex systems with strong non-linearities and unknown physical behavior [4]. Being a qualitative approach, FCM are free from most of the drawbacks that are inseparable regarding quantitative modeling techniques.

The genetic algorithm (GA) is a stochastic global search method that mimics the metaphor of natural biological evolution. GA operates on a population of potential solutions applying the principle of survival of the fittest to produce (hopefully) better and better approximations to a solution.

In this paper, we propose a method for the partition of complex system, which is based on genetic algorithm. It aims to provide a means for partitioning complex system

The paper is organized as follows. Section 2 presents an overview of FCM. Section 3 introduces partition problem of FCM. Section 4 introduces genetic algorithm. Section 5 applies the proposed algorithm to partition the FCM. Section 6 is the conclusion and suggestions for future works.

2 Fuzzy Cognitive Map

2.1 Background

Fuzzy Cognitive Maps (FCM), proposed by Kosko [2], are a tool for modeling dynamic systems, which combine elements of neural networks with fuzzy logic. They represent knowledge in a symbolic manner and relate states, variables, events, outputs and inputs using a cause and effect approach. FCM as a result of intuitive knowledge representation, and fast numerical reasoning ability, as well as with neural networks, graph theory, fuzzy logic and other areas in close contact, making it has a wide range application, the studies involve fault detection [5,6,7,8], medical diagnosis [9], management decision-making [10,11,12,13,14,15,16], the analysis of social phenomena [17,18,19,20,21], circuit analysis [22], geographic information systems [24], stock analysis [24], the chess game [25, 26], the control system [27,28,29,30,31], the complex system modelling of other areas [32,33,34,35,36,37]. The scope and range of the applications demonstrate usefulness of this method and motivate further research in this area.

FCM has several desirable properties, such as: it is relatively simple to use for representing structured knowledge [38], and the inference can be computed by numeric matrix operation instead of explicit IF/THEN rules [39]. Most importantly, they are flexible in system design, model and control, the comprehensive operation and the abstractive representation of behavior for complex systems [40]. These advantageous modeling features of FCM encourage us to study and broaden the functionality and applicability of FCM in more problems and systems.

2.2 Formalization of Fuzzy Cognitive Map

An FCM illustrates the model of a system using a graph of concepts and showing the cause and effect among concepts. Each node represents one of the factors of the modeled system. The interconnections among concepts of FCM signify the cause and effect relationship one concept has on the others. These weighted interconnections represent the direction and degree with which concepts influence the value of the interconnected concepts [41,42], and it is described with the weight w_{ij}, its value is usually normalized to the interval [–1,1]. Positive values describe promoting effect, while negative ones describe inhibiting effect. Apart from the graph representation, for computational purposes, a model can be equivalently defined by a square matrix, called connection matrix, which stores all weight values for edges between corresponding concepts represented by rows and columns. The system of n nodes can be represented by n×n connection matrix. An example of FCM model and its connection matrix are shown as follow:

$$W = \begin{bmatrix} 0 & W_{12} & 0 & 0 & 0 & 0 \\ W_{21} & 0 & 0 & 0 & 0 & 0 \\ W_{31} & 0 & 0 & W_{34} & W_{35} & W_{36} \\ 0 & W_{42} & 0 & 0 & 0 & 0 \\ 0 & 0 & W_{53} & W_{54} & 0 & 0 \\ 0 & 0 & 0 & W_{64} & 0 & 0 \end{bmatrix}$$

A connection matrix W can conveniently represents the graph Fig.1

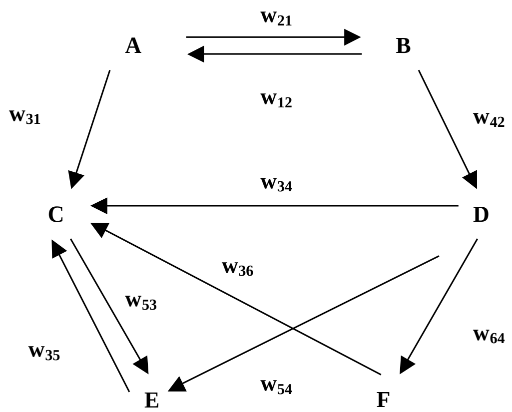

Fig.1 A Fuzzy Cognitive Map

Where wij specifies the value of a for an edge from ith to jth concept node.

From the graphical representation of FCM, it becomes clear that human knowledge and experience are reflected in the selection of concepts and weights for the interconnections between concepts of the FCM.

