entropy

MDPI

Article

Fuzzy Reasoning Numerical Spiking Neural P Systems for Induction Motor Fault Diagnosis

Xiu Yin [1], Xiyu Liu [1,*], Minghe Sun [2], Jianping Dong [3] and Gexiang Zhang [4]

[1] Academy of Management Science, Business School, Shandong Normal University, Jinan 250014, China
[2] College of Business, The University of Texas at San Antonio, San Antonio, TX 78249, USA
[3] Research Center for Artificial Intelligence, Chengdu University of Technology, Chengdu 610059, China
[4] School of Automation, Chengdu University of Information Technology, Chengdu 610225, China
[*] Correspondence: xyliu@sdnu.edu.cn

Abstract: The fuzzy reasoning numerical spiking neural P systems (FRNSN P systems) are proposed by introducing the interval-valued triangular fuzzy numbers into the numerical spiking neural P systems (NSN P systems). The NSN P systems were applied to the SAT problem and the FRNSN P systems were applied to induction motor fault diagnosis. The FRNSN P system can easily model fuzzy production rules for motor faults and perform fuzzy reasoning. To perform the inference process, a FRNSN P reasoning algorithm was designed. During inference, the interval-valued triangular fuzzy numbers were used to characterize the incomplete and uncertain motor fault information. The relative preference relationship was used to estimate the severity of various faults, so as to warn and repair the motors in time when minor faults occur. The results of the case studies showed that the FRNSN P reasoning algorithm can successfully diagnose single and multiple induction motor faults and has certain advantages over other existing methods.

Keywords: fuzzy reasoning numerical spiking neural P systems; interval-valued triangular fuzzy numbers; fault diagnosis

check for updates

Citation: Yin, X.; Liu, X.; Sun, M.; Dong, J.; Zhang, G. Fuzzy Reasoning Numerical Spiking Neural P Systems for Induction Motor Fault Diagnosis. *Entropy* **2022**, *24*, 1385. https://doi.org/10.3390/e24101385

Academic Editor: Salim Lahmiri

Received: 4 September 2022
Accepted: 26 September 2022
Published: 28 September 2022

1. Introduction

Induction motors are widely used to drive various mechanical and industrial equipment. The major components of an induction motor are usually stators, rotors, air gaps and bearings [1,2]. Due to their heavy workload and harsh working environment, induction motors are prone to various hidden troubles during operations. The occurrence of faults usually causes huge economic losses, so it is necessary to detect faults early, prevent the occurrence and development of faults and prevent the occurrence of destructive and catastrophic accidents [3–5]. The fault diagnosis of induction motors generally consists of two processes including state detection and diagnosis. Specifically, by monitoring and analyzing its relevant operating parameters, the current operating state of an induction motor is evaluated to determine whether a fault exists. If it is in a fault state, the location, severity and development trend of the fault need to be clarified [6,7].

In recent years, motor fault diagnosis methods based on artificial neural networks [8–11] have become a research hotspot. Mejia-Barron et al. [12] proposed a multi-layer neural network-based model to reproduce the current characteristics associated with inter-turn short circuit fault conditions, providing a new tool for testing and monitoring the induction motor working conditions. Deng et al. [11] proposed a new method for bearing fault diagnosis based on empirical wavelet transform, fuzzy entropy and support vector machines. Kumar and Hati [6] proposed a new detection technique for bearing faults and broken rotor bars of squirrel-cage induction motors based on an extended convolutional neural network model. Although neural networks can be used to find solutions according to the faults that need to be resolved, they also have obvious disadvantages, such as the need to learn from a large number of samples, slow convergence and serious local optimal solutions [13]. In addition, the above-mentioned methods do not have the ability to detect faults in complex conditions and cannot conduct a comprehensive diagnosis of the entire machine [14,15].

As a new high-performance distributed parallel computing model, fuzzy reasoning spiking neural P systems (FRSN P systems) [16] have been widely used in power system fault diagnosis and have achieved good results. Since spiking neural P systems (SN P systems) do not have the ability to deal with fuzzy and uncertain data in fault diagnosis problems, the FRSN P systems integrate different fuzzy logics into SN P systems. Various fuzzy reasoning algorithms for fault diagnosis using FRSN P systems have been developed [15–21]. An SN P system consists of a network of neurons connected together with synapses and can be regarded as the third generation of neural network models. SN P systems transmit information with pulses (spikes) among neurons through synapses [22]. A SN P system is also a directed graph, in which nodes represent neurons, and the connections between nodes represent synapses. Spikes are transmitted from presynaptic neurons to postsynaptic neurons along the synapses [23]. Variants of SN P systems have been developed and have been applied not only to power system fault diagnosis but also to Boolean circuits modeling [24], combinatorial optimization [25], image processing [26,27] and fingerprint recognition [28].

Fault diagnosis using FRSN P systems stems from the similarity between the transmission of impulses between neurons through synapses and the propagation of faults in power systems [4,29]. Although FRSN P systems have been used for fault diagnosis of power systems, spikes are only used as a “tool” in FRSN P systems to perform operations between the values that spikes can represent, i.e., real numbers in the interval [0, 1]. In order to take full advantage of numerical operations, this work adopts the numerical spiking neural P system (NSN P system) proposed by Wu et al. [30] and uses it for fault diagnosis of induction motors. The NSN P systems establish the connection between SN P systems and NP systems by replacing the spikes and the evolution rules in the SN P systems with numerical variables and programs in the NP systems, respectively, but still retaining the directed graph structure of the SN P systems.

The NSN P system is used first to solve the Boolean satisfiability (SAT) problem, a NP-complete problem, to demonstrate its computational capability, before the FRNSN P system is used for motor fault diagnosis. The SAT problem and the motor fault diagnosis problem have certain similarities since their cores are propositional formulas. Therefore, the NSN P system is capable of solving the motor fault diagnosis problem if it can successfully solve the SAT problem.

In order to better characterize the uncertainty in motor fault diagnosis, the fuzzy reasoning numerical spiking neural P systems (FRNSN P systems) are developed by introducing the interval-valued triangular fuzzy numbers (IVTFNs) [31] into NSN P systems in this study. The FRNSN P reasoning algorithm is designed based on the operating mechanism of the FRNSN P system. The FRNSN P systems are used to model the faults, and the FRNSN P reasoning algorithm is used to diagnose the faults of the induction motors. In addition, the relative preference relationship is used to estimate the severity of various potential faults of the motors in order to detect the faults in a timely manner. The contributions of this work are summarized as follows:

1. The NSN P system, as a combination of the SNP system and the NP system, is applied to motor fault diagnosis for the first time. In order to prove its ability to deal with induction motor fault diagnosis, the NSN P system is used to solve the SAT problem first. The results show that the NSN P system can successfully solve the SAT problem in six steps;
2. The IVTFNs are applied to the NSN P system, and the FRNSN P system is proposed to deal with the incompleteness and uncertainty of motor fault information. The FRNSN P system can successfully model the fault fuzzy production rules of induction motors;
3. A FRNSN P reasoning algorithm is designed by using the operating mechanism of FRNSN P systems, making the motor fault diagnosis intelligent;

4. The relative preference relationship is used to estimate the severity of multiple faults when they occur, so as to diagnose the faults in a timely manner and to prevent the deterioration of the faults.

The rest of this paper is organized as follows. Section 2 provides preliminaries on the IVTFNs and the relative preference relations. Section 3 presents the NSN P systems, shows a computational example, and gives the definition of the FRNSN P systems. Section 4 describes the fuzzy reasoning process of the FRNSN P systems and designs the FRNSN P reasoning algorithm. Section 5 reports the computational results to show the effectiveness of the FRNSN P reasoning algorithm for fault diagnosis of induction motors.

2. Preliminaries

2.1. The Interval-Valued Triangular Fuzzy Number

An IVTFN is defined as $A = \left[A^L A^U\right] = \left[\left(a_l^L, a_h^L, a_r^L; w_A^L\right), \left(a_l^U, a_h^U, a_r^U; w_A^U\right)\right]$, where A^L and A^U represent the lower and upper limits of A and $A^L \subseteq A^U$. When $w_A^L = w_A^U = 1$ and $a_h^L = a_h^U$, the form of A becomes $A = \left[A^L, A^U\right] = \left[\left(a_l^U, a_l^L\right), \left(a_h^L = a_h^U\right), \left(a_r^L, a_r^U\right)\right] = \left[\left(a_l^U, a_l^L\right), a_h, \left(a_r^L, a_r^U\right)\right]$, which is called a normalized IVTFN (NIVTFN). An NIVTFN is shown in Figure 1, where $\mu_A(x)$ is the membership function representing the degree of the membership of x, and $\mu_{A^L}(x)$ and $\mu_{A^U}(x)$ are the lower and the upper bounds of $\mu_A(x)$ [31].

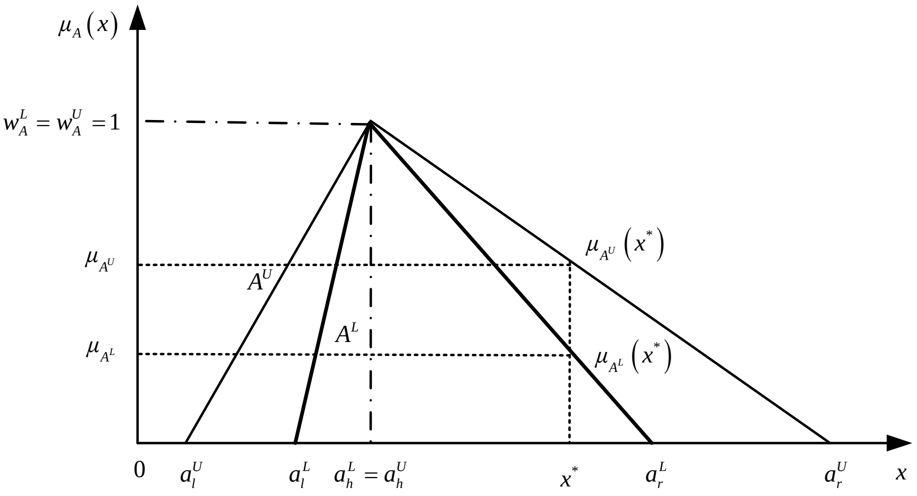


Figure 1. An NIVTFN A.

