Hindawi
Complexity
Volume 2019, Article ID 2635714, 16 pages
https://doi.org/10.1155/2019/2635714

Research Article

Automatic Implementation of Fuzzy Reasoning Spiking Neural P Systems for Diagnosing Faults in Complex Power Systems

Haina Rong,[1] Kang Yi,[1] Gexiang Zhang ,[1] Jianping Dong,[1] Prithwineel Paul,[1] and Zhiwei Huang[2]

[1] *School of Electrical Engineering, Southwest Jiaotong University, Chengdu 610031, China*
[2] *Beijing National Railway Research & Design Institute of Signal & Communication Group Co., Ltd., Chengdu Branch, Chengdu 610000, China*

Correspondence should be addressed to Gexiang Zhang; gexiangzhang@gmail.com

Received 7 February 2019; Accepted 7 May 2019; Published 19 June 2019

Academic Editor: Quanmin Zhu

As an important variant of membrane computing models, fuzzy reasoning spiking neural P systems (FRSN P systems) were introduced to build a link between P systems and fault diagnosis applications. An FRSN P system offers an intuitive illustration based on a strictly mathematical expression, a good fault-tolerant capacity, a good description for the relationships between protective devices and faults, and an understandable diagnosis model-building process. However, the implementation of FRSN P systems is still at a manual process, which is a time-consuming and hard labor work, especially impossible to perform on large-scale complex power systems. This manual process seriously limits the use of FRSN P systems to diagnose faults in large-scale complex power systems and has always been a challenging and ongoing task for many years. In this work we develop an automatic implementation method for automatically fulfilling the hard task, named membrane computing fault diagnosis (MCFD) method. This is a very significant attempt in the development of FRSN P systems and even of the membrane computing applications. MCFD is realized by automating input and output, and diagnosis processes consists of network topology analysis, suspicious fault component analysis, construction of FRSN P systems for suspicious fault components, and fuzzy inference. Also, the feasibility of the FRSN P system is verified on the IEEE14, IEEE 39, and IEEE 118 node systems.

1. Introduction

As a branch of natural computing, membrane computing was introduced by Păun in 1998 [1, 2]. The distributed parallel computational model is called a membrane system or a P system. Membrane computing aims to investigate the computational models and their applications abstracted from the structure and functioning of cells [2–5]. A large number of studies [6–10] show that many variants of P systems are Turing complete [11–14]. Moreover, distribution, maximally parallel, and expansibility [3, 15–18] make P systems suitable for solving a variety of practical problems, like engineering optimization [3, 19], combinatorial optimization [20], and membrane controllers [21–24].

With the development of membrane computing, many types of membrane systems were proposed [12, 13, 25, 26], of which spiking neural P systems (SN P systems) are a hot research topic covering language generation [27, 28], computing power [29, 30], fuzzy reasoning [31–33], and NP-hard problems [28, 30, 34]. SN P systems were introduced by Ionescu et al. in 2006 [13]. As a typical application type of SN P systems, fuzzy reasoning spiking neural P systems (FRSN P systems) were introduced to build a bridge between the P systems and fault diagnosis for complex power systems [3, 24]. An FRSN P system offers an intuitive illustration based on a strictly mathematical expression, a good fault-tolerant capacity, a good description for the relationships between protective devices and faults, and an understandable diagnosis model-building process [3, 35]. According to the investigations reported in literature, FRSN P systems have been successfully used to diagnose the faults occurring in transformers [24, 36], power transmission networks [35, 37–39], traction power supply systems of high-speed railways

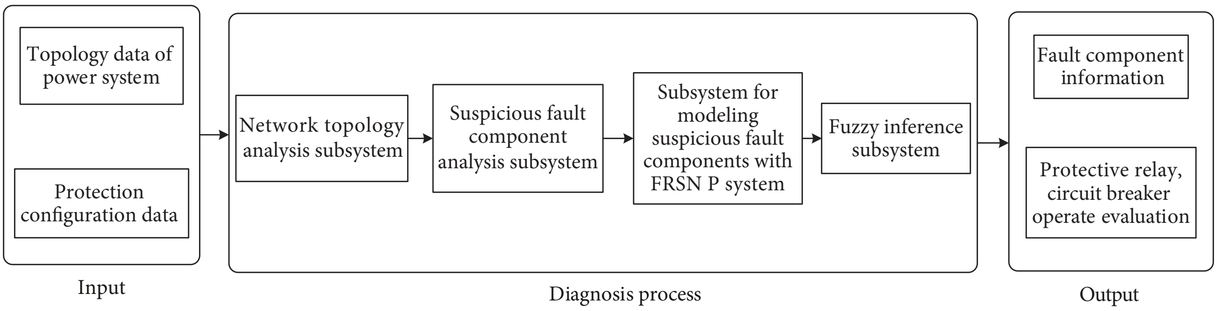

Topology data of power system
Protection configuration data
Input
Network topology analysis subsystem
Suspicious fault component analysis subsystem
Subsystem for modeling suspicious fault components with FRSN P system
Fuzzy inference subsystem
Diagnosis process
Fault component information
Protective relay, circuit breaker operate evaluation
Output

Figure 1: Schematic structure of MCFD.

[40], metro traction power systems[41], and fault classification of power transmission lines [39].

Until now, the implementation of FRSN P systems is still at a manual process, which seriously limits the use of FRSN P systems to diagnose faults in large-scale complex power systems. For example, in [3, 24, 35–37, 42], only a small-scale power system was considered. Furthermore, it is impossible to perform complexity analysis for FRSN P systems with a manual diagnosis process. These problems result in the difficulty in the process of comparisons of FRSN P systems with other diagnosis methods. Thus, the automatic implementation of FRSN P systems has always been a challenging and ongoing task for many years.

In this paper, an automatic implementation method has been developed for automatically fulfilling the task and it is called membrane computing fault diagnosis (MCFD) method. This is a very significant step in this research direction. The method automatizes input, output, and four diagnosis processes: network topology analysis, suspicious fault component analysis, construction of FRSN P system for suspicious fault components, and fuzzy inference.

The article is arranged in the following manner: Section 2 discusses MCFD, Section 3 discusses the experimental results of the system performed on IEEE14, IEEE 39, and IEEE 118 node system, Section 4 analyzes the complexity of the system, and Section 5 concludes this work.

2. MCFD

Membrane computing fault diagnosis (MCFD) method automatizes mainly three components, i.e., input, output, and diagnosis process. The main components of the diagnosis process consist of network topology analysis, suspicious fault component analysis, construction of FRSN P system for suspicious fault components, and fuzzy inference. The input data is composed of topology data of power systems and protection configuration data. The outputs include fault component information, protective relays information, and circuit breakers operation evaluation. The schematic structure of MCFD is shown in Figure 1.

2.1. Automatized Input. The information source of the fault diagnosis program is the grid static data and switch state data based on fault information system. In this paper we use access database to store network topology information and protection configuration information. The static topology information and protection configuration information of the power network are given as input into the access database to form the topology table and protection configuration table. Therefore, the input data of FRSN P system for fault diagnosis consists of two parts: topology data of a power system and protection configuration data.

2.1.1. Topology Data of Power Systems. In this paper we mainly discuss the fault diagnosis methods in power transmission network. Moreover, this paper improves the traditional line analysis [43] and redefines the components, i.e., transmission lines, busbars, transformers, and generators. The circuit breakers (CBs) work as switches.

The “Component” and “Switch” that appear in the following sections are defined in the following manner. A power system is made up of components and switch devices which connect a variety of other equipment. So, the whole electric power system grid network can be represented by the power transmission network topology as shown in Figure 2. The components shown in the figure refer to the transmission lines, busbars, transformers, and generators. The switches refer to the circuit breakers with two states: open and closed.

After there is a failure, because of the grid power system components and complex wiring, it is very difficult to find the faults in huge systems. But whenever fault occurs in a power system, the protective relays and circuit breakers will operate to isolate the fault. We investigate the actions of protective relays and tripped circuit breakers in the network and the connection relationship between them. Figure 2 shows a simple and clear power transmission network topology of the connection relation between components and circuit breakers. Then the fault component is searched according to the tripped circuit breakers.

The topology table (Table 1) is constructed from the components and switches of the entire power transmission network topology. The table stores the data of the main components and switches along with their connection relationships and the protection number associated with each component.

2.1.2. Protection Configuration Data. The SCADA system can provide the tripped circuit breakers and operated protective relays information whenever there is a fault in the grid. Moreover, the data of the components, protective relays,

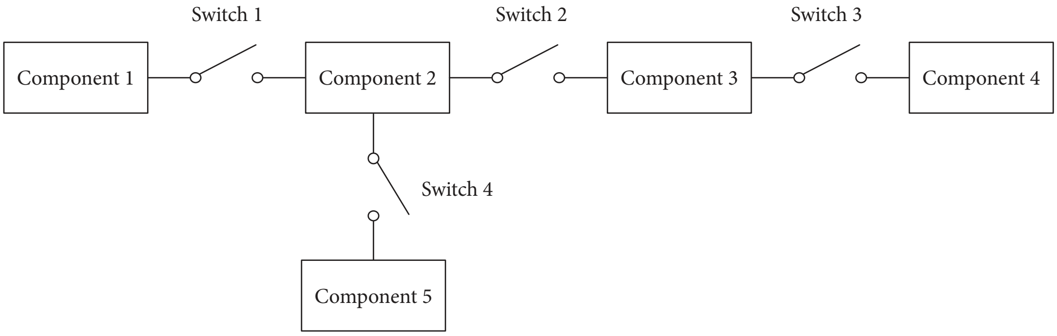


Figure 2: Schematic diagram of power transmission network topology.

Table 1: The topology table.