In order to discuss conveniently, we presented the formal definition FCM as follow:

A FCM consists of nodes-concepts, each node-concept represents one of the key-factors of the system, and it is characterized by a value $C \in (0,1)$, and a causal relationship between two concepts is represented as an edge $\boldsymbol{w_{ij}}$. w_{ij} indicates whether the relation between the two concepts is direct or inverse. The direction of causality indicates whether the concept C_i causes the concept C_j. There are three types of weights:

$\boldsymbol{w_{ij}}>0$ indicates direct causality between concepts C_i and C_j. That is, the increase (decrease) in the value of C_i leads to the increase (decrease) on the value of C_j.

$\boldsymbol{w_{ij}}<0$ indicates inverse (negative) causality between concepts C_i and C_j. That is, the increase (decrease) in the value of C_i leads to the decrease (increase) on the value of C_j.

$\boldsymbol{w_{ij}}=0$ indicates no relationship between C_i and C_j.

In order to discuss conveniently, we presented the formal definition FCM as follows:

A fuzzy cognitive map $\boldsymbol{F}$ is a 4-tuple ($\boldsymbol{V, E, C, f}$) where

--$\boldsymbol{V}=\{v_1, v_2, \dots, v_n\}$ is the set of n concepts forming the nodes of a graph.

--$\boldsymbol{E}:(v_i, v_j)\rightarrow \boldsymbol{w_{ij}}$ is a function $\boldsymbol{w_{ij}} \in \boldsymbol{E}$, $v_i, v_j \in \boldsymbol{V}$, with $\boldsymbol{w_{ij}}$ denoting a weight of directed edge from v_i to v_j. Thus $\boldsymbol{E(V\times V)=(w_{ij})}$ is a connection matrix.

--$\boldsymbol{C}$: $v_i \rightarrow \boldsymbol{C_i}$ is a function that at each concept v_i associates the sequence of its activation degrees, such as $\boldsymbol{C_i(t)}$ given its activation degree at the moment t. $\boldsymbol{C(0)}$ indicates the initial vector and specifies initial values of all concept nodes and $\boldsymbol{C(t)}$ is a state vector at iteration t.

--$\boldsymbol{f}$ is a transformation function, which includes recurring relationship between $\boldsymbol{C(t+1)}$ and $\boldsymbol{C(t)}$.

$$\mathrm{C_i}(\mathrm{t}+1)=\mathrm{f}\left(\left(\sum_{\substack{i=1\\ j\neq i}}^{n} w_{ij} C_j(t)\right)\right) \tag{1}$$

Eq. (1) describes a functional model of FCM. It describes that the value of each concept is calculated by the computation of the influence of other concepts to the specific concept, the transformation function is used to confine the weighted sum to a certain range, which is usually set to [0, 1].

$$\mathrm{o}_i(t+1)=\frac{1}{1+e^{-C(t)}} \tag{2}$$

2.3 Motivation and Objective

Fuzzy cognitive map has a good quality of intelligent control, such as:

1) By the intelligent data-driven, it is consistent with the direction of the development of AI.

2) The FCM is easily established and represents problems directly, it forms a good mapping relationship with the knowledge structure in the minds of the experts of the field, so in a lot of problems the experts often directly establish the FCM system model.

3) It uses numerical reasoning, the relative facts can be reasoned from the nodes of direct connection, without having to traverse the entire Knowledge Base.

4) For any number of knowledge sources, it can construct its own FCM, and can obtain joint distribution knowledge by integrated computation.

5) FCM can represent semantic network, and can also handle the distribution knowledge.

6) Due to FCM combines fuzzy logic with neural networks, it is easy to introduce learning mechanism. It is foundation for intelligence of system. In addition, the description of causality introduces fuzzy measurement, it can naturally and directly express logic meaning of the habit's uses of human, It very adapts to direct or high-level knowledge expression.

7) It has a feedback mechanism so that it can take the modeling for complex dynamic systems, and the structure of tree, Bayes Network and Markov network that is difficult to express the dynamic causality system of feedback.

Furthermore, due to FCM is a weighted direction graph, we can use a lot of the study result of graph theory to analyze the structure and characteristic of FCM.

3 Problem Statement

The partition problem of FCM is an optimal assignation of concept nodes of FCM to different chunks or partitions so as to minimize the switching among partitions. The objective function to be optimized reflects the extent to which the partitions are independent of each other.

In order to reach the objective, the assignment of the nodes-concepts to various partitions has to satisfy the following rule and two constraints.

Suppose a large FCM can be divided into k smaller chunks, $\{\mathrm{G_i}\}$ is a sets of s sub-graph. $\mathrm{G_i}$ is a sub-graph,

$\{\mathrm{G_i}\}$ should satisfy following rule in theory:

$$\bigcup_{i=1}^{k} G_i = G, G_i \cap G_j = \phi, i \neq j, i, j = 1, \cdots k \tag{3}$$

Farther, it is required to satisfy the following two constraints:

the assignment of nodes-concepts must be exhaustive and mutually exclusive, i.e, each and every nodes –concepts must be assigned to exactly one sub-graph.