2.2. The Relative Preference Relation

Suppose $\Psi = \{A_1, A_2, \ldots, A_n\}$ is a set of n IVTFNs, with $A_i = \left[\left(a_{il}^U, a_{il}^L\right), a_{ih}, \left(a_{ir}^L, a_{ir}^U\right)\right]$ for $i = 1, 2, \ldots, n$. The average of A_i is represented by $\overline{A}$ given by $\overline{A} = \left[\left(\overline{a}_l^U, \overline{a}_l^L\right), \overline{a}_h, \left(\overline{a}_r^L, \overline{a}_r^U\right)\right]$. A membership function $\mu_\beta\left(A_i, \overline{A}\right) \in [0,1]$ with a relative preference relation β expresses the preference of A_i to $\overline{A}$. A $\mu_\beta\left(A_i, \overline{A}\right) < \frac{1}{2}$ means $\overline{A}$ takes precedence over A_i, and a $\mu_\beta\left(A_i, \overline{A}\right) > \frac{1}{2}$ means A_i takes precedence over $\overline{A}$. The membership function $\mu_\beta\left(A_i, \overline{A}\right)$ is defined in (1) as follows

$$\mu_\beta\left(A_i, \overline{A}\right) = \frac{1}{2}\left(p \times \frac{\left(a_{il}^L - \overline{a}_r^L\right) + 2(a_{ih} - \overline{a}_h) + \left(a_{ir}^L - \overline{a}_l^L\right)}{2 \times \|T_S^{L_+}, T_S^{L_-}\|} + (1-p) \times \frac{\left(a_{il}^U - \overline{a}_r^U\right) + 2(a_{ih} - \overline{a}_h) + \left(a_{ir}^U - \overline{a}_l^U\right)}{2 \times \|T_S^{U_+}, T_S^{U_-}\|}\right) \mathit{for}\ 0 \le p \le 1, \quad (1)$$

where $$\|T_S^{L+},T_S^{L-}\| = \begin{cases} \frac{\left(t_{sl}^{L+}-t_{sr}^{L-}\right)+2\left(t_{sh}^{+}-t_{sh}^{-}\right)+\left(t_{sr}^{L+}-t_{sl}^{L-}\right)}{2}, & \text{if } t_{sl}^{L+} \geq t_{sr}^{L-} \\ \frac{\left(t_{sl}^{L+}-t_{sr}^{L-}\right)+2\left(t_{sh}^{+}-t_{sh}^{-}\right)+\left(t_{sr}^{L+}-t_{sl}^{L-}\right)}{2} + 2\left(t_{sr}^{L-} - t_{sl}^{L+}\right), & \text{if } t_{sl}^{L+} < t_{sr}^{L-} \end{cases},$$

$$\|T_S^{U+},T_S^{U-}\| = \begin{cases} \frac{\left(t_{sl}^{U+}-t_{sr}^{U-}\right)+2\left(t_{sh}^{+}-t_{sh}^{-}\right)+\left(t_{sr}^{U+}-t_{sl}^{U-}\right)}{2}, & \text{if } t_{sl}^{U+} \geq t_{sr}^{U-} \\ \frac{\left(t_{sl}^{U+}-t_{sr}^{U-}\right)+2\left(t_{sh}^{+}-t_{sh}^{-}\right)+\left(t_{sr}^{U+}-t_{sl}^{U-}\right)}{2} + 2\left(t_{sr}^{U-} - t_{sl}^{U+}\right), & \text{if } t_{sl}^{U+} < t_{sr}^{U-} \end{cases}, \text{ and}$$

$$\left.\begin{array}{l} T_S^{L-} = \left(t_{sl}^{L-}, t_{sh}^{-}, t_{sr}^{L-}\right) = \left(\min\{a_{il}^{L}\}, \min\{a_{ih}\}, \min\{a_{ir}^{L}\}\right) \\ T_S^{L+} = \left(t_{sl}^{L+}, t_{sh}^{+}, t_{sr}^{L+}\right) = \left(\max\{a_{il}^{L}\}, \max\{a_{ih}\}, \max\{a_{ir}^{L}\}\right) \\ T_S^{U-} = \left(t_{sl}^{U-}, t_{sh}^{-}, t_{sr}^{U-}\right) = \left(\min\{a_{il}^{U}\}, \min\{a_{ih}\}, \min\{a_{ir}^{U}\}\right) \\ T_S^{U+} = \left(t_{sl}^{U+}, t_{sh}^{+}, t_{sr}^{U+}\right) = \left(\max\{a_{il}^{U}\}, \max\{a_{ih}\}, \max\{a_{ir}^{U}\}\right) \end{array}\right., \text{for } i = 1, 2, \ldots, n.$$

In the above relative preference relationship, the coefficients p and $1 - p$ are the weights of the lower interval A^L and the upper interval A^U, respectively. The value of p, called the relative preference relation value, is generally determined subjectively, and several different values are usually considered. A good relative preference relationship has a value of p close to 1, and a poor relative preference relationship has a value of p close to 0. Therefore, the relative merits of IVTFNs in a specific set can be quickly judged by the relative preference relation value p [13,31,32].

3. The NSN P System and Its Extension to the FRNSN P System

The NSN P systems are described and their computational power is demonstrated by solving SAT problems. The FRNSN P system is then defined by introducing the IVTFNs into the NSN P system, which lays the foundation for fault diagnosis of induction motors.

3.1. The NSN P System

The NSN P system, described in detail below, has a slightly different threshold from that used in the literature [30,33].

An NSN P system is defined as a tuple as shown in (2) below:

$$\prod = (\sigma_1, \sigma_2, \ldots, \sigma_l, syn, in, out), \tag{2}$$

where $l \geq 1$ is the degree of the NSN P system. The notations in this definition are given below.

(1) $\sigma_1, \sigma_2, \ldots, \sigma_l$ represent l neurons with the form $\sigma_k = (\theta_k, Var_k, \Pr_k, Var_k(0))$, for $1 \leq k \leq l$, where

  (a) $\theta_k \in \mathbb{Z}$ is the threshold of neuron σ_k;

  (b) $Var_k = \{ x_{w,k} | 1 \leq w \leq h_k \}$ is a set of variables in neuron σ_k, where h_k is the number of variables in σ_k;

  (c) $Var_k(0) = \{ x_{w,k}(0) | x_{w,k}(0) \in \mathbb{R}, 1 \leq w \leq h_k \}$ refers to the set of initial values of the variables in the set Var_k;

  (d) $\Pr_k = \{ pr_{\mathrm{P},k} = F_{\mathrm{P},k}(x_{1,k}, \ldots, x_{h_k,k}) | 1 \leq \mathrm{P} \leq h'_k \}$ is a set of programs, where F is called a production function in neuron σ_k, where h'_k is the number of programs in σ_k.

(2) $syn = \{ (k,j) | 1 \leq k, j \leq l, k \neq j \}$ is the set of synapses.

(3) in and out correspond to the input neuron σ_{in} and the output neuron σ_{out}, respectively.

In NSN P system Π, $x_{w,k}$ and $pr_{\mathrm{P},k}$ represent variable w and program P in neuron σ_k, respectively. When neuron σ_k has only one variable or only one program, w or P is omitted from the subscripts. At time t, the value of variable $x_{w,k}$ is represented by $x_{w,i}(t)$ and the production value of program $pr_{\mathrm{P},k}$ is represented by $pr_{\mathrm{P},k}(t) = F_{\mathrm{P},k}(x_{1,k}(t), \ldots, x_{h_k,k}(t))$, i.e., the production value $pr_{\mathrm{P},k}(t)$ is determined by the values of the variables $x_{1,k}, \ldots, x_{h_k,k}$ at time t. Each neuron in Π has a threshold θ_k, and program $pr_{\mathrm{P},k}$ will be applied only when $pr_{\mathrm{P},k}(t) \geq \theta_k$. Once $pr_{\mathrm{P},k}$ is applied, meaning neuron σ_k fires, the values of the variables $x_{1,k}, \ldots, x_{h_k,k}$ are reset to 0 and $pr_{\mathrm{P},i}(t)$ is simultaneously transmitted to the variables of the postsynaptic neurons of neurons σ_k. If $pr_{\text{P},k}(t) < \theta_k$, neuron σ_k will not fire and $pr_{\text{P},k}(t)$ will disappear at this moment.

If the sum of the production values received by variable $x_{w,k}$ at time t is $pr(t)$, then $x_{w,k}(t+1)$ is updated according to (3) in the following:

$$x_{w,k}(t+1) = \begin{cases} pr(t), & \text{if the application of program } pr_{\text{P},k} \text{ involves variable } x_{w,k} \\ pr(t) + x_{w,k}(t), & \text{if the application of program } pr_{\text{P},k} \text{ does not involve variable } x_{w,k} \end{cases} \quad (3)$$

All neurons work in parallel in the NSN P system, and each neuron applies one program at most at each moment. If more than one program can be applied, only one can be selected non-deterministically.

3.2. An Application to the SAT Problem

A SAT problem checks whether the variables of a given Boolean formula can be consistently replaced with the values TRUE and FALSE such that the formula evaluates to TRUE. The instances of SAT problems are determined by two parameters m and n representing the numbers of clauses and variables, respectively. Given a set of Boolean variables $\mathbb{Q} = \{q_1, q_2, \ldots, q_n\}$, a clause C can be expressed in the form $q_1(\neg q_1) \vee \cdots \vee q_i(\neg q_i) \vee \cdots \vee q_n(\neg q_n)$, where $\vee$ indicates the disjunction. A $q_i = 1$ means that q_i is assigned a true value. In general, if $q_i = 1$, then $\neg q_i = 0$, and if $q_i = 0$, then $\neg q_i = 1$. As long as a variable in C is given a true value, C is assigned a value of 1, meaning C is satisfiable. The SAT problem is stated as:

INSTANCE: A clause set $\mathbb{C} = \{C_1, C_2, \ldots, C_m\}$, constructed from a finite set $\{q_1, q_2, \ldots, q_n\}$ of Boolean variables.