ID number	Switches operation	Component/switch	Type	Type number	Associated components/switches	Associated protective relays
10108	- -	B08	Bus	101	CB0807.CB08G2	10110108. 21310214

Table 2: The protection configuration table.

ID number	Protective relay	Relays operation	Operated switches	Protected equipment
10110113	Main_ protective relay	0	010426.10438.10439	B13

and circuit breakers are used to construct the correlation database. The correlation relationships are introduced in [44–46] in the following manner:

"Component - Protective" relay means that the protective relays can be divided into the main protective relays of the component and one of the first backup protective relays.

"Protective relay – Switch" means that the circuit breaker can trip in principle once the protective relays operation is performed.

"Component1 - Component2" relates to the scope of protection of the second backup protective relay of component1 which can protect component2.

With these correlations, the protection configuration table can be described as in Table 2.

2.2. Automatized Network Topology Analysis. After the failure in the power system transmission network, the fault component is eventually isolated by tripped circuit breakers. Moreover, the fault component will be isolated in the passive network. We have elaborated in Section 2.1 the entire topology database and protective relay database. Also, we have established the corresponding topology table which represents the correlation between the whole transmission network topology structure and protective relays. Whenever a failure occurs in the power system transmission network, at first the information is received from SCADA system by circuit breaker opening and closing state, and then the suspicious fault component is found by using the network topology analysis method [47, 48]. The specific network topology analysis method is as follows:

(1) Set up M and store all component IDs into M.

(2) Set up the subset N. Take a component from the set M and put it in the subset N. Find all closed circuit breakers connected to it. If there is no closed circuit breaker, then turn to step (5).

(3) Identify the components connected to the closed circuit breaker and add the found components to the subset N.

(4) Continue to search for closed circuit breakers that are connected to the components in step (3) (except for circuit breakers used in step (3)). If there is a closed circuit breaker, go to step (3).

(5) Remove all components in the set M that appear in the subset collection N. If the set M is not empty, then transfer to step (2).

(6) List all the subsets N.

In fault diagnosis, the network topology analysis method is used to search all subsets, and then the passive networks are found from these subsets. These passive networks are the outage areas. In this way, the diagnosis scope can be reduced and then the suspicious fault component is diagnosed. It also reduces the amount of operations and improves the efficiency of fault diagnosis. The process of the searching of the passive networks is shown in Figure 3.

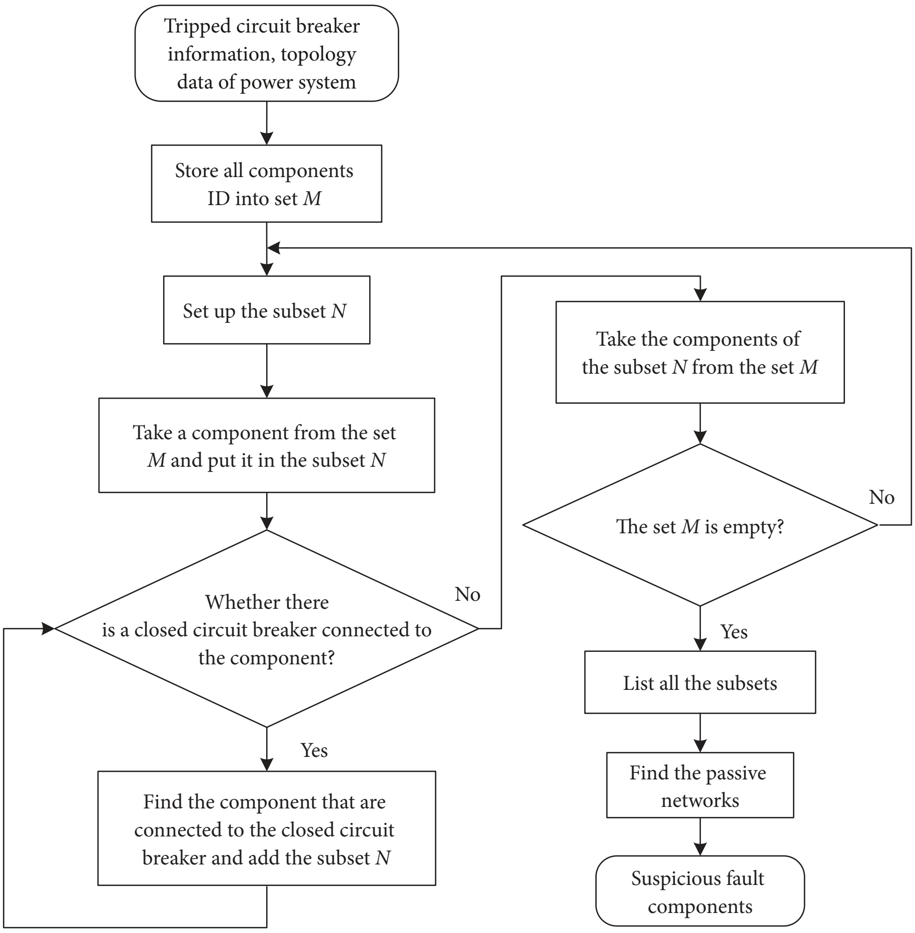

Tripped circuit breaker information, topology data of power system
Store all components ID into set M
Set up the subset N
Take a component from the set M and put it in the subset N
Whether there is a closed circuit breaker connected to the component?
No
Yes
Find the component that are connected to the closed circuit breaker and add the subset N
Take the components of the subset N from the set M
The set M is empty?
No
Yes
List all the subsets
Find the passive networks
Suspicious fault components

Figure 3: Search the outage area flow chart.

2.3. Automatized Suspicious Fault Component Analysis. The network topology analysis method is used to find a passive network and the diagnosis of the suspicious fault components in the passive network. The modeling of FRSN P systems is very complex because of the existence of many components in the complex grid network. In order to improve the diagnosis efficiency and accuracy of the algorithm, in this paper, we introduce the concept of suspected fault component logic analysis. At first a logic diagram is constructed in such a manner that the suspected fault component is considered as the starting point. Then it searches and builds towards each connection to protect the component within the scope of the protection of all components and switches. The FRSN P system model is constructed according to the fault production rules of the suspected fault components. Hence the fault area is reduced and the fault component is identified.

In the logic diagram the suspicious component along with other system components and switches in the passive network are represented by a node and the edge between the two nodes represents the connection between the components and switches. Moreover, the condition of the path search termination is

(1) The search is complete when all the protective relays and switches that can protect the suspicious fault components are searched on different paths.

(2) If the search path is disconnected from the peripheral device due to normal operations (such as the operation of the blade, etc.), then the search will be terminated.

(3) If the search direction is opposite to the rule, the search path will terminate.

(4) Search for a loop structure or parallel edge structure on the search path, and if it exists then terminate this direction search.

The logic diagram of the suspected fault component describes the topological association of the suspected fault components and its associated protection in the power grid. The following example illustrates the method of forming the logic diagram of the suspected component. In Figure 4, it is assumed that the suspicious fault component is B3 by the method of network topology analysis. The logic diagram of the suspected fault component is established by bus B3 in three outgoing line paths: $B_3 \longrightarrow CB_5 \longrightarrow L_3 \longrightarrow CB_2$; $B_3 \longrightarrow CB_6 \longrightarrow L_4 \longrightarrow CB_7$; $B_3 \longrightarrow CB_9 \longrightarrow L_5 \longrightarrow CB_{10}$,

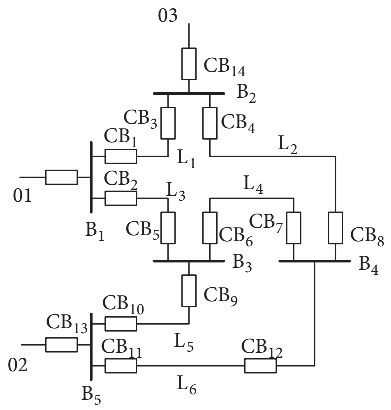
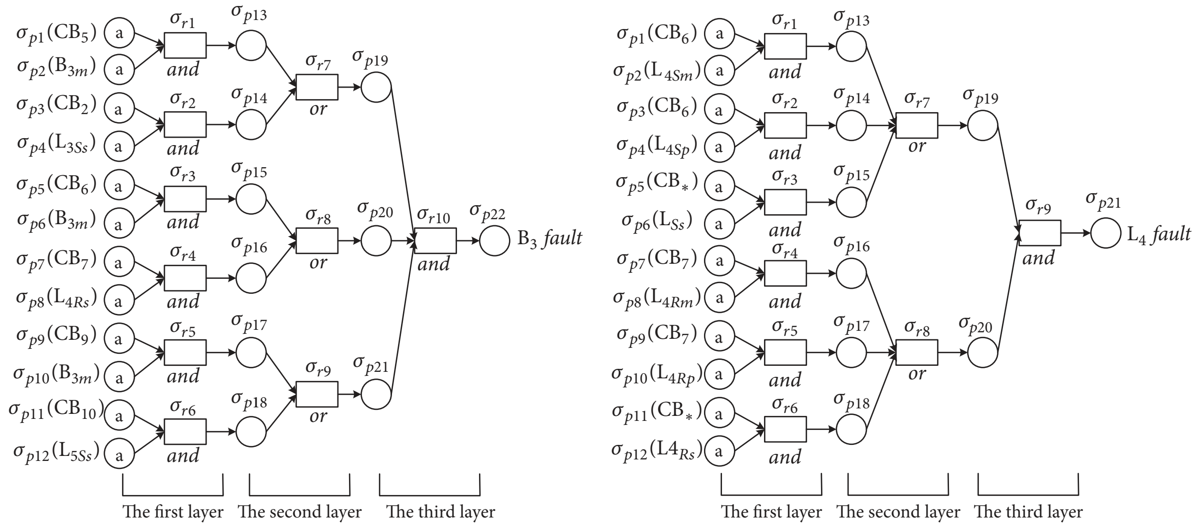


Figure 4: A local grid.