The size of a sub-graph should not exceed a specified maximum size.

FCM is directed graph, and the relationship of concepts reflects the cause influence between concepts, so we define a new concept 'vector distance' as the relativity measurement between concepts.

Definition: vector distance

Let C_1, C_2,...C_n are the nodes of an FCM, the causal links between nodes are represented by directed weighted edges that illustrate how much one concept influences the interconnected concepts, if the between C_i and C_j exist causal influence w_{ij}, define the cause influence degree w_{ij} for vector distance. If between C_i and C_j do not exist causal relationship, the vector distance is 0.

According to the definition of vector distance, the vector distance between C_i and C_j is corresponding with the element of connection matrix of FCM, so connection matrix FCM construct distance matrix.

4 Proposed Method

The partition problem of FCM is known to be a NP-complete. GA is strong tool to solve NP problem. In this paper, we present a genetic algorithm for partition of FCM.

4.1 Genetic Algorithm

The basic principles of the genetic algorithm (GA) were first proposed by Holland. Thereafter, a series of literature and reports [43,44,45,46,47,48,49,50,51,52,53] became available. GA is inspired by the mechanism of natural selection where stronger individuals are likely the winners in a competing environment. Here, GA uses a direct analogy of such natural evolution. Through the genetic evolution method, an optimal solution can be found and represented by the final winner of the genetic game.

GA presumes that the potential solution of any problem is an individual and can be represented by a set of parameters. These parameters are regarded as the genes of a chromosome and can be structured by a string of values in binary form. A positive value, generally known as a fitness value, is used to reflect the degree of "goodness" of the chromosome for the problem, which would be highly related with its objective value [54].

Throughout a genetic evolution, the fitter chromosome has a tendency to yield good quality offspring, which means a better solution to any problem. In a practical GA application, a population pool of chromosome has to be installed and these can be randomly set initially. The size of this population varies from one problem to another although some guidelines are given in. In each cycle of genetic operation, termed as an evolving process, a subsequent generation is created from the chromosomes in the current population. This can only succeed if a group of these chromosomes, generally called "parents" or a collection term "mating pool" is selected via a specific selection routine. The genes of the parents are mixed and recombined for the production of offspring in the next generation. It is expected that from this process of evolution (manipulation of genes), the "better" chromosome will create a larger number of offspring, and thus has a higher chance of surviving in the subsequent generation, emulating the survival-of-the-fittest mechanism in nature.

Because GA methodology is particularly suited for optimization problems, they have been successfully applied to a wide range of real-world problems of significant complexity. They work with a coding of the parameter sets, not the parameter themselves so that they can get rid of the analytical limitation of search spaces. They only require objective function. They operate on a population of potential solutions applying the principle of survival of the fittest to produce better and better approximations to a solution by three fundamental operators. At each generation, a new set of approximations is created by the process of selecting individuals according to their level of fitness in the problem domain and breeding them together using operators borrowed from natural genetics. This process leads to the evolution of populations of individuals that are better suited to their environment than the individuals that they were created from, just as in natural adaptation [55].

GA differs substantially from more traditional search and optimization methods. The four most significant differences are:

1) GA searches a population of points in parallel, not a single point.

2) GA does not require derivative information or other auxiliary knowledge; only the objective function and corresponding fitness levels influence the directions of search.

3) GA uses probabilistic transition rules, not deterministic ones.

4) GA works on an encoding of the parameter set rather than the parameter set itself (except in where real-valued individuals are used).

Individuals, or current approximations, are encoded as strings, chromosomes, composed over some alphabet(s), so that the genotypes (chromosome values) are uniquely mapped onto the decision variable (phenotypic) domain.

Having decoded the chromosome representation into the decision variable domain, it is possible to assess the performance, or fitness, of individual members of a population. This is done through an objective function that characterises an individual's performance in the problem domain. In the natural world, this would be an individual's ability to survive in its present environment. Thus, the objective function establishes the basis for selection of pairs of individuals that will be mated together during reproduction.

During the reproduction phase, each individual is assigned a fitness value derived from its raw performance measure given by the objective function. This value is used in the selection to bias towards more fit individuals. Highly fit individuals, relative to the whole population, have a high probability of being selected for mating whereas less fit individuals have a correspondingly low probability of being selected.

Once the individuals have been assigned a fitness value, they can be chosen from the population, with a probability according to their relative fitness, and recombined to produce the next generation. Genetic operators manipulate the characters (genes) of the chromosomes directly, using the assumption that certain individual's gene codes, on average, produce fitter individuals. The recombination operator is used to exchange genetic information between pairs, or larger groups, of individuals. The simplest recombination operator is that of single-point crossover.