TASK: Find if there is an assignment of the variables $q_1, q_2, \ldots, q_n$ satisfying all the clauses in $\mathbb{C}$.

When the assignment of the variables satisfies all the clauses, $\mathbb{C}$ is satisfiable and each clause C_j, for $1 \leq j \leq m$, is given a value of 1. In the following, the SAT problems are solved uniformly with a family of NSN P systems.

The NSN P systems, working non-deterministically, can solve the SAT problem in finite time steps. The general structure of the NSN P systems is shown in Figure 2, with modules Q_i, for $1 \leq i \leq n$, and Y_j, for $1 \leq j \leq m$, corresponding to variables q_i and clauses C_j, respectively. Each module Q_i has three synapses connected to module Y_j.

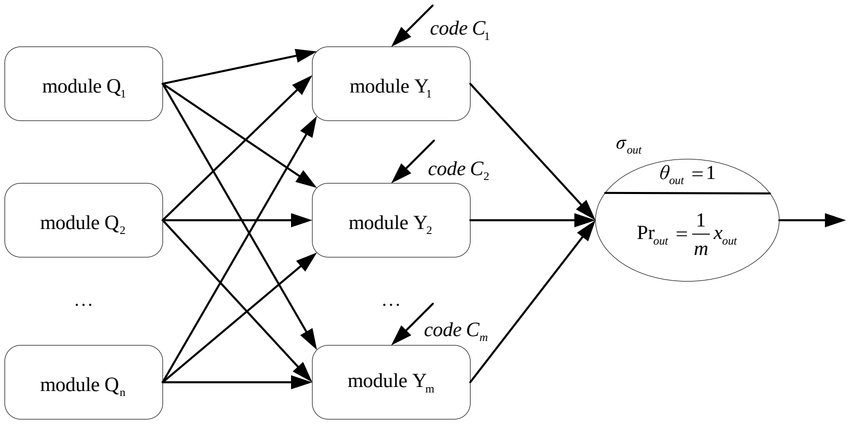


Figure 2. The structure of the NSN P systems for solving the SAT problems.

The following method is used to encode a given SAT instance in order to obtain a uniform solution. A propositional formula $\gamma = C_1 \wedge C_2 \wedge \ldots \wedge C_m$ is considered in the conjunctive normal form, where $\wedge$ indicates the conjunction. Since variable q_i may or may not appear in a clause C_j and can or cannot be negated when it appears, two bits binary numbers are used to code the relationship between q_i and C_j with 00 indicating q_i not appearing in C_j, 01 or equivalently 10 indicating q_i appearing in C_j, and 11 indicating $\neg q_i$ appearing in C_j.

Each clause corresponds to an input neuron, and a sequence of $2n$ digits of 0 s and 1 s is introduced into the input neuron to describe the clause. Therefore, $2n$ steps are required to input the code of the clause with n variables. For example, $\gamma = (\neg q_1 \vee q_2) \wedge (q_1 \vee \neg q_3)$ is a propositional formula composed of clauses $C_1 = \neg q_1 \vee q_2$ and $C_2 = q_1 \vee \neg q_3$, and the sequences 110100 and 010011 corresponding to clauses C_1 and C_2 will be introduced into the associated input neurons within six steps, respectively.

Module Q_i is shown in Figure 3. The neurons σ_{c_1}, σ_{c_2}, σ_{c_3} and σ_{c_4} in each module Q_i are allowed to appear only once in order to reduce the computational complexity. Initially only variable x_{c_1} of neuron σ_{c_1} is assigned a value of 1. Module Q_i non-deterministically produces a truth assignment for variable q_i by non-deterministically choosing a program between $\mathrm{Pr}_{1,d_i} = x_{d_i}$ and $\mathrm{Pr}_{2,d_i} = x_{d_i} - 1$ in neuron σ_{d_i}. Neuron σ_{e_i} will fire if program $\mathrm{Pr}_{1,d_i} = x_{d_i}$ is applied and will not fire if program $\mathrm{Pr}_{2,d_i} = x_{d_i} - 1$ is applied. In this way, neuron σ_{e_i} transmits the value of 1 or nothing to neuron σ_{z_j} in module Y_j. Then q_i is assigned the true value if the value of 1 is transmitted. In addition to feeding neuron σ_{d_i}, neuron σ_{c_1} initially transmits a value of 1 to neuron σ_{c_2}. This value is transmitted along the path $\sigma_{c_3} \to \sigma_f \to \cdots \to \sigma_g$ or $\sigma_{c_4} \to \sigma_f \to \cdots \to \sigma_g$ to neuron σ_{z_j} in module Y_j.

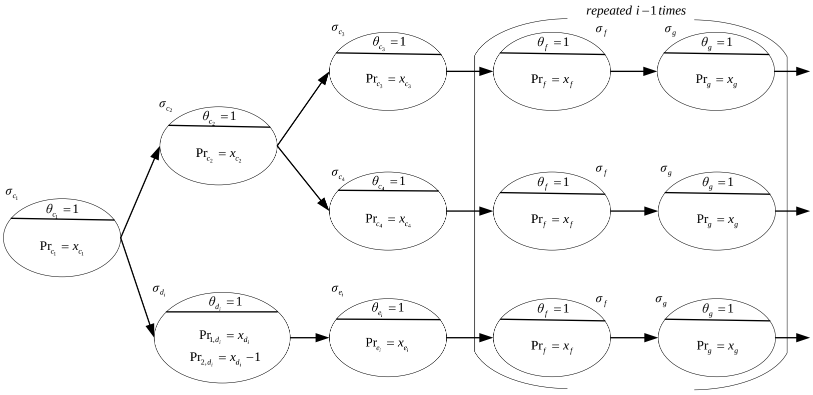


Figure 3. Module Q_i.

Delay neurons labeled σ_f and σ_g are used to maintain the synchronization of the transmission, i.e., neuron σ_{z_j} receives the value from module Q_i and the value from the input neuron associated with clause C_j simultaneously. For example, module Q_1 does not need delay neurons and module Q_2 needs two delay neurons per row. By analogy, each row of module Q_i needs $i - 1$ pairs of delay neurons to guarantee synchronization. Therefore, in step $1 + 2i$, neuron σ_{z_j} receives the assignment of variable q_i and the value from the input neuron. Further processing will be carried out in module Y_j, as shown in Figure 4.

In steps $3, 5, \ldots, 2n + 1$, neuron σ_{z_j} may receive the following values:

2 if $q_i = 0$, but q_i and $\neg q_i$ do not appear in C_j,
3 if $q_i = 1$, but q_i and $\neg q_i$ do not appear in C_j,
3 if $q_i = 0$, but q_i appears and $\neg q_i$ does not appear in C_j,
4 if $q_i = 1$, but q_i appears and $\neg q_i$ does not appear in C_j,
4 if $q_i = 0$, but $\neg q_i$ appears and q_i does not appear in C_j,
5 if $q_i = 1$, but $\neg q_i$ appears and q_i does not appear in C_j.

Program $\mathrm{Pr}_z = \frac{1}{4} x_z$ in neuron σ_{z_j} will be activated and will produce a value of 1 in two cases, one is when $q_i = 1$ and q_i appears in C_j and the other is when $q_i = 0$ and $\neg q_i$ appears in C_j. In either case, the assignment of variable q_i satisfies clause C_j. Neuron $\sigma_{z'_j}$ is used to ensure that σ_{z_j} fires only once by passing the production value -5 to variable x_{z_j}. In this way, it also ensures that variable x_{out} receives a value of 1 at most once.

In step $2i + 2$, if all clauses are satisfied, the sum of the values received by variable x_{out} is m, and neuron σ_{out} fires. So far, it shows that there is a variable assignment so that the proposition formula γ is satisfiable. Therefore, NSN P systems, containing a total of $6n^2 - n + 2m + 1$ neurons working non-deterministically, can solve the SAT problem in finite time steps.

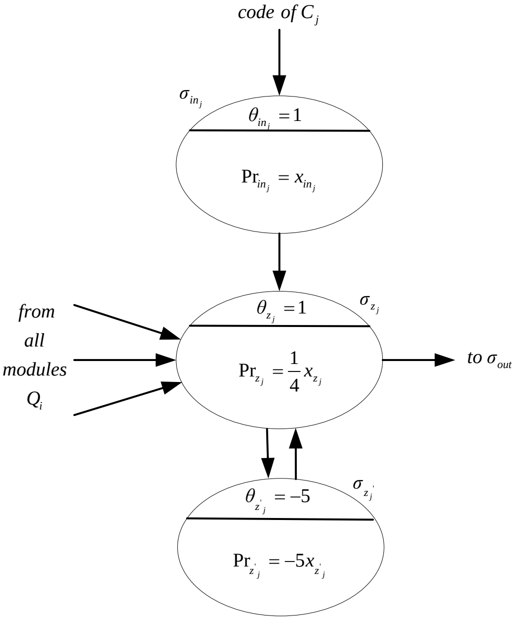


Figure 4. Module Y_j.

The computation time can be shortened by using more input neurons after modifying modules Y_j, $j = 1, 2, \ldots, m$. The structure of the modified module Y_j is shown in Figure 5. The modified module Y_j uses n input neurons to introduce the binary code of a clause in two steps instead of bit by bit in one input neuron. Each of these n neurons receives a two bit binary number 00, 01 (or 10) or 11. When receiving a value of 1, the input neuron $\sigma_{in_{j,i}}$ will fire and transmit a value of 1 to neurons $\sigma_{z_{j,i}}$, $1 \leq i \leq n$ and $1 \leq j \leq m$. Neurons $\sigma_{z_{j,i}}$ and σ_z in module Y_j have similar structures and perform the same functions, i.e., checking whether the assignment of variable q_i satisfies clause C_j.