(a) B_3 FRSN P system model

(b) L_4FRSN P system model

Figure 5: Suspect fault components FRSN P system model.

respectively. Moreover, the mutual cooperation between the protective relay and circuit breaker will cut off the connection with the whole grid.

2.4. Automatized Modeling Suspicious Fault Components with FRSNP System. Before performing the reasoning algorithm, we need to build a FRSN P system diagnosis model. A local grid is shown in Figure 4 and the network topology analysis subsystem obtains the bus B_3 as the suspicious component. Also, the bus B_3 and line L_4 are used to build the FRSN P system fault diagnosis model.

At first, the bus B_3 in Figure 4 is used to describe the model where the fault confidence level of bus B_3 is the value of output of the FRSN P system. Moreover, we discussed the mutual cooperation between the protective relay and circuit breaker by the suspicious fault component logic diagram in Section 2.3. The fault production rules of bus B_3 are described as follows:

$\boldsymbol{R}_1$: IF (B_{3m} operates and CB_5 trips) OR (L_{3Ss}operates and CB_2 trips) THEN B_3faults ($CF = c_i$),

$\boldsymbol{R}_2$: IF (B_{3m} operates and CB_6 trips) OR (L_{4Ss} operates and CB_7 trips) THEN B_3 faults ($CF = c_i$),

$\boldsymbol{R}_3$: IF (B_{3m} operates and CB_9 trips) OR (L_{5Ss} operates and CB_{10} trips) THEN B_3 faults ($CF = c_i$).

The following fault diagnosis model based on FRSN P system for bus B_3 is built according to these fault production rules shown in Figure 5(a). The FRSN P system Π is a construct of the form

$$\Pi = \left(O, \sigma_{p1}, \dots, \sigma_{p20}, \sigma_{r1}, \dots, \sigma_{r10}, syn, in, out\right) \quad (1)$$

where

(1) $O = \{a\}$ is the singleton alphabet (a is called spike);

(2) $\sigma_{p1}, \dots, \sigma_{p20}$ are proposition neurons corresponding to the propositions with fuzzy truth values $\theta_1, \theta_2, \dots, \theta_{22}$;

(3) $\sigma_{r1}, \dots, \sigma_{r10}$ are rule neurons, where $\sigma_{r1}, \dots, \sigma_{r6}$, and σ_{r10} are *and* rule neurons and σ_{r7}, σ_{r8}, and σ_{r9} are *or* rule neurons. A real number $c_i \epsilon [0, 1]$ is used

Table 3: Confidence levels of operate protective devices.

Components Name	Main		First backup		Second backup	
	Relays	CBs	Relays	CBs	Relays	CBs
Bus	0.8564	0.9833	-	-	0.7	0.75
Line	0.9913	0.9833	0.8	0.85	0.7	0.75

Table 4: Confidence levels of nonoperate protective devices.

Components Name	Main		First backup		Second backup	
	Relays	CBs	Relays	CBs	Relays	CBs
Bus	0.2	0.2	-	-	0.2	0.2
Line	0.2	0.2	0.2	0.2	0.2	0.2

to represent the certainty factor (CF) of the fuzzy production rule associated with $\sigma_{ri}(1 \leq i \leq 10)$;

(4) $syn \subseteq \{1, 2, \ldots, 22\} \times \{1, 2, \ldots, 10\}$ with $i \neq j$ for all $(i, j) \in syn, 1 \leq i, j \leq 22$, is a directed graph of synapses between the linked neurons;

(5) $in = \{\sigma_{p1}, \ldots, \sigma_{p12}\}$, $out = \{\sigma_{r10}\}$.

The transmission line L_4 in Figure 4 is used to describe model building of transmission line, where the fault confidence level of transmission line L_4 is the value of output of the FRSN P system. The fault production rules of transmission line L_4 are described as follows:

$\boldsymbol{R}_1$: IF (L_{4Sm} operates and CB_6 trips) OR (L_{4Sp}operates and CB_6 trips) OR (second backup protection operates and CB trips) THEN L_4 faults ($CF = c_i$),

$\boldsymbol{R}_2$: IF (L_{4Rm} operates and CB_7 trips) OR (L_{4Rs} operates and CB_7 trips) OR (second backup protection operates and CB trips) THEN L_4 faults ($CF = c_i$).

Therefore, fault diagnosis model based on FRSN P system for transmission line L_4 is built according to the fault production rules shown in Figure 5(b).

In order to make the reasoning reflect the operation of the actual power grid more accurately, the uncertain factors in the protection and the circuit breaker action information are obtained from the power system dispatching center. In this paper, the initial value of the confidence level for operate and nonoperate protective relays and circuit breakers are given in [49], as shown in Tables 3 and 4, respectively. At the same time, considering the uncertainty of the rule credibility, the certainty factor c_i of each fuzzy production rule is considered to be 0.95.

In this paper, a proposition neuron is used for all second backup protective relays and circuit breakers at both ends of the line. If there are multiple second backup protective relays, a factor μ_1, μ_2 as (2), (3) is applied before the confidence level of the proposition neuron, and the two ends of the line (S end and R end), respectively.

$$\mu_1 = \frac{Number\ of\ protective\ relays\,(CB)\ \ for\ S\ end\ operation}{Number\ of\ all\ protective\ relays\,(CBs)\ \ on\ the\ S\ end} \tag{2}$$

$$\mu_2 = \frac{Number\ of\ protective\ relays\,(CB)\ \ for\ R\ end\ operation}{Number\ of\ all\ protective\ relays\ \ (CBs)\ \ on\ the\ R\ end} \tag{3}$$

The automatic generation of the FRSN P system model is shown below.

Step 1. According to the suspected fault component logic diagram, take one path of the suspected fault component logic diagram, and the components and switches involved in the path can be expressed as a fault production rule.

Step 2. Set up two set θ and δ, where θ is the proposition neuron corresponding to the proposition with fuzzy truth values. The initial value setting of the corresponding protective relay and circuit breaker of the first layer of proposition neurons in the FRSN P system model is set by the information of the input from SCADA system. δ is the certainty factor which is added to describe the credibility of the fuzzy generated rules of the neuron.

Step 3. Store the values of θ and δ according to the confidence levels of operate and nonoperate protective relays and circuit breakers in the first path.

Step 4. Repeat the first three steps until all the values of θ and δ represented by the paths are added corresponding to one branch direction.

2.5. Automatized Fuzzy Inference. After obtaining the confidence level of the proposition expressed by the proposition neuron and the certainty factor value of the rule neuron, the next step is to carry out the reasoning operation. By executing the following reasoning algorithm, the fuzzy values of the propositions are represented by the output proposition neurons which can be obtained quickly and simply. Specific algorithm steps [35] are as follows (where s represents the number of proposition neurons, t represents the number of rule neurons and $s + t = m$).

Step 1. Set the initial state to $g = 0$. Set the termination condition to $0_1 = (0, \ldots, 0)^T_{1\times t}$. The initial values of θ and δ

are set to $\theta_g = (\theta_{1g}, \theta_{2g}, \ldots, \theta_{sg})^T$ and $\delta_g = (\delta_{1g}, \delta_{2g} \ldots, \delta_{tg})^T$, respectively.

Step 2. g is increased by one.

Step 3. The firing condition of each input neuron $g = 1$ or each proposition neuron ($g > 1$) is evaluated. If the condition is satisfied and there is a presynaptic rule neuron, the neuron fires and transmits a spike to the next rule neuron. Compute the fuzzy truth value vector δ_g according to (4):

$$\delta_g = D_1^T \otimes \theta_{g-1} + D_2^T \oplus \theta_{g-1} + D_3^T \odot \theta_{g-1} \qquad (4)$$

Step 4. If $\delta_g = 0_1$, then the algorithm stops and output the reasoning results. Otherwise, the algorithm continues.

Step 5. Evaluate the firing condition of each rule neuron. If the condition is satisfied, then the rule neuron fires and transmits a spike to the next proposition neuron. Next, compute the fuzzy truth value vector θ_g according to (5). Then, the algorithm goes to Step 2:

$$\delta_g = E^T \odot \left(C \otimes \delta_g\right) \qquad (5)$$

Parameter vectors and matrices are described in the following manner:

(1) $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_s)^T$ is a real truth value vector of s proposition neurons. $\theta_i (1 \leq i \leq s)$ is a real number in $[0, 1]$ and represents the potential value contained in the ith proposition neuron. If there is no spike in a proposition neuron, its potential value is 0.

(2) $\boldsymbol{\delta} = (\delta_1, \delta_2 \ldots, \delta_t)^T$ is a real truth value vector of t rule neurons. $\delta_t (1 \leq j \leq t)$ is a real number in $[0, 1]$ and represents the potential value contained in the jth rule neuron. If there is no spike in a rule neuron, its potential value is 0.

(3) $\boldsymbol{C} = \operatorname{diag}(c_1, c_2, \ldots, c_t)$ is a diagonal matrix, where $c_t (1 \leq j \leq t)$ is a real number in $[0, 1]$ representing the certainty factor of the jth fuzzy production rule.

(4) $\boldsymbol{D}_1 = (d_{ij})_{s \times t}$ is a synaptic matrix representing the directed connection from proposition neurons to general rule neurons. If there is a directed arc (synapse) from the proposition neuron σ_i to the *general* rule neuron σ_j, $d_{ij} = 1$; otherwise, $d_{ij} = 0$.

(5) $\boldsymbol{D}_2 = (d_{ij})_{s \times t}$ is a synaptic matrix representing the directed connection from proposition neurons to and rule neurons. If there is a directed arc (synapse) from the proposition neuron σ_i to the *and* rule neuron σ_j, $d_{ij} = 1$; otherwise, $d_{ij} = 0$.