Consider the two parent binary strings:

P_1 = 1 0 0 1 0 1 1 0, and

P_2 = 1 0 1 1 1 0 0 0.

If an integer position, i, is selected uniformly at random between 1 and the string length, l, minus one [1, l-1], and the genetic information exchanged between the individuals about this point, then two new offspring strings are produced. The two offspring below are produced when the crossover point i = 5 is selected,

O_1 = 1 0 0 1 0 0 0 0, and

O_2 = 1 0 1 1 1 1 1 0.

This crossover operation is not necessarily performed on all strings in the population. Instead, it is applied with a probability *Px* when the pairs are chosen for breeding. A further genetic operator, called mutation, is then applied to the new chromosomes, again with a set probability, *Pm*. Mutation causes the individual genetic representation to be changed according to some probabilistic rule. In the binary string representation, mutation will cause a single bit to change its state, 0 ->1 or 1-> 0. So, for example, mutating the fourth bit of O_1 leads to the new string,

O_{1m} = 1 0 0 0 0 0 0 0.

Mutation is generally considered to be a background operator that ensures that the probability of searching a particular subspace of the problem space is never zero. This has the effect of tending to inhibit the possibility of converging to a local optimum, rather than the global optimum.

After recombination and mutation, the individual strings are then, if necessary, decoded, the objective function evaluated, a fitness value assigned to each individual and individuals selected for mating according to their fitness, and so the process continues through subsequent generations. In this way, the average performance of individuals in a population is expected to increase, as good individuals are preserved and bred with one another and the less fit individuals die out. The GA is terminated when some criteria are satisfied, e.g. a certain number of generations, a mean deviation in the population, or when a particular point in the search space is encountered.

Assume that GA is a 8-tuple (C, E, P0, M, S, R, U, T), where:

C-code
E-fitness function
P0--initialise population
M-population size
S-select operator
R-crossover operator
U-mutation operator
T-stopping condition.

The genetic algorithm can be represented by the following flowchart, see Fig.2

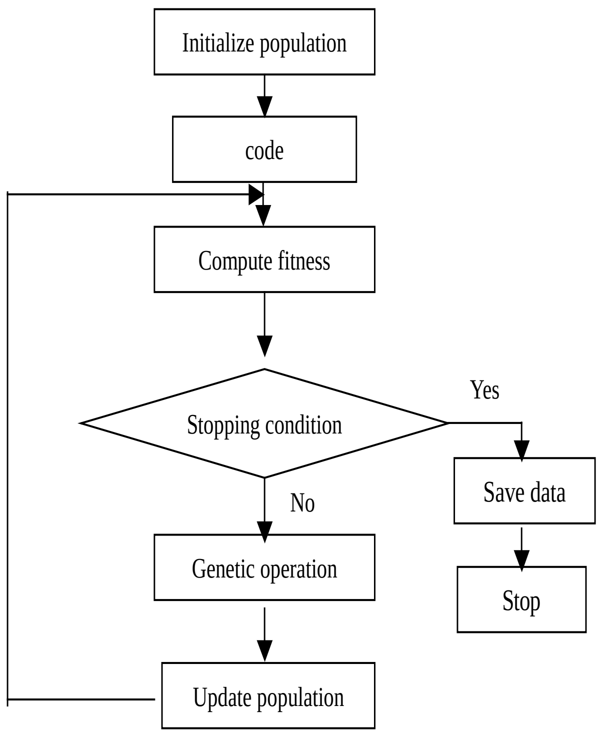

Fig . 2 Flowchart of the algorithm

The cycle of evolution is repeated until a desired termination criterion is reached. The criterion can be set by the number of evolution cycles (computational runs), or by the amount of individuals between different generations, or a pre-defined value of fitness.

4.2 Partition of Fuzzy Cognitive Map

The following sections provide the details to the essential elements of the algorithm, including DNA coding, objective function, stopping condition, and genetic operators.

4.2.1 Chromosome structure

The representation of a solution is crucial for the performance of a GA. The classical genetic algorithms rely on a binary string representation of solutions. But we do not adopt this representation here because of the high time complexity associated with crossover and mutation operations. Instead, we go in for an integer string representation in which fitness evaluation of the solution is straightforward. In the integer string representation, the ordinal value of an integer in the string represents the number of the node while the value represents the identity of the partition number of the corresponding node in the final solution.

4.2.2 Fitness function

In genetic algorithms, the objective function is a measuring mechanism used to evaluate the status of a chromosome.

In order to construct objective function, we define partition variable x.