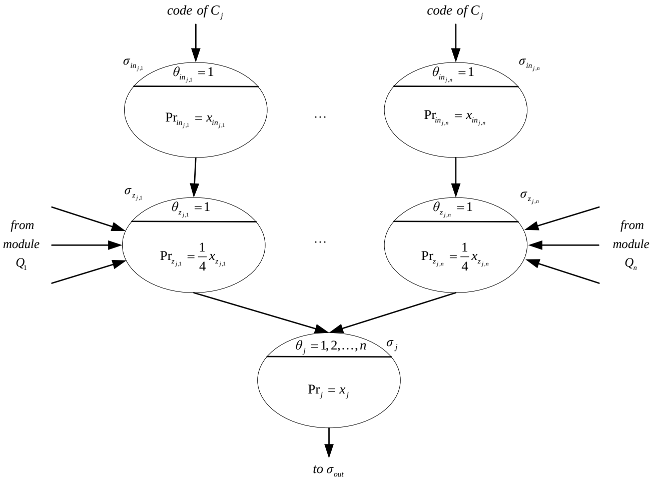


Figure 5. Modified module Y_j.

The delay neurons σ_f and σ_g in module Q_i are no longer needed while all other parts remain unchanged. In step 3, module Q_i, for $1 \leq i \leq n$, also transmits values of 2 or 3 to all neurons $\sigma_{z_{j,i}}$. All variables and all clauses are then checked in parallel. In step 4, if the assignment of variable q_i satisfies clause C_j, the program in neuron $\sigma_{z_{j,i}}$ will be enabled. As a result, variable x_j of neuron σ_j may receive values of $1, 2, \ldots, n$. No matter which value

σ_j receives, program $\text{Pr}_{1,j} = x_j$ will be enabled in step 5 due to the values of the threshold $\theta_j = 1, 2, \ldots, n$.

The firing of neuron σ_j shows that C_j is satisfiable. Each neuron σ_j is connected to neuron σ_{out}. In step 6, if the sum of the values received by neuron σ_{out} is m, the assignments of variables $q_1, q_2, \ldots, q_n$ all satisfy the clauses in $\mathbb{C}$. At the same time, program $\text{Pr}_{out} = \frac{1}{m} x_{out}$ is applied and neuron σ_{out} fires. Obviously, the SAT problem is solved in six steps. With the modified modules Y_j, the NSN P systems use a total of $2nm + m + 2n + 5$ neurons. The computation time is greatly reduced compared to the system with the original modules Y_j although it is not clear about how many more neurons are needed.

To show the computational power of the NSN P system, its time steps for solving the SAT problem are compared with those of DDSN P systems [34], WSN P systems [35] and SN P systems with neuron division and budding [36]. The comparisons are shown in Table 1. Obviously, the NSN P system can solve the SAT problem with the fewest steps.

Table 1. Comparisons of time steps of different P systems for solving the SAT problem.

Modules	Time Steps
NSN P systems	6
DDSN P systems [34]	$2n + m + 3$
WSN P systems [35]	$2n + m + 3$
SN P systems with neuron division and budding [36]	$2n + mn + 6$

3.3. Definition of the FRNSN P System

The FRNSN P system is presented in this subsection as an extension of the NSN P system. A fuzzy reasoning numerical spiking neural P (FRNSN P) system of degree m is defined in (4) as follows

$$\Pi = (\gamma, syn, in, out) \tag{4}$$

where $\gamma = \gamma_p \cup \gamma_r = \{\sigma_1, \ldots, \sigma_l\}$ is a set of l neurons, with $\gamma_p = \{\sigma_1, \ldots, \sigma_s\}$ representing the set of proposition neurons and $\gamma_r = \{\sigma_{s+1}, \ldots, \sigma_{s+n}\}$ representing the set of rule neurons, such that $l = s + n$. Each proposition neuron has the form of $\sigma_i = (\theta_i, x_i, \text{Pr}_i, x_i(0))$, for $1 \le i \le s$, and each rule neuron has the form of $\sigma_j = \big(\theta_j, c_j, x_j, \text{Pr}_j, x_j(0)\big)$, for $1 \le j \le n$. The details of the notations in the definition of Π are given below.

(1) (a) $\theta_k \in \Psi$ is the firing threshold of neuron σ_k, for $1 \le k \le l$;
(b) $c_j \in \Psi$ indicates the confidence factor of neuron σ_j, for $1 \le j \le n$.
(c) x_k is the variable of neuron σ_k, for $1 \le k \le l$;
(d) $x_k(0)$ is the initial fuzzy value of variable x_k, for $1 \le k \le l$.
(e) $\text{Pr}_k = \left\{ pr_k = F_k(x_k) = \begin{cases} x_k, & \text{if neuron } \sigma_k \text{ is a propositinal neuron} \\ x_k c_k, & \text{if neuron } \sigma_k \text{ is a rule neuron} \end{cases} \right.$, is a set of programs, where F is called the production function, for $1 \le k \le l$.

(2) $syn \subseteq \{1, 2, \ldots, l\} \times \{1, 2, \ldots, l\}$ with $(k, k) \notin syn$ is the set of synapses.

(3) in and out correspond to the input neuron σ_{in} and the output neuron σ_{out}, respectively.

In the FRNSN P system, each neuron contains only one variable and one program, each threshold θ_k, each confidence factor c_j or the initial value of the variable x_k is an NIVTFN and each program has only two special forms $pr_k = x_k$ and $pr_k = x_k c_j$. Everything else in the FRNSN P system is the same as that in the NSN P system.

For convenience and intuition, NIVTFNs are associated with some linguistic semantics. The linguistic semantics used in this work are widely used in the literature [13,15,31] and are shown in Table 2. These linguistic semantics vividly reflect the probability that an event occurs.

Table 2. The correspondence between linguistic terms and NIVTFNs.

Linguistic Terms	NIVTFNs
Extremely Low (EL)	$[(0,0),0,(0,0)]$
Very Low (VL)	$[(0,0.06),0.12,(0.18,0.23)]$
Low (L)	$[(0.20,0.24),0.27,(0.30,0.39)]$
Fairly Low (FL)	$[(0.33,0.36),0.44,(0.46,0.52)]$
Medium (M)	$[(0.454,0.48),0.52,(0.55,0.64)]$
Fairly High (FH)	$[(0.62,0.642),0.67,(0.721,0.78)]$
High (H)	$[(0.73,0.79),0.82,(0.84,0.90)]$
Very High (VH)	$[(0.86,0.90),0.93,(0.97,1)]$
Extremely High (EH)	[(1.00, 1.00), 1.00, (1.00, 1.00)]

In addition, the following arithmetic and logic operations, involved in the operations of the FRNSN P system, are defined.

Premise: $A = \left[\left(a_l^U, a_l^L\right), a_h, \left(a_r^L, a_r^U\right)\right]$ and $B = \left[\left(b_l^U, b_l^L\right), b_h, \left(b_r^L, b_r^U\right)\right]$ are two NIVTFNs, with a and b being real numbers in the interval [0, 1].

Given the above premise, the following arithmetic operation is defined:

$$A \times B = \left[\left(a_l^U \times b_l^U, a_l^L \times b_l^L\right), a_h \times b_h, \left(a_r^L \times b_r^L, a_r^U \times b_r^U\right)\right]$$

Given the above premise, the following logical operations are defined:

(1) *And*: $A \wedge B = \left[\left(a_l^U \wedge b_l^U, a_l^L \wedge b_l^L\right), a_h \wedge b_h, \left(a_r^L \wedge b_r^L, a_r^U \wedge b_r^U\right)\right]$, where $a \wedge b = \min(a,b)$;
(2) *Or*: $A \vee B = \left[\left(a_l^U \vee b_l^U, a_l^L \vee b_l^L\right), a_h \vee b_h, \left(a_r^L \vee b_r^L, a_r^U \vee b_r^U\right)\right]$, where $a \vee b = \max(a,b)$;
(3) If $a \geq b$, then $A \geq B$.

4. The FRNSN P Reasoning Algorithm

This section first uses the FRNSN P system to model the fuzzy production rules of the induction motors, and then proposes the FRNSN P reasoning algorithm based on the reasoning process of the FRNSN P system.

4.1. Modeling and Fuzzy Reasoning

Fuzzy production rules are usually used for knowledge representation, and the following three types of fuzzy production rules are involved in this work:

General rule R_j: IF p_1, THEN p_2 $\left(C = c_j\right)$;
And rule R_j: IF p_1 AND p_2 AND . . . AND p_{s-1}, THEN p_s $\left(C = c_j\right)$;
Or rule R_j: IF p_1 OR p_2 OR . . . OR p_{s-1}, THEN $p_s\left(C = c_j\right)$;

where $p_1, \ldots, p_s$ are fuzzy propositions, and $C = c_j$ represents the credibility of the fuzzy production rule R_j.

The FRNSN P system is used to model the above three types of fuzzy production rules. Four types of, i.e., proposition, *G-rule*, *A-rule* and *O-rule*, neurons, as shown in Figure 6, are used in the FRNSN P systems. A proposition neuron represents a fuzzy proposition. The *G-rule*, *A-rule* and *O-rule* neurons represent the three types of rules, as discussed below.

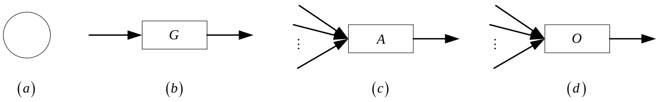


Figure 6. Four types of neurons: (**a**) proposition neuron, (**b**) *G-rule* neuron, (**c**) *A-rule* neuron and (**d**) *O-rule* neuron.