(6) $\boldsymbol{D}_3 = (d_{ij})_{s \times t}$ is a synaptic matrix representing the directed connection from proposition neurons to and rule neurons. If there is a directed arc (synapse) from the proposition neuron σ_i to the *or* rule neuron σ_j, $d_{ij} = 1$; otherwise, $d_{ij} = 0$.

(7) $\boldsymbol{E} = (e_{ji})_{t \times s}$ is a synaptic matrix representing the directed connection between rule neurons and proposition rule neurons. If there is a directed arc (synapse) from the rule neuron σ_j to the proposition neuron σ_i, $e_{ji} = 1$; otherwise, $e_{ji} = 0$.

Subsequently, we introduce the following three multiplication operations:

(1) $D^T \otimes \theta = (\overline{d}_1, \overline{d}_2, \ldots, \overline{d}_t)^T$, where $\overline{d}_j = d_{1j} * \theta_1 + d_{2j} * \theta_2 + \cdots + d_{sj} * \theta_s$, $j = 1, 2, \ldots, t$.

(2) $D^T \oplus \theta = (\overline{d}_1, \overline{d}_2, \ldots, \overline{d}_t)^T$, where $\overline{d}_j = \min(d_{1j} * \theta_1, d_{2j} * \theta_2, \ldots, d_{sj} * \theta_s)$, $j = 1, 2, \ldots, t$.

(3) $D^T \odot \theta = (\overline{d}_1, \overline{d}_2, \ldots, \overline{d}_t)^T$, where $\overline{d}_j = \max(d_{1j} * \theta_1, d_{2j} * \theta_2, \ldots, d_{sj} * \theta_s)$, $j = 1, 2, \ldots, t$.

2.6. Automatized Output. The outputs include fault component information, protective relays information, and circuit breakers operation evaluation. After the fuzzy value of the output proposition neuron is obtained according to the fault threshold, the suspected fault components can be determined if a failure has occurred. If it is a faulty component, the system outputs the fault component information and fault credibility value and goes to the reasoning process to estimate whether the protection and circuit breaker has the situation of misoperation and misoperation or not. If it is a normal component, the output appears normal.

Thus, the network topology analysis algorithm to find the passive region and the diagnosis of the suspected fault components in the passive region are repeated. At first, the suspicious fault component analysis subsystem is called to form the logic diagram of the suspicious fault components, and then the FRSN P system diagnosis model is performed by using the query protection configuration data. Finally, by calling FRSN P system inference algorithm, the fault confidence levels of the suspected fault components are obtained and the fault of the components is determined to realize the whole diagnosis process.

3. Experiments

To verify the effectiveness of the automatic implementation, the IEEE14 node power system network model shown in Figure 6 is tested. It is a 14-bus system containing B_{01}~B_{14} 14 buses and L_{0102}~L_{1314} 20 lines. The protective device consists of 134 protective relays and 40 circuit breakers, and $B_{01m}, \ldots, B_{14m}$ are the main protective relays of the buses. Moreover, L_{XSm} and L_{XRm} are the main protective relays on both ends of the line, L_{XSp} and L_{XRp} are the first backup protective relays on both ends of the line, and L_{XSs} and L_{XRs} are the second backup protective relays on both ends of the line (where X represents the ID of the line). Before diagnosing the suspected fault component, the components and switches of the entire power transmission network must adopt the method described in the input data section to construct their respective topology table. The detailed protection configuration of various components, as well as the relationship between the protective relays and the circuit breakers, has been represented by the corresponding

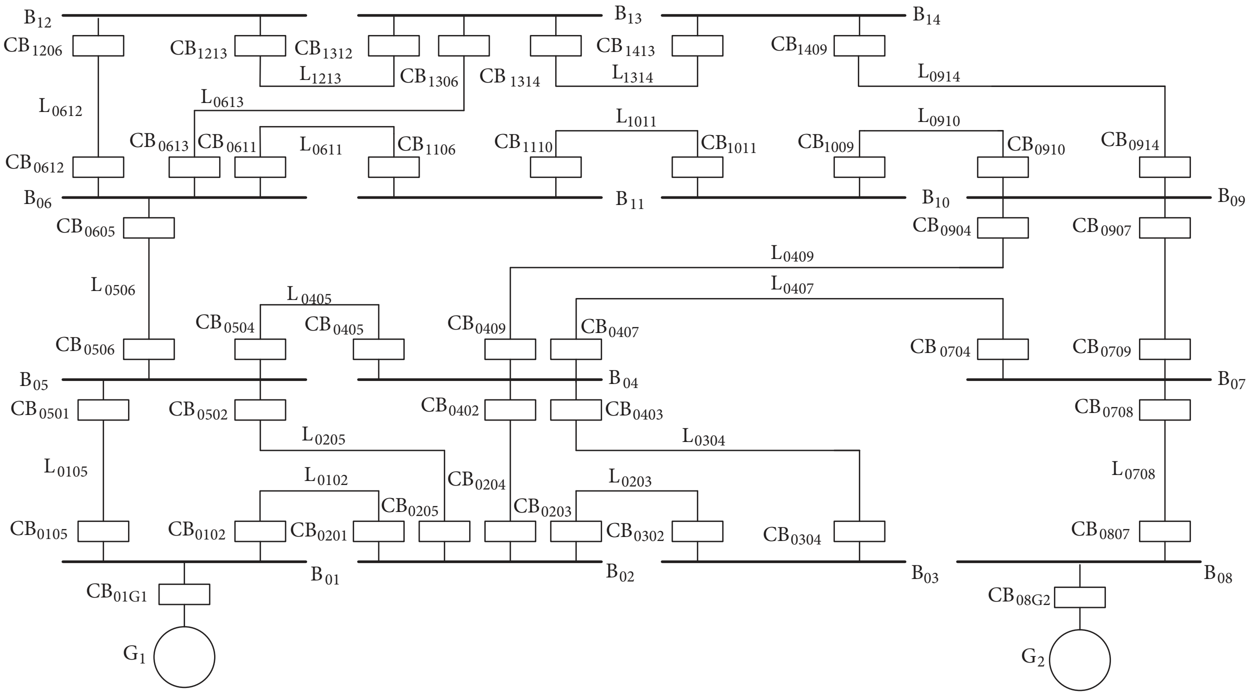


Figure 6: IEEE14 node power system network.

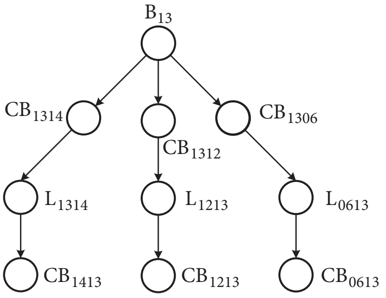


Figure 7: B_{13} Logic diagram.

database in Tables 5 and 6. Furthermore, the whole diagnosis process is carried out in the MATLAB environment. We give an example to illustrate the steps of MCFD method.

Case. Bus B_{13} has a fault

(I) Operated relays: B_{13m}; tripped circuit breakers: $CB_{1306}, CB_{1312}, CB_{1314}$.

(II) Call the network topology analysis algorithm to get the passive zone {13}.

(III) To query the topology database of power system, the corresponding logic diagram of suspected fault components is formed, as shown in Figure 7.

(IV) According to logic diagram of suspected fault components, query protection configuration database is used to form a FRSN P system model, as shown in Figure 8.

(V) The FRSN P system algorithm is invoked to perform inference operations on suspicious fault components.

(VI) The FRSN P system algorithm is invoked to perform inference operations on suspicious fault components.

The reasoning process is as follows:

(1) Parameter initialization: $g = 0$, θ_0 and δ_0 are set according to the pulse values contained in each neuron; i.e., the pulse values can be set according to the initial confidence level in Tables 3 and 4. Therefore, $\theta_0 = (0.9833, 0.8564, 0.2, 0.2, 0.9833, 0.8564, 0.2, 0.2, 0.9833, 0.8564, 0.2, 0.2\ 0, \ldots, 0)^T$, $\delta_0 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0)^T$.

(2) $g = 1$, $\theta_1 = (0, \ldots, 0, 0.8136, 0.19, 0.8136, 0.19, 0.8136, 0.19, 0, 0, 0, 0)^T$, $\delta_1 = (0.8564, 0.2, 0.8564, 0.2, 0.8564, 0.2, 0, 0, 0, 0)^T$.

(3) $g = 2$, $\theta_2 = (0, \ldots, 0, 0.7729, 0.7729, 0.7729)^T$, $\delta_2 = (0, 0, 0, 0, 0, 0, 0.8136, 0.8136, 0.8136, 0)^T$.

Table 5: IEEE14 node power system network topology database.