Suppose a large FCM can be divided into k smaller chunks, if C_i is assigned to partition k, x_{ik}=1, otherwise x_{ik}=0.

$$x_{ik} = \begin{cases} 1 & \text{if } C_i \text{ is assigned to partition k} \\ 0 & \text{otherwise} \end{cases}$$

as a specifically assignment scheme of nodes, x_{ik} will form a X matrix. i is i^{th} node, j is j^{th} sub-graph.

X={ x_{ik} }

For any partition k the quantity $x_{ik}(1-x_{jk})w_{ij}$ is equal to 0 if the two nodes i and j belong to partition k, and is equal to 1 if node i belongs to partition k and node j belongs to other partition. Consequently, the objective function may be formulated as:

$$BC = \sum_{k=1}^{m}\sum_{i=2}^{n}\sum_{j=1}^{i-1} x_{ik}(1-x_{jk})w_{ij} \qquad (4)$$

4.2.3 Stopping condition

Because the GA is a stochastic search method, it is difficult to formally specify convergence criteria. As the fitness of a population may remain static for a number of generations before a superior individual is found, the application of conventional termination criteria becomes problematic. The stopping condition takes into consideration two possible scenarios of the learning process. The learning should be terminated when the fitness function value reaches a threshold value called max-fitness or reaches a maximum number of generations. The algorithm uses the latter.

4.2.4 Genetic operators

The implementation of genetic operations is same as in genetic algorithms. It including the crossover operator and mutation operator that requires the selection of the crossover point and mutation point for each antibody under a predetermined crossover probability and mutation probability. The crossover operator provides search of the sample space to produce good solutions. The mutation operator performs random perturbations to selected solutions to avoid the local optimum.

1) Selection operator: Selection is the process of determining the number of times, or trials, a particular individual is chosen for reproduction and, thus, the number of offspring that an

individual will produce. The selection of individuals can be viewed as two separate processes:

- determination of the number of trials an individual can expect to receive;
- conversion of the expected number of trials into a discrete number of offspring.

The first part is concerned with the transformation of raw fitness values into a real-valued expectation of an individual's probability to reproduce and is dealt with in the previous subsection as fitness assignment. The second part is the probabilistic selection of individuals for reproduction based on the fitness of individuals relative to one another and is sometimes known as sampling. The remainder of this subsection will review some of the more popular selection methods in current usage.

2) Crossover (Recombination) operator: The basic operator for producing new chromosomes in the GA is that of crossover. Like its counterpart in nature, crossover produces new individuals that have some parts of both parent's genetic material. A random crossover point is generated between 0 and maximum vector size of the solution. This point divides the two chromosomes into two parts each. New chromosomes(offspring) are formed by swapping that parts of the parent strings demarcated by the crossover point. After producing the two offspring solution, we check whether they are valid. If a solution is invalid, we discard it.

3) Mutation operator: In natural evolution, mutation is a random process where one allele of a gene is replaced by another to produce a new genetic structure. In GA, mutation is randomly applied with low probability, typically in the range 0.001 and 0.01, and modifies elements in the chromosomes. Usually considered as a background operator, the role of mutation is often seen as providing a guarantee that the probability of searching any given string will never be zero and acting as a safety net to recover good genetic material that may be lost through the action of selection and crossover. The number of mutation points is selected to a maximum of one third the string length.

4.3 The evaluation of partition quality

In order to evaluating quality of partition quantificationally we define cohesiveness, coupling and independence.

Definition: the coupling of sub-graph

The sum of interrelated weight among sub-graphs is called coupling, its measure may be formulated as:

$$BC = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} f(G_i, G_j)$$

$$BC = \sum_{k=1}^{m}\sum_{i=2}^{n}\sum_{j=1}^{i-1} x_{ik}(1-x_{jk})w_{ij} \tag{5}$$

Definition: the cohesiveness of sub-graph

The sum of interrelated weight of ith sub-graph Gi are called cohesiveness, its measure may be formulated as:

$$\boldsymbol{C = \sum_{i=1}^{n} f(G_i, G_i)}$$

$$C_m = \sum_{j=1}^{n}\sum_{i=1}^{n} x_{im}x_{jm}w_{ij} \tag{6}$$

Definition: the coupling of the sub-graph G_i with its complement graph

The sum of interrelated weight of sub-graph G_p with its complement graph G_q are called the coupling of the sub-graph G_i with its complement graph, it may be formulated as:

$$SC = \sum_{j=1}^{n} f(G_i, G_j)$$

$$SC_{pq} = \sum_{q=1, q\neq p}^{m}\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ip}x_{jq}w_{ji} \tag{7}$$

Definition: the quality of partitioning

The ratio of the cohesiveness with the coupling of sub-graph Gp and Gq is defined quality of partitioning, it may be formulated as:

$$I = \frac{C}{SC}$$

$$I = \frac{\sum_{j=1}^{n}\sum_{i=1}^{n} x_{im}x_{jm}w_{ij}}{\sum_{q=1, q\neq p}^{m}\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ip}x_{jq}w_{ji}} \tag{8}$$

5 Application

To demonstrate the feasibility of the proposed method, we adopted following example (T able 1) to validate.