The *General rule* is modeled by FRNSN P system Π_1 shown in Figure 7 (Π_1). System Π_1 is specified in (5) as follows

$$\Pi_1 = (\{\sigma_1, \sigma_2, \sigma_3\}, syn, in, out) \tag{5}$$

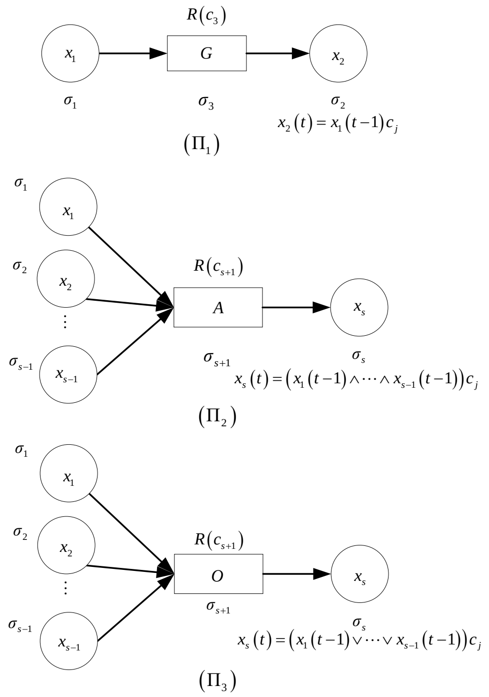


Figure 7. FRNSN P systems modeling the three types of fuzzy production rules.

The details of the notations used in system Π_1 are given below.

(1) $\sigma_i = (\theta_i, Var_i, \mathrm{Pr}_i, Var_i(0))$ is a proposition neuron representing fuzzy propositions p_i for $i = 1, 2$;
(2) $\sigma_3 = (\theta_3, c_3, Var_3, \mathrm{Pr}_3, Var_3(0))$ is a *G-rule* neuron;
(3) $syn = \{(\sigma_1, \sigma_3), (\sigma_3, \sigma_2)\}$ is the set of synapses;
(4) $in = \{\sigma_1\}$ and $out = \{\sigma_2\}$ are the input and output proposition neurons.

The fuzzy reasoning process is automatically performed as follows. Initially, the variable of neuron σ_1 is assigned a value of $x_1(0)$. Neuron σ_1 fires and the variable of neuron σ_3 receives the production value $pr_1(0)$ if $pr_1(0) = x_1(0) \geq \theta_1$, or does not fire and the value $pr_1(0)$ disappears otherwise, at time $t = 0$. When rule neuron σ_3 satisfies the firing condition, it fires and transmits the production value of $pr_3(1) = x_3(1)c_3 = x_1(0)c_3$ to variable x_2 at time $t = 1$. Thus, the value $x_1(0)c_3$ is the result of the computation of system Π_1.

The *AND rule* is modeled by FRNSN P system Π_2 shown in Figure 7 (Π_2). System Π_2 is specified in (6) as follows

$$\Pi_2 = (\{\sigma_1, \ldots, \sigma_s, \sigma_{s+1}\}, syn, in, out) \tag{6}$$

The details of the notations used in system Π_2 are given below.

(1) $\sigma_i = (\theta_i, Var_i, \mathrm{Pr}_i, Var_i(0))$ is a proposition neuron representing fuzzy proposition p_i for $1 \leq i \leq s$;
(2) $\sigma_{s+1} = (\theta_{s+1}, c_{s+1}, Var_{s+1}, \mathrm{Pr}_{s+1}, Var_{s+1}(0))$ is an *A-rule* neuron;
(3) $syn = \{(\sigma_1, \sigma_{s+1}), \ldots, (\sigma_{s-1}, \sigma_{s+1}), (\sigma_{s+1}, \sigma_s)\}$ is the set of synapses;
(4) $in = \{\sigma_1, \ldots, \sigma_{s-1}\}$ and $out = \{\sigma_s\}$ are the set of input neurons and the output neuron.

The fuzzy reasoning process is automatically performed as follows. The variables of neurons $\sigma_1, \ldots, \sigma_{s-1}$ are assigned initial values of $x_1(0), \ldots, x_{s-1}(0)$, respectively. For $1 \leq i \leq s-1$, neuron σ_i fires and $x_{s+1}(t) = x_1(0) \wedge \ldots \wedge x_{s-1}(0)$ if $pr_i(t) \geq \theta_i$, and does not fire and $pr_i(t)$ disappears if $pr_i(t) < \theta_i$. When the *A-rule* neuron fires the next time, the production value $pr_{s+1}(t) = x_{s+1}(t)c_{s+1}$ will be transmitted to variable x_s. Therefore, the result computed by Π_2 is $x_s = x_{s+1}(t)c_{s+1}$.

The *OR rule* is modeled by FRNSN P system Π_3, as shown in Figure 7 (Π_3). System Π_3 is specified in (7) as follows

$$\Pi_3 = (\{\sigma_1, \ldots, \sigma_s, \sigma_{s+1}\}, syn, in, out) \tag{7}$$

The details of the notations used in system Π_3 are given below:

(1) $\sigma_i = (\theta_i, Var_i, \Pr_i, Var_i(0))$ is a proposition neuron representing fuzzy proposition p_i for $1 \leq i \leq s$;
(2) $\sigma_{s+1} = (\theta_{s+1}, c_{s+1}, Var_{s+1}, \Pr_{s+1}, Var_{s+1}(0))$ is the *O-rule* neuron;
(3) $syn = \{(\sigma_1, \sigma_{s+1}), \ldots, (\sigma_{s-1}, \sigma_{s+1}), (\sigma_{s+1}, \sigma_s)\}$ is the set of synapses;
(4) $in = \{\sigma_1, \ldots, \sigma_{s-1}\}$ and $out = \{\sigma_s\}$ are the set of input neurons and the output neuron.

The fuzzy reasoning process of system Π_3 is similar to that of system Π_2, and its description is omitted.

4.2. The FRNSN P Reasoning Algorithm

This subsection introduces the FRNSN P reasoning algorithm, as detailed in Algorithm 1. The related matrices, vectors and multiplication operators, as well as a function, are introduced first. The flowchart of the FRNSN P reasoning algorithm is then presented.

(1) $X_p(t) = (x_1(t), \ldots, x_s(t))^T$ is a vector consisting of the fuzzy values of the s variables contained in the s proposition neurons, where $x_i(t)$ is an NIVTFN, for $1 \leq i \leq s$;
(2) $X_r(t) = (x_1(t), \ldots, x_n(t))^T$ is a vector consisting of the fuzzy values of the n variables contained in the n rule neurons, where $x_j(t)$ is an NIVTFN, for $1 \leq j \leq n$;
(3) $\Theta = (\theta_1, \ldots, \theta_l)^T$ is a vector consisting of the l firing thresholds of the l neurons, where θ_k is an NIVTFN, for $1 \leq k \leq l$;
(4) $C = diag(c_1, \ldots c_n)$ is a diagonal matrix consisting of the confidence factors of the n rule neurons, where c_j, for $1 \leq j \leq n$, is the confidence factor of neuron σ_j, an NIVTFN, representing the credibility of the fuzzy production rule R_j;
(5) $D_1 = \left(d_{ij}^{(1)}\right)_{s \times n}$ is a matrix representing the synaptic connections from proposition neurons to $G - rule$ neurons, such that $d_{ij}^{(1)} = 1$ if a synapse exists from proposition neuron σ_i to $G - rule$ neuron σ_j, and $d_{ij}^{(1)} = 0$ otherwise, for $1 \leq i \leq s$ and $1 \leq j \leq n$;
(6) $D_2 = \left(d_{ij}^{(2)}\right)_{s \times n}$ is a matrix representing the synaptic connections from proposition neurons to $A - rule$ neurons, such that $d_{ij}^{(2)} = 1$ if a synapse exists from proposition neuron σ_i to $A - rule$ neuron σ_j, and $d_{ij}^{(2)} = 0$ otherwise, for $1 \leq i \leq s$ and $1 \leq j \leq n$;
(7) $D_3 = \left(d_{ij}^{(3)}\right)_{s \times n}$ is a matrix representing the synaptic connections from proposition neurons to $O - rule$ neurons such that $d_{ij}^{(3)} = 1$ if a synapse exists from proposition neuron σ_i to $O - rule$ neuron σ_j, and $d_{ij}^{(3)} = 0$ otherwise, for $1 \leq i \leq s$ and $1 \leq j \leq n$;
(8) $E = (e_{ji})_{n \times s}$ is a matrix representing the synaptic connections from rule neurons to proposition neurons such that $e_{ji} = 1$ if a synapse exists from rule neuron σ_j to proposition neuron σ_j, and $e_{ji} = 0$ otherwise, for $1 \leq i \leq s$ and $1 \leq j \leq n$;
(9) $V_p(t) = (v_{p1}(t), \ldots, v_{ps}(t))^T$ is a vector consisting of the values passed by proposition neuron σ_i to the postsynaptic rule neuron variable. If neuron σ_i does not have a postsynaptic neuron, then this value is passed to the environment as the output value. In particular, $v_{pi}(0) = 0$, for $1 \leq i \leq s$;
(10) $V_r(t) = (v_{r1}(t), \ldots, v_{rn}(t))^T$ is a vector consisting of the values passed by rule neuron σ_j to the postsynaptic proposition neuron variable. In particular, $v_{rj}(0) = 0$ for $1 \leq j \leq n$.

In addition, several multiplication operators for the above matrices and vectors are defined:

(1) $C \otimes X_r(t) = (c_1 x_1(t), \ldots, c_n x_n(t))^T$. Similarly, $D_1{}^T \otimes X_p(t) = \left(\overline{d}_1^{(1)}, \overline{d}_2^{(1)}, \ldots \overline{d}_n^{(1)}\right)^T$, where $\overline{d}_j^{(1)} = d_{1j}^{(1)} x_1(t) + d_{2j}^{(1)} x_2(t) + \ldots + d_{sj}^{(1)} x_s(t)$, for $1 \le j \le n$;

(2) $D_2^T \odot X_p(t) = \left(\overline{d}_1^{(2)}, \overline{d}_2^{(2)}, \ldots \overline{d}_n^{(2)}\right)^T$, where $\overline{d}_j^{(2)} = d_{1j}^{(2)} x_1(t) \vee d_{2j}^{(2)} x_2(t) \vee \ldots \vee d_{sj}^{(2)} x_s(t)$, for $1 \le j \le n$;

(3) $D_3^T \oplus X_p(t) = \left(\overline{d}_1^{(3)}, \overline{d}_2^{(3)}, \ldots \overline{d}_n^{(3)}\right)^T$, where $\overline{d}_j^{(3)} = d_{1j}^{(3)} x_1(t) \wedge d_{2j}^{(3)} x_2(t) \wedge \ldots \wedge d_{sj}^{(3)} x_s(t)$, for $1 \le j \le n$.