ID number	Switch operation	Component/switch	Type	Type number	Associated components/switches	Associated protective relays
10101	- -	B01	Bus	101	CB01G1.CB0102.CB0105	10110101.22310201.22310202
10102	- -	B02	Bus	101	CB0201.CB0203.CB0204.CB0205	10110102.21310201.22310203.22310204.22310205
10103	- -	B03	Bus	101	CB0302.CB0304	10110103.21310203.22310206
10104	- -	B04	Bus	101	CB0402.CB0403.CB0405.CB0407.CB0409	10110104.21310204.21310206.22310207.22310208.22310209
10105	- -	B05	Bus	101	CB0501.CB0502.CB0504.CB0506	10110105.21310202.21310205.21310207.22310210
10106	- -	B06	Bus	101	CB0605.CB0611.CB0612.CB0613	10110106.21310210.22310211.22310212.22310213
10107	- -	B07	Bus	101	CB0704.CB0708.CB0709	10110107.21310208.22310214.22310215
10108	- -	B08	Bus	101	CB0807.CB08G2	10110108.21310214
10109	- -	B09	Bus	101	CB0904.CB0907.CB0910.CB0914	10110109.21310209.21310215.22310216.22310217
10110	- -	B10	Bus	101	CB1009.CB1011	10110110.21310216.22310218
10111	- -	B11	Bus	101	CB1106.CB1110	10110111.21310211.21310218
10112	- -	B12	Bus	101	CB1206.CB1213	10110112.21310212.22310219
10113	- -	B13	Bus	101	CB1306.CB1312.CB1314	10110113.21310213.21310219.22310220
10114	- -	B14	Bus	101	CB1409.CB1413	10110114.21310217.21310220
10201	- -	L0102	Line	102	B01.B02	21110201.22110201.21210201.22210201.22310202.22310203.22310204.22310205
10202	- -	L0105	Line	102	B01.B05	21110202.22110202.21210202.22210202.21310205.21310207.22310201.22310210
10203	- -	L0203	Line	102	B02.B03	21110203.22110203.21210203.22210203.21310201.22310204.22310205.22310206
10204	- -	L0204	Line	102	B02.B04	21110204.22110204.21210204.22210204.21310201.21310206.22310203.22310205.22310207.22310208.22310209
10205	- -	L0205	Line	102	B02.B05	21110205.22110205.21210205.22210205.21310201.21310202.21310207.22310203.22310204.22310210
10206	- -	L0304	Line	102	B03.B04	21110206.22110206.21210206.22210206.21310203.21310204.22310207.22310208.22310209
10207	- -	L0405	Line	102	B04.B05	21110207.22110207.21210207.22210207.21310202.21310204.21310205.21310206.22310208.22310209.22310210
10208	- -	L0407	Line	102	B04.B07	21110208.22110208.21210208.22210208.21310204.21310206.22310207.22310209.22310214.22310215
10209	- -	L0409	Line	102	B04.B09	21110209.22110209.21210209.22210209.21310204.21310206.21310215.22310207.22310208.22310216.22310217
10210	- -	L0506	Line	102	B05.B06	21110210.22110210.21210210.22210210.21310202.21310205.21310207.22310211.22310212.22310213
10211	- -	L0611	Line	102	B06.B11	21110211.22110211.21210211.22210211.21310210.22310212.22310213
10212	- -	L0612	Line	102	B06.B12	21110212.22110212.21210212.22210212.22310211.22310213.22310219
10213	- -	L0613	Line	102	B06.B13	21110213.22110213.21210213.22210213.21310210.21310219.22310211.22310212.22310220
10214	- -	L0708	Line	102	B07.B08	21110214.22110214.21210214.22210214.21310208.22310215
10215	- -	L0709	Line	102	B07.B09	21110215.22110215.21210215.22210215.21310208.21310209.22310214.22310216.22310217
10216	- -	L0910	Line	102	B09.B10	21110216.22110216.21210216.22210216.21310209.21310215.22310217.22310218
10217	- -	L0914	Line	102	B09.B14	21110217.22110217.21210217.22210217.21310209.21310215.21310220.22310216
10218	- -	L1011	Line	102	B10.B11	21110218.22110218.21210218.22210218.21310211.21310216
10219	- -	L1213	Line	102	B12.B13	21110219.22110219.21210219.22210219.21310212.21310213.22310220
10220	- -	L1314	Line	102	B13.B14	21110220.22110220.21210220.22210220.21310213.21310217.21310219

Table 6: IEEE14 node power system network protection configuration database.

ID number	Protective relay	Relays operation	Operated switches	Protected equipment
10110101	Main_protective relay	0	10401.10403.10501	B01
10110102	Main_protective relay	0	10402.10405.10407.10409	B02
10110103	Main_protective relay	0	10406.10411.	B03
10110104	Main_protective relay	0	10408.10412.10413.10415.10417	B04
10110105	Main_protective relay	0	10404.10410.10414.10419	B05
10110106	Main_protective relay	0	10420.10421.10423.10425	B06
10110107	Main_protective relay	0	10416.10427.10429	B07
10110108	Main_protective relay	0	10428.10442	B08
10110109	Main_protective relay	0	10418.10430.10431.10433	B09
10110110	Main_protective relay	0	10432.10435	B10
10110111	Main_protective relay	0	10422.10436	B11
10110112	Main_protective relay	0	10424.10437	B12
10110113	Main_protective relay	0	10426.10438.10439	B13
10110114	Main_protective relay	0	10434.10440	B14
21110201	Main_protective relay	0	10401	L0102
21110202	Main_protective relay	0	10403	L0105
21110203	Main_protective relay	0	10405	L0203
21110204	Main_protective relay	0	10407	L0204
21110205	Main_protective relay	0	10409	L0205
21110206	Main_protective relay	0	10411	L0304
21110207	Main_protective relay	0	10413	L0405
21110208	Main_protective relay	0	10415	L0407
21110209	Main_protective relay	0	10417	L0409
21110210	Main_protective relay	0	10419	L0506
21110211	Main_protective relay	0	10421	L0611
21110212	Main_protective relay	0	10423	L0612

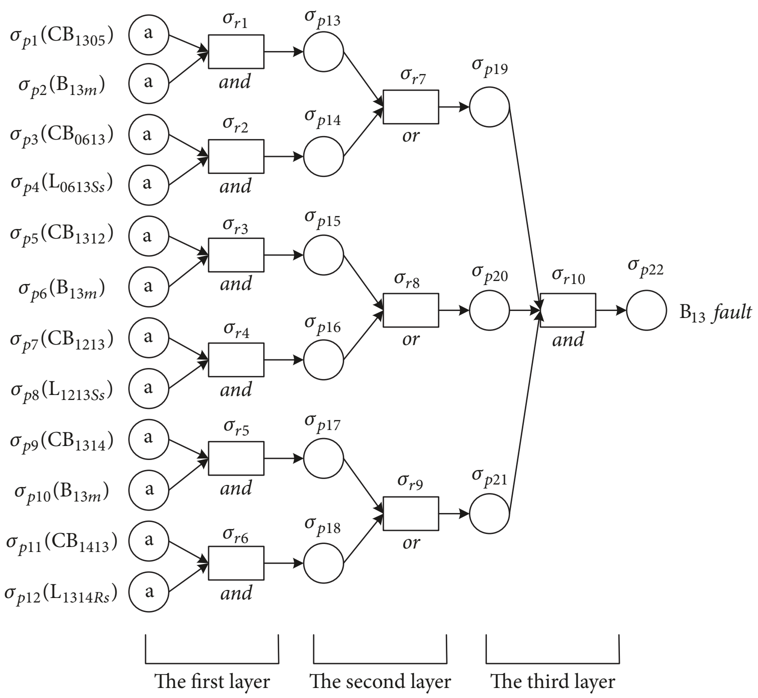
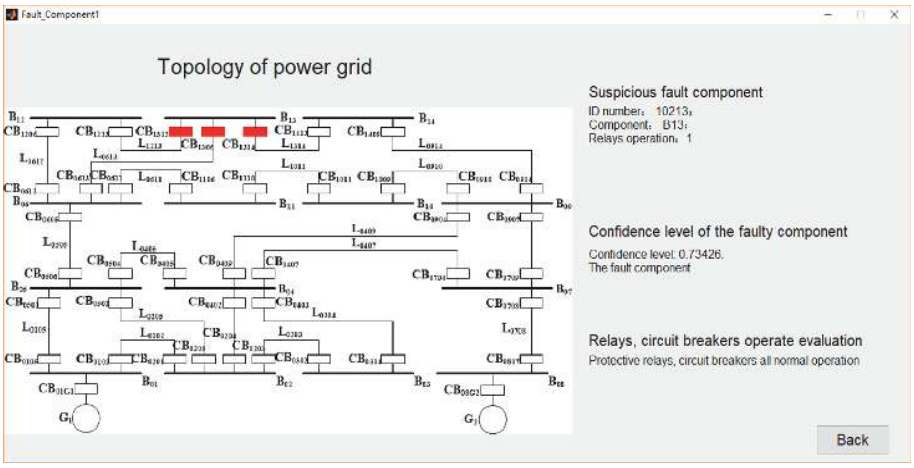

$\sigma_{p1}(\mathrm{CB}_{1305})$
$\sigma_{p2}(\mathrm{B}_{13m})$
$\sigma_{p3}(\mathrm{CB}_{0613})$
$\sigma_{p4}(\mathrm{L}_{0613Ss})$
$\sigma_{p5}(\mathrm{CB}_{1312})$
$\sigma_{p6}(\mathrm{B}_{13m})$
$\sigma_{p7}(\mathrm{CB}_{1213})$
$\sigma_{p8}(\mathrm{L}_{1213Ss})$
$\sigma_{p9}(\mathrm{CB}_{1314})$
$\sigma_{p10}(\mathrm{B}_{13m})$
$\sigma_{p11}(\mathrm{CB}_{1413})$
$\sigma_{p12}(\mathrm{L}_{1314Rs})$
a
σ_{r1} and
σ_{r2} and
σ_{r3} and
σ_{r4} and
σ_{r5} and
σ_{r6} and
σ_{p13}
σ_{p14}
σ_{p15}
σ_{p16}
σ_{p17}
σ_{p18}
σ_{r7} or
σ_{r8} or
σ_{r9} or
σ_{p19}
σ_{p20}
σ_{p21}
σ_{r10} and
σ_{p22}
B_{13} *fault*
The first layer
The second layer
The third layer

Figure 8: B_{13} FRSN P system model.



Fault_Component1

Topology of power grid

Suspicious fault component

ID number: 10213;
Component: B13;
Relays operation: 1

Confidence level of the faulty component

Confidence level: 0.73426.
The fault component

Relays, circuit breakers operate evaluation

Protective relays, circuit breakers all normal operation

Figure 9: IEEE14 node power system fault diagnosis program simulation results (B_i represents the bus, L_{ij} represents the line between the bus B_i and B_j, CB_{ij} represents the circuit breaker connecting the B_i bus and the L_j line, and G_i represents the generator).