Table 1 The weight of FCM

node	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
1	0	0.8	0	0.8	0	0	0	0	0	0	0	0	0	0	0
2	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0.6	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0.6	0.6	0.6	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0.8	0	0	0.6	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0
8	0	0	0	0	0	0.2	0	0.9	0	0	0	0	0	0	0
9	0	0	1	0	0	0.2	0.9	0.1	0	0	0	0	0	0	0
10	0	0	0.4	0	0	0.2	0	0	0.1	0	0	0	0.4	0.4	0.2
11	0	0	0	0	0	0	0	0	0.2	0	0	0	0.9	0	0.8
12	0	0	0	0	0	0	0	0	0.4	0	0	0	0.8	0.6	0.4
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0.2	0.3
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8
15	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0

The algorithm is implemented in MATLAB with the Genetic Toolbox.

The parameters of the algorithm are reported below: population size: 60, probability of crossover: 0.7, probability of mutation: 0.01, the maximum number of generations: 800.

The partition result is follows:

$$x = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

15 nodes are divided into 3 sub-graphs, 15 nodes assignment case is follows:

Subgraph1: 8,10,11,12,13,14,15

Subgraph2: 1,2,3,4,6

Subgraph3: 7,9

According to (5)-(8), we can grain the evaluation quality of partition quantificationally.

The cohesiveness of sub-graph G_i:

The sub-graph 1: Cm＝5.6

The sub-graph 2: Cm＝5.4

The sub-graph 3: Cm＝1.0

The coupling among sub-graph:

BC＝2.2

The coupling of sub-graph with its complement graph:

The sub-graph 1: SC＝1.2

The sub-graph 2: SC＝1.4

The sub-graph 3: SC＝1.3

The independent of partition:

The sub-graph1:I＝5.6

The sub-graph 2: I＝3.0

The sub-graph 3: I＝0.67

6 Conclusion

We presented a genetic algorithm for solving the partitioning problem of fuzzy cognitive map. We utilized the feature of fuzzy cognitive map to construct partition rules and use GA to partition the knowledge base. The result of study suggests that the method is capable of partitioning fuzzy cognitive map.

References:

[1] Papageorgiou, E., Stylios, C.D., Groumpos, P .P., Fuzzy cognitive map learning based on nonlinear Hebbian rule, *Australian Conf. on Artificial Intelligence*, 2003:256–268.

[2] Kosko, B. Fuzzy Cognitive Maps[J], *International Journal of Man-Machine Studies* 24,pp.65-75, 1986

[3] Aguilar J., A survey about fuzzy cognitive maps papers, *International Journal of Computational Cognition*, vol. 3, no. 2, 2005, pp. 27-33

[4] Campello R.J.G.B., Amaral W.C., Towards true linguistic modelling through optimal numerical solutions, *International Journal of Systems Science*, vol. 34, no. 2, 2003, pp. 139-157

[5] Lee Keesang, on-line Fault Diagnosis by Using Fuzzy Cognitive Map, *Proceeding of the 1995 Joint Technical Conference on Circuits/System Computers and Communications* 1996, 921-922

[6] Pelaez C E., Bowles J B., Applying Fuzzy cognitive Maps Knowledge representation to Failure Modes Effects Analysis, *Information Sciences*. 1996, 88: 177-199

[7] Pelaez C.E., Bowles J.B., Applying fuzzy cognitive maps knowledge representation to failure modes effects analysis, *Proc. IEEE*