Finally, a function (8) for production value $pr_k(t)$ and threshold θ is defined.

$$v_k(t) = \begin{cases} pr_k(t), & \text{if } pr_k(t) \ge \theta_k \\ 0, & \text{otherwise} \end{cases}, \; 1 \le k \le l, \; s + n = l. \tag{8}$$

Algorithm 1: The FRNSN P reasoning algorithm

Input: Θ, C, D_1, D_2, D_3, E, $X_P(0)$, $X_r(0)$

```
Let $t = 0$;
Set the stopping condition $0_r = (0, \ldots, 0)_n^T$;
  while ($X_r(t) \neq 0_r$) do
    for each of the (input) proposition neurons do
      if the proposition neuron has a postsynaptic rule neuron then
        Calculate $X_r(t) = (D_1^T \otimes V_p(t)) + (D_2^T \odot V_p(t)) + (D_3^T \oplus V_p(t))$;
        if $pr_i(t) \ge \theta_i$ then
          Transmits the value $v_{pi}(t)$ to the rule neuron;
        else
          Transmits the value 0 to the rule neuron;
        end if
      end if
    end for
    for each of the rule neurons do
      if $pr_j(t) \ge \theta_j$ then
        Transmits the value $v_{rj}(t)$ to the postsynaptic proposition neuron;
        Calculate $X_p(t) = E^T \oplus (C \otimes V_r(t))$;
      end if
    end for
    $t = t + 1$;
end while
```

Output: The fuzzy values of the output proposition neurons.

Matrices Θ, C and $X_p(0)$ were obtained from expert experience and historical data, and matrices D_1, D_2, D_3 and E were obtained from the topology of the FRNSN P system. The flowchart of the FRNSN P reasoning algorithm is shown in Figure 8.

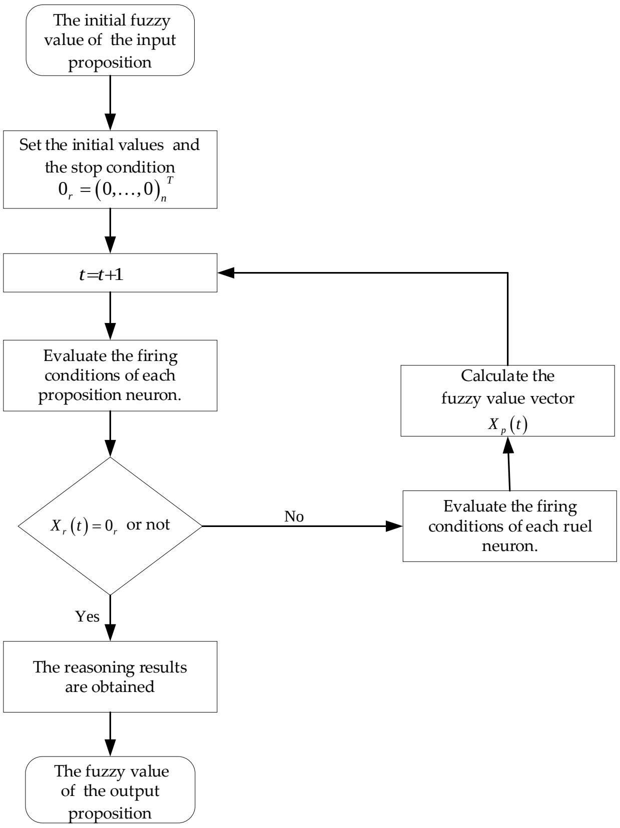

The initial fuzzy value of the input proposition
Set the initial values and the stop condition
$0_r = (0,\ldots,0)_n^T$
t=t+1
Evaluate the firing conditions of each proposition neuron.
$X_r(t) = 0_r$ or not
No
Yes
Evaluate the firing conditions of each ruel neuron.
Calculate the fuzzy value vector
$X_p(t)$
The reasoning results are obtained
The fuzzy value of the output proposition

Figure 8. The flowchart of the FRNSN P reasoning algorithm.

5. Fault Diagnosis of Induction Motors Using the FRNSN P Reasoning Algorithm

The fault mechanism of induction motors is complex, and the relationship between a fault and a symptom is not one-to-one correspondent but is complex. Generally, a fault manifests as multiple symptoms and different faults may correspond to the same symptom [37–39]. The faults of induction motors are mostly related to windings, bearings and rotors. The single-fault cases "Winding insulation burnt", "Bearing damage" and "Broken rotor bar", and the multiple-fault cases "Winding insulation burnt and bearing damage" and "Bearing damage and broken rotor bar", as listed in Table 3, were investigated using Algorithm 1. Due to the similarity of the reasoning processes, the multi-fault case "Winding insulation burnt and bearing damage" is used as an example for detailed description. A flowchart showing the induction motor fault diagnosis process is in Figure 9.

Table 3. Comparisons of the reasoning results of FRNSN P and the other three methods.

		Preset					Result	
	Cases	**Fault Locations**	**Fault Symptoms**	**Fault Cases**	**Methods**	**Fault Events**	**Fault Sources**	**Fault Cases**
1	Broken rotor bar	σ_{38}	$\sigma_{12}(H)$	$\sigma_{12}, \sigma_{23}, \sigma_{34}$	FFPN [37]	σ_{38}	σ_{12}	$\sigma_{12}, \sigma_{23}, \sigma_{34}$
					CLPSO-FPN [38]	σ_{38}	σ_{12}	$\sigma_{12}, \sigma_{23}, \sigma_{34}$
					rMFRSNPs [40]	σ_{38}	σ_{12}	$\sigma_{12}, \sigma_{23}, \sigma_{34}$
					FRNSN P	σ_{38}	σ_{12}	$\sigma_{12}, \sigma_{23}, \sigma_{34}$
2	Winding insulation burnt	σ_{36}	$\sigma_2(FH), \sigma_3(H)$	$\sigma_2, \sigma_3, \sigma_{17}, \sigma_{27}$	FFPN [37]	σ_{36}	σ_2, σ_3	$\sigma_2, \sigma_3, \sigma_{17}, \sigma_{27}$
					CLPSO-FPN [38]	σ_{36}	σ_2, σ_3	$\sigma_2, \sigma_3, \sigma_{17}, \sigma_{27}$
					rMFRSNPs [40]	σ_{36}	σ_2, σ_3	$\sigma_2, \sigma_3, \sigma_{17}, \sigma_{27}$
					FRNSN P	σ_{36}	σ_2, σ_3	$\sigma_2, \sigma_3, \sigma_{17}, \sigma_{27}$

Table 3. *Cont.*

		Preset				Result		
	Cases	**Fault Locations**	**Fault Symptoms**	**Fault Cases**	**Methods**	**Fault Events**	**Fault Sources**	**Fault Cases**
3	Bearing damage	σ_{37}	$\sigma_8(H), \sigma_9(FH)$	$\sigma_8, \sigma_{21}, \sigma_{32}$	FFPN [37]	σ_{37}	σ_8, σ_9	$\sigma_8, \sigma_9, \sigma_{21}, \sigma_{32}$
					CLPSO-FPN [38]	σ_{37}	σ_8, σ_9	$\sigma_8, \sigma_9, \sigma_{21}, \sigma_{32}$
					rMFRSNPs [40]	σ_{37}	σ_8	$\sigma_8, \sigma_{21}, \sigma_{32}$
					FRNSN P	σ_{37}	σ_8	$\sigma_8, \sigma_{21}, \sigma_{32}$
4	Bearing damage and broken rotor bar	σ_{37}, σ_{38}	$\sigma_8(H), \sigma_{10}(FH), \sigma_{11}(VH)$	$\sigma_8, \sigma_{11}, \sigma_{21}, \sigma_{22}, \sigma_{23}, \sigma_{32}, \sigma_{33}, \sigma_{34}$	FFPN [37]	σ_{38}	σ_{11}	$\sigma_{11}, \sigma_{23}, \sigma_{34}$
					CLPSO-FPN [38]	σ_{37}, σ_{38}	σ_8, σ_{11}	$\sigma_8, \sigma_{11}, \sigma_{22}, \sigma_{23}, \sigma_{32}, \sigma_{33}, \sigma_{34}$
					rMFRSNPs [40]	σ_{37}, σ_{38}	σ_8, σ_{11}	$\sigma_8, \sigma_{11}, \sigma_{21}, \sigma_{22}, \sigma_{23}, \sigma_{32}, \sigma_{33}, \sigma_{34}$
					FRNSN P	σ_{37}, σ_{38}	σ_8, σ_{11}	$\sigma_8, \sigma_{11}, \sigma_{21}, \sigma_{22}, \sigma_{23}, \sigma_{32}, \sigma_{33}, \sigma_{34}$
5	Winding insulation burnt and bearing damage	σ_{36}, σ_{37}	$\sigma_2(FH), \sigma_3(VH), \sigma_5(VH), \sigma_6(H), \sigma_8(H), \sigma_9(H), \sigma_{10}(FH), \sigma_{13}(FH)$	$\sigma_3, \sigma_5, \sigma_6, \sigma_8, \sigma_9, \sigma_{17}, \sigma_{19}, \sigma_{20}, \sigma_{21}, \sigma_{27}, \sigma_{29}, \sigma_{30}, \sigma_{31}, \sigma_{32}$	FFPN [37]	σ_{36}	$\sigma_3, \sigma_5, \sigma_6$	$\sigma_3, \sigma_5, \sigma_6, \sigma_{17}, \sigma_{20}, \sigma_{27}, \sigma_{30}$
					CLPSO-FPN [38]	σ_{36}	$\sigma_3, \sigma_5, \sigma_6$	$\sigma_3, \sigma_5, \sigma_6, \sigma_{17}, \sigma_{20}, \sigma_{27}, \sigma_{30}$
					rMFRSNPs [40]	$\sigma_{36}, \sigma_{37}, \sigma_{38}$	$\sigma_2, \sigma_3, \sigma_5, \sigma_6, \sigma_8, \sigma_9, \sigma_{13}$	$\sigma_2, \sigma_3, \sigma_5, \sigma_6, \sigma_8, \sigma_9, \sigma_{13}, \sigma_{17}, \sigma_{19}$ $\sigma_{20}, \sigma_{21}, \sigma_{27}$ $\sigma_{29}, \sigma_{30}, \sigma_{31}, \sigma_{32}$
					FRNSN P	σ_{36}, σ_{37}	$\sigma_3, \sigma_5, \sigma_6, \sigma_8, \sigma_9$	$\sigma_3, \sigma_5, \sigma_6, \sigma_8, \sigma_9, \sigma_{17}, \sigma_{19}, \sigma_{20}, \sigma_{21}, \sigma_{27}, \sigma_{29}, \sigma_{30}, \sigma_{31}, \sigma_{32}$