(4) $g = 3, \theta_3 = (0, \ldots, 0, 0.7343)^T, \delta_3 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0.7729)^T$.

(5) $g = 4, \delta_4 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0)^T = 0$. The termination condition is satisfied and output is obtained.

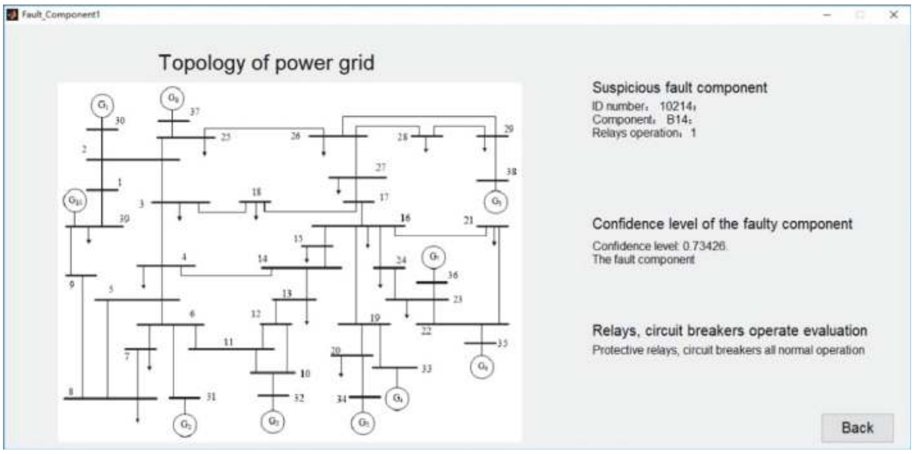
The fault component confidence level of bus B_{13} is 0.7343, according to the decision rule. Bus B_{13} is the fault component. The diagnostic results in the form of a graphical user interface have been displayed in Figure 9.

(VII) Protective relays and circuit breakers perform all operations normally.

In addition, this method is also applicable to the large-scale power grid. In this paper, the IEEE39 node power system (show as Figure 10), and the IEEE118 node power system (show as Figure 11) are tested in the same way, whether there is a single fault or multiple faults, even if the protective relay and circuit breaker information are lost. The faulty component can still be quickly diagnosed.

4. Complexity Analysis

It is assumed that there are $\boldsymbol{n}$ busbars, lines, transformer components, and switches in a given power grid. If each of these components and switches is treated as a node, the power

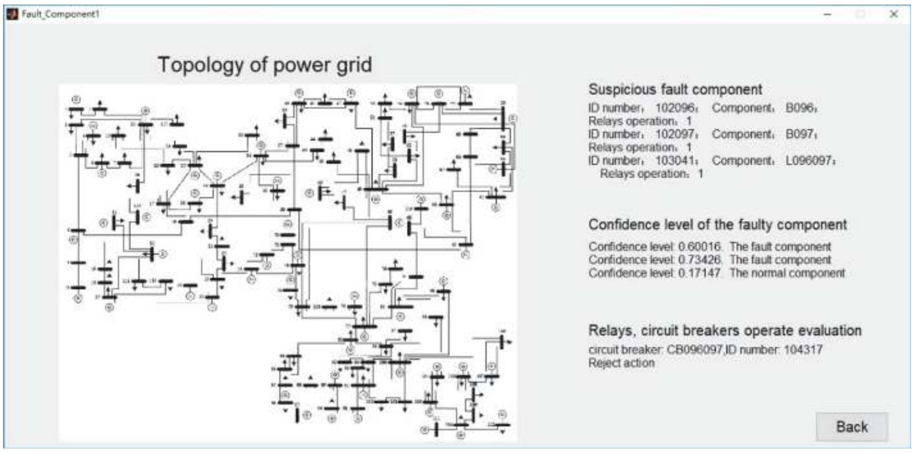


Suspicious fault component
ID number： 10214，
Component： B14：
Relays operation： 1

Confidence level of the faulty component
Confidence level: 0.73426.
The fault component

Relays, circuit breakers operate evaluation
Protective relays, circuit breakers all normal operation

Figure 10: IEEE39 node power system fault diagnosis program simulation results (B_i represents the bus, L_{ij} represents the line between the bus B_i and B_j, CB_{ij} represents the circuit breaker connecting the B_i bus and the L_j line, and G_i represents the generator).



Suspicious fault component
ID number， 102096， Component， B096，
Relays operation， 1
ID number， 102097， Component， B097，
Relays operation， 1
ID number， 103041， Component， L096097：
Relays operation： 1

Confidence level of the faulty component
Confidence level: 0.60016. The fault component
Confidence level: 0.73426. The fault component
Confidence level: 0.17147. The normal component

Relays, circuit breakers operate evaluation
circuit breaker: CB096097,ID number: 104317
Reject action

Figure 11: IEEE118 node power system fault diagnosis program simulation results (B_i represents the bus, L_{ij} represents the line between the bus B_i and B_j, CB_{ij} represents the circuit breaker connecting the B_i bus and the L_j line, and G_i represents the generator).

grid can be viewed as a topology with $\boldsymbol{n}$ nodes. For a given power grid, the number of outgoing lines of each component is determined by a constant set as $\boldsymbol{k}$. In this paper, we mainly analyze the complexity of this automatic implementation method with busbar components as an example. For busbar components with $\boldsymbol{k}$ outgoing lines, the protection settings are 1 bus main protection, $\boldsymbol{k}$ backup protection on line components with different outgoing directions, a total of $\boldsymbol{k}+1$ protections, and the number of circuit breakers involved which is $2*\boldsymbol{k}$. Analysis of each algorithm module is described as follows.

In the suspicious fault component analysis algorithm module (*as shown in Pseudocode 1*), according to the protection configuration of bus components, the relation between the number of nodes and the number of outgoing lines in the formed logical graph is $3\boldsymbol{k}+1$. At the same time, each node in the logic diagram needs to determine whether it is the component associated with the protection of the suspect fault component. Therefore, $(3\boldsymbol{k}+1)*(\boldsymbol{k}+1)$ steps need to be performed for the formation of logical diagram of each suspect fault component, and a total of $(3\boldsymbol{k}+1)*(\boldsymbol{k}+1)*\boldsymbol{n}$ steps need to be performed for an $\boldsymbol{n}$-node system.

In the modeling suspicious fault components with FRSN P system algorithm module (*as shown in Pseudocode 2*), there are $\boldsymbol{k}$ branches in the logic diagram for the suspect fault component. At the same time, $\boldsymbol{k}+1$ protections are configured. In these protections, either the main protection action or the backup protection action will cause the related $\boldsymbol{k}$ circuit breakers to trip. Therefore, obtaining the initial value θ_0 of FRSN P system diagnostic model from logic diagram need performs a total of $(\boldsymbol{k}+1)*\boldsymbol{k}*\boldsymbol{k}$ steps.

In the fuzzy inference module (*as shown in Pseudocode 3*). In the FRSN P system diagnosis model, there are $7\boldsymbol{k}+1$ proposition neurons and $3\boldsymbol{k}+1$ rule neurons. In order to obtain the pulse value of the output proposition neuron, the algorithm performs a total of $(7\boldsymbol{k}+1)*(3\boldsymbol{k}+1)$ steps. Therefore, when the proposed method diagnoses the faulty component, the algorithm performs a total of

```
Input: Suspicious fault component node set $N$, Component node-switch node adjacent matrix $G$
Output: Information about each component node and breaker node in the logic diagram
Set stop condition: All associated protection of suspicious fault components are accessed, $protective\_mark = (1, 1, \ldots, 1)_p$;
Set Initialize state: $protective\_mark = (0, 0, \ldots, 0)_p$, $n = 0$;
Add the suspicious fault component node to set $Set_0$;
while (all $protective\_mark \neq 1$) do
    for each node $i$ in set $Set_n$ do
        for each node $j$ in node-switch adjacent matrix $G$ do
            if node adjacent matrix $G(i, j) == 1$ then
                The searched node $j$ are added to set $Set_{n+1}$;
                $G(i, j) = 0$;
                Add the node $j$ to logic diagram;
                for each associated protection of suspicious fault component $k$ do
                    if $j$ is the node of protection $k$-switch association set then
                        $protective\_mark_k = 1$;
                    end if
                end for
            end if
        end for
    end for
    $n = n + 1$;
end while
```

Pseudocode 1: The suspicious fault component analysis algorithm.

```
Input: Suspicious fault component logic diagram adjacent matrix $S_G$, Information about each component node and breaker node in the logic diagram
Output: The fuzzy truth values of the input proposition neurons $\theta_0$
Set stop condition: All outgoing line directions of the suspicious fault component logic diagram are access completed;
Set Initialize state: Set initialize access outgoing line direction number $k = 1$, set all outgoing line directions number of suspicious fault component $K$;
while ($k \leq K$) do
    for each associated protection of suspicious fault components $i$ do
        if associated protection $i$ is the main protection then
            query the protection database, obtain breaker ID number that should be tripped when protection $i$ is active;
            for the must trip breaker $j$ when main protection $i$ action do
                if $j$ is the breaker in the outgoing line direction $k$ then
                    According to the operation information of breaker $j$ and protection $i$ set the fuzzy truth value vector $\theta_0$;
                end if
            end for
        else associated protection $i$ is the backup protection then
            query the protection database, obtain breaker ID number that should be tripped when protection $i$ is active;
            for the must trip breaker $j$ when backup protection $i$ action do
                if $j$ is the breaker in the outgoing line direction $k$ then
                    According to the operation information of breaker $j$ and protection $i$ set the fuzzy truth value vector $\theta_0$;
                end if
            end for
        end if
        $k = k + 1$;
    end for
end while
```

Pseudocode 2: The modeling suspicious fault components with FRSN P system algorithm.