Annu. Symp. on Reliability and Maintainability. 1995, 450－456
[8] Wu zhuang, Fault Diagnosis And Reliability Analysis Based on Fuzzy Method, *Beijing Aeronautics and Astronautics University [D]*. 2000
[9] Georgopoulos V.C., Malandraki G..A., Stylios C.D. A fuzzy cognitive map approach to differential diagnosis of space’s language impairment, *J. Artif. Intel. Med.* 2003, 29 (3): 261－278.
[10] Zhi-Qiang L., Satur R., Contextual Fuzzy Cognitive Map for Decision Support in Geographic Information Systems, *IEEE Transactions on Fuzzy Systems*. 1999, 7(5): 494-507
[11] Inwon Kang, Sangjae Leeb, Jiho Choi, Using fuzzy cognitive map for the relationship management in airline service, *Expert Systems with Applications*. 2004, 26: 545–555
[12] Tsadiras A., Margaritis K., Mertzios B., Strategic planning using extended fuzzy cognitive maps, *Studies in Informatics and Control*. 1995. l.4(3):.237-245
[13] Kardaras D., Karakostas B., The use of fuzzy cognitive maps to simulate the information systems strategic planning process, *Information and Software Technology*. 1999, 41: 197-210
[14] Otto X. Cordero and C., Enrique Peláez. Dynamic Business Intelligence. Automated Decision Making with Fuzzy Cognitive Maps, *Business Information Systems. Proceedings of BIS*. 2002
[15] Stylios C.D., Groumpos P.P., The Challenge of Modeling Supervisory Systems Using Fuzzy Cognitive Maps, *Journal of Intelligent Manufacturing* 1998, 9
[16] M. Khan, M. Quaddus., Group decision support using fuzzy cognitive maps for causal reasoning, *Group Decision Negotiation* J. 2004,13 (5): 463－480.
[17] Siraj S. Bridges, Vaughn R. Fuzzy cognitive maps for decision support in an intelligent intrusion detection system, *IFSA World Congr. and 20th NAFIPS Internet. Conf*. 2001, 4: 2165 －2170.
[18] Taber R.., Fuzzy cognitive maps model social systems, *AI Expert*. 1994, 9: 18-23
[19] Cole J.R., Persichitte K.A.. Fuzzy Cognitive Mapping: Applications in Education. International, *Journal of Intelligent Systems*. 2000，15: 1-25
[20] Perusich Karl, Fuzzy Cognitive Maps for Policy Analysis, *Proc. IEEE International Symposium on Technology and Society Technical Expertise and Public Decisions*. 1996, 369-373
[21] Craiger J.P., Coovert M.D., Modeling dynamic social and psychological processes with fuzzy cognitive maps, *Proceedings of the IEEE International Conference on Fuzzy Systems*. 1994, 1873-1877
[22] J.P. Craiger, D.F. Goodman, R.J. Weiss, A.B. Butler, Modeling organizational behavior with fuzzy cognitive maps, *Int. J. Computational Intelligence and Organizations 1*. 1996, 120-123
[23] Satur R. and Zhi-Qiang L., A Contextual Fuzzy Cognitive Map Framework for Geographic Information Systems, *IEEE Transactions on Fuzzy Systems*. 1999, 7(5): 481-494
[24] Z.-Q. Liu, Fuzzy Cognitive Maps in GIS Data Analysis, *Soft Computing*. 2003, 7: 394-401
[25] Koulouriotis D.E., Diakoulakis I.E., Emiris D.M.. A Fuzzy Cognitive Map-Based Stock Market Model, *Synthesis. Analysis and Experimental Results, 10th IEEE International Conference on Fuzzy Systems (FUZZ-2001)* 2001, 465-468
[26] Chen Meien, Huang Yoping, Dynamic fuzzy reasoning model with fuzz cognitive map in Chinese chess, *Proc of the 1995 IEEE International conference on Neural Networks*. Part 3.1995,1353-1357
[27] Chen Meien, Huang Yoping, Guard heuristic by dynamic fuzzy reasoning node for Chinese chess, *3rd international symposium on uncertainty modeling and analysis and annual conference of the north American fuzzy information processing society*. 1995, 530-633
[28] Groumpos P.P., Stylios C.D., Modeling Supervisory Control Systems Using Fuzzy Cognitive Maps, *Chaos. Solitons And Fractals*. 2000, 11: 329-336
[29] Gotoh K., Murakami J., Yamaguchi T., Yamanaka Y, Application of fuzzy cognitive maps to supporting for plant control, *Proc. SICE Joint Symp. 15th Systems Symp. And Tenth Knowledge Engineering* Symp.1989, 99-104
[30] Peter P. Groumpos, Chrysostomos D. Stylios, Modeling supervisory control systems using fuzzy cognitive maps Chaos, *Solitions and Fractals* 11. 2000, 329-336
[31] Sheridan T.B., Supervisory control in: G. Salvendy (Ed.), *Handbook of Human Factors*. Wiley. New York. 1987, 1243-1268.