5.1. Fuzzy Production Rules for Induction Motors

The fuzzy production rules related to motor faults are presented and the relevant fault events are enumerated, as shown in Figure 10 [37,38,40]. There is a one-to-one correspondence between fault events and propositions in fuzzy production rules. Fault events 36, 37 and 38 are the immediate causes of "motor fault", and the motor is considered faulty whichever of the three faults occurs. The events in bold in Figure 10 are fault symptom events of faults 36, 37 and 38, and event 7 is a symptom of all the three faults. The fuzzy production rules are listed below:

R_1: IF p_1, THEN P_{15} $(C = c_1)$;
R_2: IF p_2 AND p_3, THEN P_{16} $(C = c_2)$;
R_3: IF p_3, THEN P_{17} $(C = c_3)$;
R_4: IF p_4, THEN P_{18} $(C = c_4)$;
R_5: IF p_5, THEN P_{19} $(C = c_5)$;
R_6: IF p_6 OR p_7, THEN P_{20} $(C = c_6)$;
R_7: IF p_8 OR p_9, THEN P_{21} $(C = c_7)$;
R_8: IF p_{10} OR p_{11}, THEN P_{22} $(C = c_8)$;
R_9: IF p_{11}, THEN P_{23} $(C = c_9)$;
R_{10}: IF p_{12}, THEN P_{24} $(C = c_{10})$;
R_{11}: IF p_7, THEN P_{25} $(C = c_{11})$;
R_{12}: IF p_{13} OR p_{14}, THEN P_{26} $(C = c_{12})$;
R_{13}: IF p_{15} OR p_{16} OR p_{17}, THEN P_{27} $(C = c_{13})$;
R_{14}: IF p_{18}, THEN P_{28} $(C = c_{14})$;
R_{15}: IF p_{19}, THEN P_{29} $(C = c_{15})$;

R_{16}: IF p_{20}, THEN P_{30} $(C = c_{16})$;
R_{17}: IF p_{20}, THEN P_{31} $(C = c_{17})$;
R_{18}: IF p_{21}, THEN P_{32} $(C = c_{18})$;
R_{19}: IF p_{22}, THEN P_{33} $(C = c_{19})$;
R_{20}: IF p_{23} OR p_{24} OR p_{25}, THEN P_{34} $(C = c_{20})$;
R_{21}: IF p_{26}, THEN P_{35} $(C = c_{21})$;
R_{22}: IF p_{27} OR p_{28} OR p_{29} OR p_{30}, THEN P_{36} $(C = c_{22})$;
R_{23}: IF p_{31} OR p_{32} OR p_{33}, THEN P_{37} $(C = c_{23})$;
R_{24}: IF p_{34} OR p_{35}, THEN P_{38} $(C = c_{24})$;
R_{25}: IF p_{36} OR p_{37} OR p_{38}, THEN P_{39} $(C = c_{25})$.

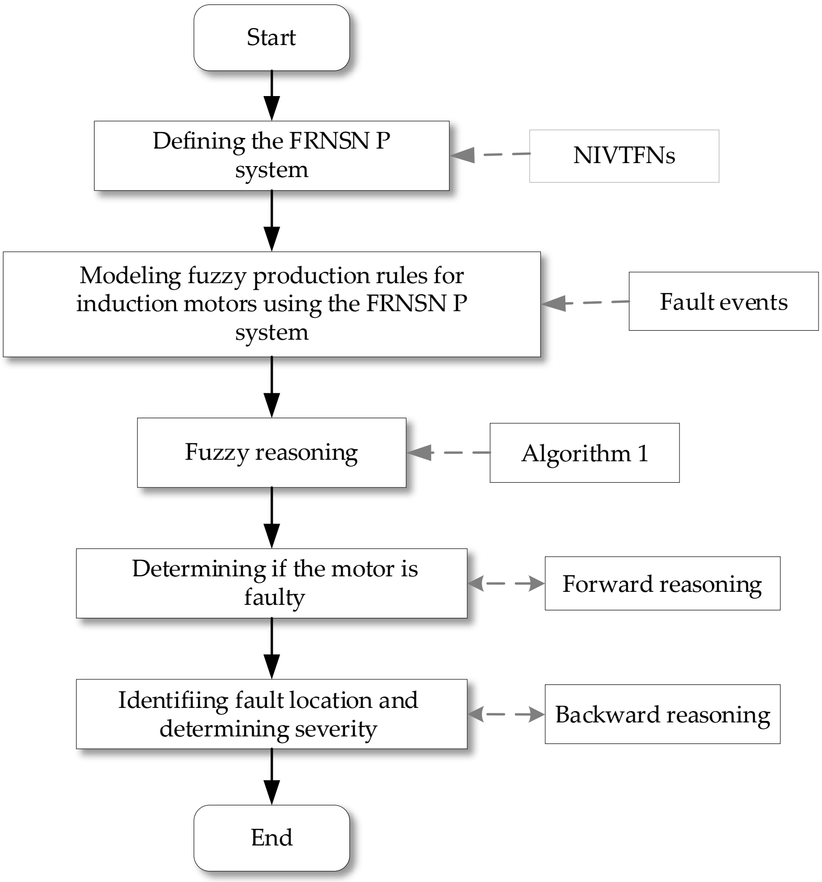
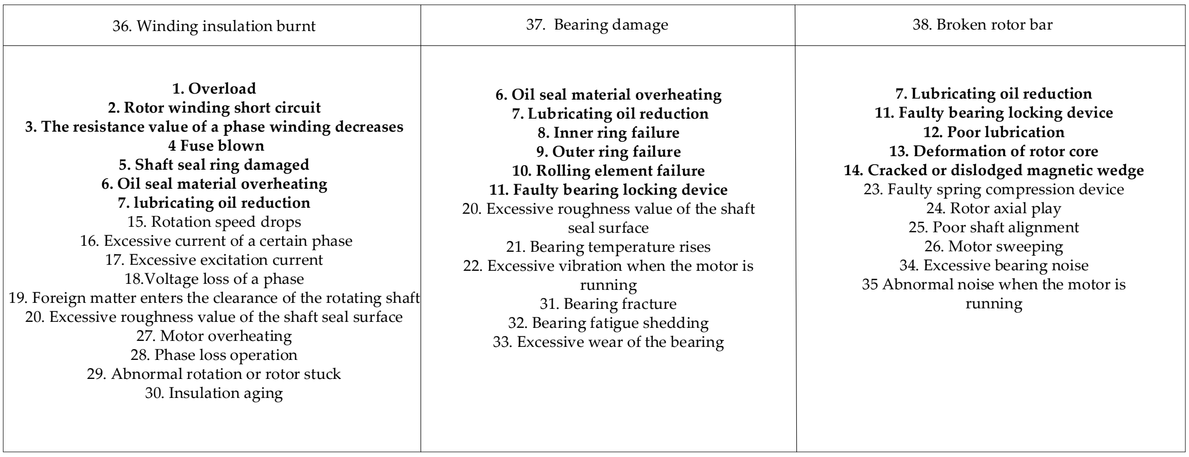

Start
Defining the FRNSN P system
NIVTFNs
Modeling fuzzy production rules for induction motors using the FRNSN P system
Fault events
Fuzzy reasoning
Algorithm 1
Determining if the motor is faulty
Forward reasoning
Identifiing fault location and determining severity
Backward reasoning
End

Figure 9. The flowchart of induction motor fault diagnosis process using Algorithm 1.



36. Winding insulation burnt	37. Bearing damage	38. Broken rotor bar
1. Overload **2. Rotor winding short circuit** **3. The resistance value of a phase winding decreases** **4 Fuse blown** **5. Shaft seal ring damaged** **6. Oil seal material overheating** **7. lubricating oil reduction** 15. Rotation speed drops 16. Excessive current of a certain phase 17. Excessive excitation current 18.Voltage loss of a phase 19. Foreign matter enters the clearance of the rotating shaft 20. Excessive roughness value of the shaft seal surface 27. Motor overheating 28. Phase loss operation 29. Abnormal rotation or rotor stuck 30. Insulation aging	**6. Oil seal material overheating** **7. Lubricating oil reduction** **8. Inner ring failure** **9. Outer ring failure** **10. Rolling element failure** **11. Faulty bearing locking device** 20. Excessive roughness value of the shaft seal surface 21. Bearing temperature rises 22. Excessive vibration when the motor is running 31. Bearing fracture 32. Bearing fatigue shedding 33. Excessive wear of the bearing	**7. Lubricating oil reduction** **11. Faulty bearing locking device** **12. Poor lubrication** **13. Deformation of rotor core** **14. Cracked or dislodged magnetic wedge** 23. Faulty spring compression device 24. Rotor axial play 25. Poor shaft alignment 26. Motor sweeping 34. Excessive bearing noise 35 Abnormal noise when the motor is running

Figure 10. Fault events related to motor faults.