$(3\boldsymbol{k}^2) * \boldsymbol{n} + \boldsymbol{k}^2 + 22\boldsymbol{k}^2 + 10\boldsymbol{k} + 1$ steps, so the time complexity of the proposed algorithm is $\boldsymbol{O}(\boldsymbol{n})$, which is very competitive to other power system diagnosis techniques like Petri nets [43–46, 48].

5. Conclusions

In this paper, the whole diagnosis process of power system fault diagnosis based on FRSN P systems is realized by an

```
Input: The fuzzy truth values of the input proposition neurons $\theta_0$
Output: The fuzzy truth value of the output proposition neurons $\theta_g$
Set stop condition: $0_1 = (0, 0, \ldots, 0)_s^T, 0_1 = (0, 0, \ldots, 0)_t^T$;
Set reasoning step: $g = 1$;
while ($\theta_g \neq 0_1$ or $\delta_g \neq 0_2$) do
    for each input neurons (g = 1) or proposition neurons ($g > 1$) do
        if the condition of fire $E = a^n$ is satisfied then
            Compute the fuzzy value vector $\delta_g$, $\delta_g = D_1^T \otimes \theta_{g-1} + D_2^T \odot \theta_{g-1} + D_3^T \circledast \theta_{g-1}$;
            if this proposition neurons have a postsynaptic neuron then
                Transmits a spike to the next rule neuron;
            else
                this proposition neurons only accumulates spike values;
            end if
        end if
    end for
    for each rule neurons do
        if the condition of fire $E = a^n$ is satisfied then
            Compute the fuzzy truth value vector $\theta_g$, $\theta_g = E^T \circledast (C \otimes \delta_g)$;
            Transmits a spike to the next proposition neuron;
        end if
    end for
    $g = g + 1$;
end while
```

Pseudocode 3: The fuzzy inference algorithm.

automatic implementation method. The whole diagnostic algorithm includes data structure module, power grid topology analysis algorithm module, suspicious fault component logic analysis algorithm module, and FRSN P system inference algorithm module. IEEE14, IEEE39, and IEEE118 node systems are discussed to verify the diagnosis algorithm program. The diagnosis of the results is consistent with manual calculation results, which verifies the feasibility and reliability of automatic diagnosis algorithm procedures. The FRSN P system is used for fault diagnosis of the entire program diagnostic algorithm and also the speed and complexity of the various modules have been studied. Moreover, this method can be used to improve the automation of fault diagnosis and to explore the superiority of the fault diagnosis method based on FRSN P system in the large-scale grid fault diagnosis with more nodes in comparison with other methods.

Data Availability

The Data Availability Statement for our manuscript 2635714 titled "Automatic Implementation of Fuzzy Reasoning Spiking Neural P Systems for Diagnosing Faults in Complex Power Systems" submitted to Complexity are as follows: (1) The data for the 14-bus power system used to support the findings of this study are available in Ref. [47] "X. Luo., M. Kezunovic. Implementing fuzzy reasoning Petri nets for fault section estimation, IEEE Transactions on Power Delivery, vol. 23, no. 2, pp. 676-685, 2008". (2) The data for the 39-bus power system used to support the findings of this study are available in Ref. [50] "K. Huang, T. Wang, Y. He, G. Zhang, M.J. Pérez-Jiménez. Temporal fuzzy reasoning spiking neural P systems with real numbers for power system fault diagnosis. Journal of Computational and Theoretical Nanoscience, vol. 13, no. 6, pp. 3804-3814, 2016". (3) The data for the 118-bus power system used to support the findings of this study are available in the reference "T. Bi, F. Wen, Y. Ni, F.F. Wu. Distributed fault section estimation system using radial basis function neural network and its companion fuzzy system, International Journal of Electrical Power & Energy Systems, vol. 25, no. 5, pp. 377-386, 2003". This reference will be inserted in the updated version.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61702428 and 61672437), the New Generation Artificial Intelligence Science and Technology Major Project of Sichuan Province (2018GZDZX0043), and the Sichuan Science and Technology Program (2018GZ0185, 2018GZ0086, and 2018GZ0095).

References

[1] G. Păun, "Computing with membranes," *Journal of Computer and System Sciences*, vol. 61, no. 1, pp. 108–143, 2000.

[2] G. Paun, *Membrane Computing: An Introduction*, Springer, Berlin, Germany, 2002.

[3] G. Zhang, M. Gheorghe, and M. J. Perez-Jimenez, *Real-life Applications with Membrane Computing*, Springer, 2017.

[4] L. Pan, G. Păun, and G. Zhang, "Foreword: starting JMC," *Journal of Membrane Computing*, vol. 1, no. 1, pp. 1-2, 2019.

[5] D. Orellana-Martín, L. Valencia-Cabrera, A. Riscos-Núñez, and M. J. Pérez-Jiménez, "P systems with proteins: a new frontier when membrane division disappears," *Journal of Membrane Computing*, vol. 1, no. 1, pp. 29–39, 2019.

[6] E. Csuhaj, M. Margenstern, G. Vaszil, and S. Verlan, "On small universal antiport P systems," *Theoretical Computer Science*, vol. 372, no. 2-3, pp. 152–164, 2007.

[7] G. H. Paun, M. J. Perez-Jimenez, and A. Riscos, "Tissue P systems with cell division," *International Journal of Computers Communications & Control*, vol. 3, no. 3, pp. 295–303, 2008.

[8] L. Pan, J. Wang, and H. J. Hoogeboom, "Spiking neural P systems with astrocytes," *Neural Computation*, vol. 24, no. 3, pp. 805–825, 2012.

[9] V. Manca, "Metabolic computing," *Journal of Membrane Computing*, vol. 1, no. 1, 2019.

[10] G. Román, "Inference of bounded L systems with polymorphic P systems," *Journal of Membrane Computing*, vol. 1, no. 1, pp. 52–57, 2019.

[11] F. G. C. Cabarle, H. N. Adorna, and M. J. Perez-Jimenez, "Sequential spiking neural P systems with structural plasticity based on max/min spike number," *Neural Computing and Applications*, vol. 27, no. 5, pp. 1337–1347, 2016.

[12] C. Martín-Vide, G. Păun, J. Pazos, and A. Rodríguez-Patón, "Tissue P systems," *Theoretical Computer Science*, vol. 296, no. 2, pp. 295–326, 2003.

[13] M. Ionescu, G. Păun, and T. Yokomori, "Spiking neural P systems," *Fundamenta Informaticae*, vol. 71, no. 2-3, pp. 279–308, 2006.

[14] R. Mayne, N. Phillips, and A. Adamatzky, "Towards experimental P-systems using multivesicular liposomes," *Journal of Membrane Computing*, vol. 1, no. 1, pp. 20–28, 2019.

[15] A. Leporati, L. Manzoni, G. Mauri, A. E. Porreca, and C. Zandron, "Characterizing PSPACE with shallow non-confluent P systems," *Journal of Membrane Computing*, vol. 1, no. 2, 2019.

[16] E. Sánchez-Karhunen and L. Valencia-Cabrera, "Modelling complex market interactions using PDP systems," *Journal of Membrane Computing*, vol. 1, no. 1, pp. 40–51, 2019.

[17] D. Díaz-Pernil, M. A. Gutiérrez-Naranjo, and H. Peng, "Membrane computing and image processing: a short survey," *Journal of Membrane Computing*, vol. 1, no. 1, pp. 58–73, 2019.

[18] D. Orellana-Martín, L. Valencia-Cabrera, A. Riscos-Núñez, and M. J. Pérez-Jiménez, "Minimal cooperation as a way to achieve the efficiency in cell-like membrane systems," *Journal of Membrane Computing*, vol. 1, no. 2, pp. 10–1007, 2019.

[19] J. He, J. Xiao, X. Liu, T. Wu, and T. Song, "A novel membrane-inspired algorithm for optimizing solid waste transportation," *Optik - International Journal for Light and Electron Optics*, vol. 126, no. 23, pp. 3883–3888, 2015.

[20] G. Zhang, H. Rong, F. Neri, and M. J. Pérez-Jiménez, "An optimization spiking neural P system for approximately solving combinatorial optimization problems," *International Journal of Neural Systems*, vol. 24, no. 5, Article ID 1440006, 2014.

[21] C. Buiu, C. Vasile, and O. Arsene, "Development of membrane controllers for mobile robots," *Information Sciences*, vol. 187, no. 1, pp. 33–51, 2012.

[22] X. Wang, G. Zhang, F. Neri et al., "Design and implementation of membrane controllers for trajectory tracking of nonholonomic wheeled mobile robots," *Integrated Computer-Aided Engineering*, vol. 23, no. 1, pp. 15–30, 2016.

[23] G. Zhang, J. Cheng, M. Gheorghe, and Q. Meng, "A hybrid approach based on differential evolution and tissue membrane systems for solving constrained manufacturing parameter optimization problems," *Applied Soft Computing*, vol. 13, no. 3, pp. 1528–1542, 2013.

[24] H. Peng, J. Wang, M. J. Pérez-Jiménez, H. Wang, J. Shao, and T. Wang, "Fuzzy reasoning spiking neural P system for fault diagnosis," *Information Sciences*, vol. 235, pp. 106–116, 2013.

[25] B. Song, C. Zhang, and L. Pan, "Tissue-like P systems with evolutional symport/antiport rules," *Information Sciences*, vol. 378, pp. 177–193, 2017.

[26] B. Song, L. Pan, H. Adorna et al., "A quick survey of tissue-like P systems," *Romanian Journal of Information Science and Technology*, vol. 21, no. 3, pp. 310–321, 2018.