[32] Sangjae Lee, Ingoo Han, Fuzzy cognitive map for the design of EDI controls, *Information & Management*. 2000, 37: 37-50
[33] Koulouriotis D.E., Diakoulakis I.E., Emiris D.M., Zopounidis C.D., Development of dynamic cognitive networks as complex systems approximates, *validation in financial time series Applied Soft Computin*g. 2005, 5: 157–179
[34] Stach W., L. Kurgan, Modeling software development project using fuzzy cognitive maps, *Proc. 4th ASERC Workshop on Quantitative and Soft Software Engineering* (QSSE'04). 2004, 55－60.
[35] Stach W., Kurgan L., Pedrycz W., Reformat M, parallel fuzzy cognitive maps as a tool for modeling software development project, *Proc. 2004 North American Fuzzy Information Processing Society Conf. (NAFIPS'04)*. Banff. AB. 2004, 28-33
[36] Stylios C.D., Groumpos P .P., Modeling complex systems using fuzzy cognitive maps, *IEEE Trans. Systems Man. Cybern. Part A: Systems Humans*. 2004, 34 (1)
[37] E. Koulouriotis, I.E. Diakoulakis, D. M. Emiris, et al., Efficiently modeling and controlling complex dynamic systems using evolutionary fuzzy cognitive maps (Invited Paper), *Internat. J. Comput. Cognition 1 (2)* , 2003: 41–65.
[38] K.E. Parsopoulos,E.I. Papageorgiou,P .P. Groumpos,M.N. Vrahatis, A first study of fuzzy cognitive maps learning using particle swarm optimization, *Proc. IEEE 2003 Congr. on Evolutionary Computation*,2003, pp. 1440–1447.
[39] D.E. Koulouriotis,I.E. Diakoulakis,D.M. Emiris,E.N. Antonidakis,I.A. Kaliakatsos,Efficiently, modeling and controlling complex dynamic systems using evolutionary fuzzy cognitive maps (Invited Paper), *Internat. J. Comput.* Cognition 1 (2) (2003) 41–65.
[40] A.K. Tsadiras,K. Margaritis, An experimental study of the dynamics of the certainty neuron fuzzy cognitive maps, *Neurocomputing* 24 (1999) 95–116.
[41] Bart Kosko, *Fuzzy Engineering [M]*, Prentice Hall, 1997
[42] John F. Sowa, *Knowledge Representation*, Beijing: Mechanism Industry Publishing Company, 2003
[43] NODA E, FREITAS A A, LOPES H S. Discovering interesting prediction rules with a genetic algorithms, *Proc. Congress on Evolutionary Computation(CEC-99)*, Washington D C, USA: 1322-1329.
[44] Qing-Sheng R., Zhong-Xing Y., Genetic algorithms used in the analysis of the operator [J], *Electronic Journal*, 2000; 28 (5): 113 ~ 114
[45] A. Taneda, Cofolga: A genetic algorithm for finding the common folding of two RNAs, *Computational Biology and Chemistry*, vol. 29, no. 2, pp. 111-119, 2003.
[46] M. H. Fatemi, M. Jalali-Heravi, and E. Konuze, Prediction of bioconcentration factor using genetic algorithm and artificial neural network, *Analytica Chimica Acta*, vol. 486, no. 1, pp. 101-108, 2003.
[47] R. Ruiz, C. Maroto, and J. Alcaraz, Two new robust genetic algorithms for the flow shop scheduling problem, *Int. J. of Management Science Omega*, vol. 34, no. 5, pp. 461-476, 2006.
[48] R. M. Jarvis and R. Goodacre, Genetic algorithm optimization for pre-processing and variable selection of spectroscopic data, *Bioinformatics*, vol. 21, no. 7, pp. 860-868, 2005.
[49] T.-C. Chen and G.-W. Fischer, A GA-based search method for the tolerance allocation problem, *Artificial Intelligence*, vol. 14, no. 2, pp. 133-141, 2000.
[50] T. T. H. Ng and G. S. B. Leng, Application of genetic algorithms to conceptual design of a micro-air vehicle, *Engineering Applications of Artificial Intelligence*, vol. 15, no. 5, pp. 439-445, 2002.
[51] H. Hirani and N. P. Suh, Journal bearing design using multi-objective genetic algorithm and axiomatic design approaches, *Tribology International*, vol. 38, no. 5, pp. 481-491, 2002.
[52] E. G. Carrano, L. A. E. Soares, R. H. C. Takahashi, and O. M. Neto, Electric distribution network multiobjective design using a problem-specific genetic algorithm, *IEEE Trans. On Power Delivery*, vol. 21, no. 2, pp. 995-1005, 2006.
[53] K. Huang and Z.-W. Xu, Research on the application of improved genetic algorithm in mechanical optimization and design, *Modern Manufacturing Engineering*, vol. 10, no. 5, pp. 80-81, 2005.
[54] Zhang Lin, Zhang Bo, Genetic Algorithm Mechanism Study, *Software Transaction*, 2000,11(7):945-952
[55] Papageorgiou E.I., Groumpos P.P., Optimization of fuzzy cognitive map model in clinical radiotherapy through differential evolution algorithm, *Biomed. Soft Compute*. Um. Sci. 2004, 9(2)