[27] G. H. Paun, M. J. Perez-Jimenez, and G. Rozenberg, "Spike train in spiking neural P systems," *International Journal of Foundations of Computer Science*, vol. 17, no. 4, pp. 975–1002, 2006.

[28] K. Jiang, W. Chen, Y. Zhang, and L. Pan, "On string languages generated by sequential spiking neural P systems based on the number of spikes," *Natural Computing*, vol. 15, no. 1, pp. 87–96, 2016.

[29] L. Pan and G. Păun, "Spiking neural P systems: an improved normal form," *Theoretical Computer Science*, vol. 411, no. 6, pp. 906–918, 2010.

[30] L. Pan, G. Păun, G. Zhang, and F. Neri, "Spiking neural P systems with communication on request," *International Journal of Neural Systems*, vol. 27, no. 08, p. 1750042, 2017.

[31] J. Wang, P. Shi, H. Peng, M. J. Perez-Jimenez, and T. Wang, "Weighted fuzzy spiking neural P systems," *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 2, pp. 209–220, 2013.

[32] J. Wang and H. Peng, "Adaptive fuzzy spiking neural P systems for fuzzy inference and learning," *International Journal of Computer Mathematics*, vol. 90, no. 4, pp. 857–868, 2013.

[33] T. Wu, Z. Zhang, and L. Pan, "On languages generated by cell-like spiking neural P systems," *IEEE Transactions on NanoBioscience*, vol. 15, no. 5, pp. 455–467, 2016.

[34] A. Leporati, G. Mauri, C. Zandron et al., "Uniform solutions to SAT and Subset Sum by spiking neural P systems," *Natural Computing*, vol. 8, no. 4, pp. 681–702, 2009.

[35] T. Wang, G. Zhang, J. Zhao, Z. He, J. Wang, and M. J. Perez-Jimenez, "Fault diagnosis of electric power systems based on fuzzy reasoning spiking neural P systems," *IEEE Transactions on Power Systems*, vol. 30, no. 3, pp. 1182–1194, 2015.

[36] G. Xiong, D. Shi, L. Zhu et al., "A new approach to fault diagnosis of power systems using fuzzy reasoning spiking neural P systems," *Mathematical Problems in Engineering*, vol. 2013, no. 1, pp. 211–244, 2013.

[37] M. Tu, J. Wang, H. Peng, and P. Shi, "Application of adaptive fuzzy spiking neural P systems in fault diagnosis of power systems," *Chinese Journal of Electronics*, vol. 23, no. 1, pp. 87–92, 2014.

[38] J. Wang, H. Peng, M. Tu, P.-J. J. Mario, and P. Shi, "A fault diagnosis method of power systems based on an improved adaptive fuzzy spiking neural P systems and PSO algorithms," *Journal of Electronics*, vol. 25, no. 2, pp. 320–327, 2016.

[39] H. Rong, M. Zhu, Z. Feng, G. Zhang, and K. Huang, "A novel approach to fault classification of power transmission lines using singular value decomposition and fuzzy reasoning spiking neural P systems," *Romanian Journal of Information Science and Technology*, vol. 20, no. 1, pp. 18–31, 2017.

[40] T. Wang, G. Zhang, M. J. Pérez-Jiménez, and J. Cheng, "Weighted fuzzy reasoning spiking neural P systems: Application to fault diagnosis in traction power supply systems of

high-speed railways," *Journal of Computational and Theoretical Nanoscience*, vol. 12, no. 7, pp. 1103–1114, 2015.

[41] Y. He, T. Wang, K. Huang, G. X. Zhang, and M. J. Perez-Jimenez, "Fault diagnosis of metro traction power systems using a modified fuzzy reasoning spiking neural P system," *Romanian Journal of Information Science and Technology*, vol. 18, no. 3, pp. 256–272, 2015.

[42] T. Wang, G. Zhang, and M. J. Perez-Jimenez, "Fuzzy membrane computing: theory and applications," *International Journal of Computers Communications & Control*, vol. 10, no. 6, pp. 144–175, 2015.

[43] P. Fang, Y. Li, and X. Yang, "A fault diagnosis method of transmission network combining Petri net with expert system," *Proceedings of the Chinese Society of Universities for Electric Power System and Automation*, vol. 17, no. 2, pp. 26–30, 2005.

[44] J. Sun, S. H. Qin, and Y. Song, "A fault diagnosis method of power system based on Petri net and probabilistic information," *Automation of Electric Power Systems*, vol. 27, no. 13, pp. 10–14, 2003.

[45] F. Wen, Z. H. Han, and L. Tian, "The analytical model and method of fault diagnosis of power system based on genetic algorithm," *Proceedings of the Chinese Society of Universities for Electric Power System and Automation*, vol. 10, no. 3, pp. 1–7, 1998.

[46] F. Wen, Z. H. Han, and L. Tian, "The analysis model and method of power system fault diagnosis based on genetic algorithm are implemented in the second part," *Proceedings of the Chinese Society of Universities for Electric Power System and Automation*, vol. 10, no. 3, pp. 8–14, 1998.

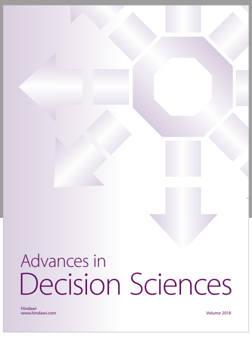
[47] X. Luo and M. Kezunovic, "Implementing fuzzy reasoning Petri-nets for fault section estimation," *IEEE Transactions on Power Delivery*, vol. 23, no. 2, pp. 676–685, 2008.

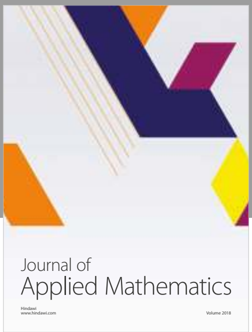
[48] J. Sun, S.-Y. Qin, and Y.-H. Song, "Fault diagnosis of electric power systems based on fuzzy Petri nets," *IEEE Transactions on Power Systems*, vol. 19, no. 4, pp. 2053–2059, 2004.

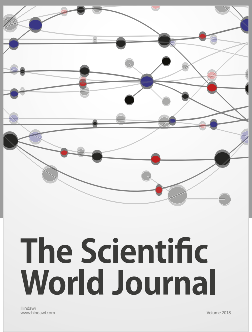
[49] L. Zhou, Y. Wang, and M. Zhao, "In 2004, the state grid relay protection and safety automatic device operation," *Power System Technology*, vol. 29, no. 16, pp. 42–48, 2005.

[50] K. Huang, T. Wang, Y. He, G. Zhang, and M. J. Pérez-Jiménez, "Temporal fuzzy reasoning spiking neural P systems with real numbers for power system fault diagnosis," *Journal of Computational and Theoretical Nanoscience*, vol. 13, no. 6, pp. 3804–3814, 2016.

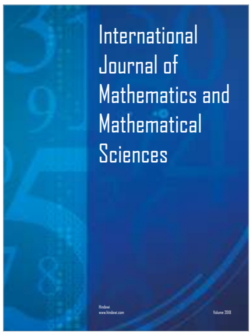
Advances in
Operations Research


Advances in
Decision Sciences

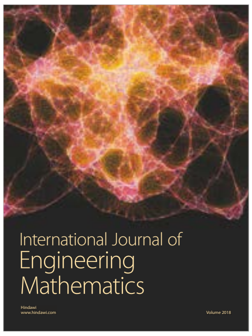

Journal of
Applied Mathematics


The Scientific
World Journal

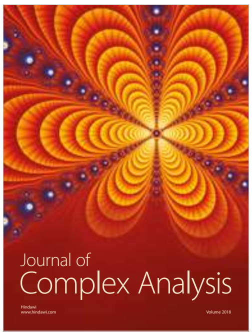
Journal of
Probability and Statistics

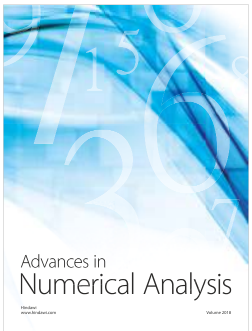

International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Optimization

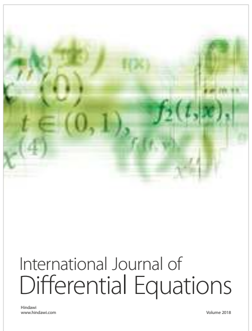

International Journal of
Engineering
Mathematics

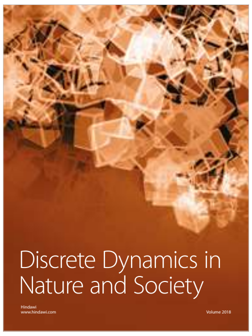
International Journal of
Analysis

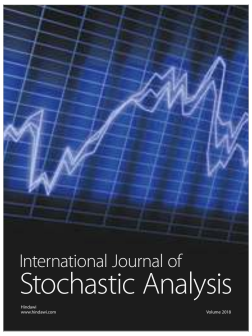

Journal of
Complex Analysis


Advances in
Numerical Analysis

Mathematical Problems
in Engineering

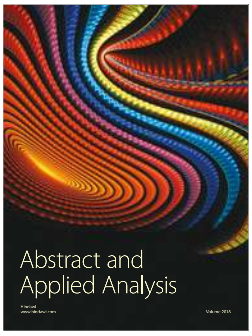

International Journal of
Differential Equations

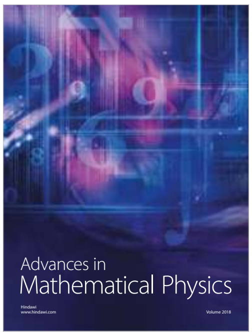

Discrete Dynamics in
Nature and Society


International Journal of
Stochastic Analysis

Journal of
Mathematics

Journal of
Function Spaces


Abstract and
Applied Analysis


Advances in
Mathematical Physics