5.2. Parameter Settings

The relevant parameters of the FRNSN P reasoning algorithm are specified in this subsection. The confidence factors c_j for $1 \leq j \leq n$ of the *O-rule* neurons, the *G-rule* neurons and the *A-rule* neurons were set to $EH = [(1.00,\ 1.00),\ 1.00,\ (1.00,\ 1.00)]$, $VH = [(0.86, 0.90), 0.93, (0.97, 1)]$ and $H = [(0.73, 0.79), 0.82, (0.84, 0.90)]$, respectively, based on experience and historical data [39,40]. The firing thresholds θ_k for $1 \leq k \leq l$ of the proposition neurons and the rule neurons were set to $M = [(0.454, 0.48), 0.52, (0.55, 0.64)]$. In addition, if the NIVTFN of the variable in a proposition neuron satisfies $x_i(t) \geq FH = [(0.62, 0.642), 0.67, (0.721, 0.78)]$, then the fault event corresponding to the proposition neuron has indeed occurred. If the NIVTFN of the variable in a proposition neuron satisfies $x_i(t) \leq FL = [(0.33, 0.36), 0.44, (0.46, 0.52)]$, then the fault event corresponding to the proposition neuron has not occurred.

5.3. Case Studies

In this subsection, the potential fault of the motor is modeled using the fuzzy production rules, as shown in Figure 11. The fault diagnosis of the motor was carried out through Algorithm 1. Specifically, fault diagnosis contains two phases. The first phase is forward reasoning, which is to infer whether the motor will fail according to the probability of occurrence of failure events. The second phase is backward reasoning, that is to infer the fault cause and fault path of the motor after determining the motor fault. Suppose that the fault symptom events 2, 3, 5, 6, 8, 9, 10 and 13 occurred according to the online monitoring system, indicating that the initial NIVTFNs of the variables in neurons σ_2, σ_3, σ_5, σ_6, σ_8, σ_9, σ_{10} and σ_{13} are all greater than or equal to FH as defined in Table 2.

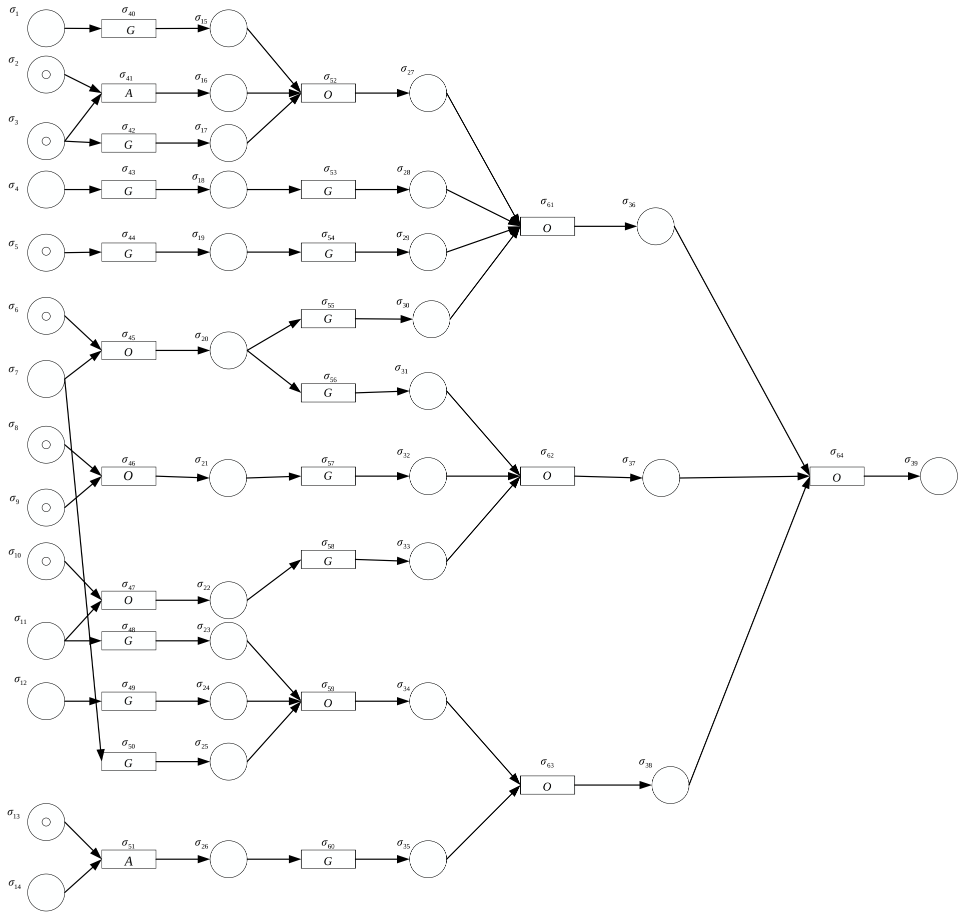


Figure 11. The forward reasoning model for induction motor fault diagnosis using the FRNSN P system.

5.3.1. Forward Reasoning

The threshold vector Θ and the confidence factor matrix C were presented in Section 5.2. The synaptic connection matrices D_1, D_2, D_3 and E are given in the topological structure of the FRNSN P system in Figure 11. The initial IVTFNs of the variables of the input proposition neurons, i.e., the probabilities of occurrences of fault symptom events, were obtained according to the historical data and the experienced fault diagnosis reports in the industry [39,40].

The detailed process of forward reasoning using the FRNSN P system in Figure 11 is as follows. Initially only the variables of the input proposition neurons contain nonzero values. A **0** represents a vector of 0 s, i.e., the NIVTFNs in the neurons are $[(0,0),0,(0,0)]$. An input proposition neuron fires and passes the production value to the rule neurons if it satisfies the threshold condition and does not fire and the contained production value disappears otherwise. The neurons in Figure 11 fire hierarchically and the production values are passed from presynaptic to postsynaptic neurons. According to the fuzzy reasoning process of the three FRNSN P systems in Section 4.1, the values of the variables, represented by the NIVTFNs, in the neurons at each time step are as follows.

$$X_p(0)=\begin{bmatrix}[(0.33,0.36),0.44,(0.46,0.52)]\\ [(0.62,0.642),0.67,(0.721,0.78)]\\ [(0.86,0.90),0.93,(0.97,1)]\\ [(0.20,0.24),0.27,(0.30,0.39)]\\ [(0.86,0.90),0.93,(0.97,1)]\\ [(0.73,0.79),0.82,(0.84,0.90)]\\ [(0.20,0.24),0.27,(0.30,0.39)]\\ [(0.73,0.79),0.82,(0.84,0.90)]\\ [(0.73,0.79),0.82,(0.84,0.90)]\\ [(0.62,0.642),0.67,(0.721,0.78)]\\ [(0.20,0.24),0.27,(0.30,0.39)]\\ [(0,0.06),0.12,(0.18,0.23)]\\ [(0.62,0.642),0.67,(0.721,0.78)]\\ [(0.20,0.24),0.27,(0.30,0.39)]\\ 0\end{bmatrix},\ X_r(0)=[0].$$

When $t=1$,

$$X_r(1)=\begin{bmatrix}[(0,0),0,(0,0)]\\ [(0.62,0.642),0.67,(0.721,0.78)]\\ [(0.86,0.90),0.93,(0.97,1)]\\ [(0,0),0,(0,0)]\\ [(0.86,0.90),0.93,(0.97,1)]\\ [(0.73,0.79),0.82,(0.84,0.90)]\\ [(0.73,0.79),0.82,(0.84,0.90)]\\ [(0.62,0.642),0.67,(0.721,0.78)]\\ 0\end{bmatrix},\ X_p(1)=\begin{bmatrix}0\\ [(0.7396,0.81),0.8649,(0.9409,1)]\\ [(0,0),0,(0,0)]\\ [(0.7396,0.81),0.8649,(0.9409,1)]\\ [(0.73,0.79),0.82,(0.84,0.9)]\\ [(0.73,0.79),0.82,(0.84,0.9)]\\ [(0.62,0.642),0.67,(0.721,0.78)]\\ 0\end{bmatrix}.$$

When $t=2$,

$$X_r(2)=\begin{bmatrix}0\\ [(0.7396,0.81),0.8649,(0.9409,1)]\\ [(0,0),0,(0,0)]\\ [(0.7396,0.81),0.8649,(0.9409,1)]\\ [(0.73,0.79),0.82,(0.84,0.9)]\\ [(0.73,0.79),0.82,(0.84,0.9)]\\ [(0.73,0.79),0.82,(0.84,0.9)]\\ [(0.62,0.642),0.67,(0.721,0.78)]\\ 0\end{bmatrix},\ X_p(2)=\begin{bmatrix}0\\ [(0.7396,0.81),0.8649,(0.9409,1)]\\ [(0,0),0,(0,0)]\\ [(0.6361,0.729),0.8044,(0.9127,1)]\\ [(0.6278,0.711),0.7626,(0.8148,0.9)]\\ [(0.6278,0.711),0.7626,(0.8148,0.9)]\\ [(0.6278,0.711),0.7626,(0.8148,0.9)]\\ [(0.5332,0.5778),0.6231,(0.6994,0.78)]\\ 0\end{bmatrix}.$$

When $t=3$,

$$X_r(3)=\begin{bmatrix}0\\ [(0.7396,0.81),0.8649,(0.9409,1)]\\ [(0.6278,0.711),0.7626,(0.8148,0.9)]\\ 0\end{bmatrix},\ X_p(3)=\begin{bmatrix}0\\ [(0.7396,0.81),0.8649,(0.9409,1)]\\ [(0.6278,0.711),0.7626,(0.8148,0.9)]\\ 0\end{bmatrix}.$$

When $t=4$,

$$X_r(4) = \begin{bmatrix} 0 \\ [(0.7396, 0.81), 0.8649, (0.9409, 1)] \\ 0 \end{bmatrix}, X_p(4) = \begin{bmatrix} 0 \\ [(0.7396, 0.81), 0.8649, (0.9409, 1)] \\ 0 \end{bmatrix}.$$

When $t = 5$,

$$X_r(5) = [0]$$

When the computation completes at $t = 4$, the value of the variable in the output proposition neuron σ_{39} is $[(0.7396, 0.81), 0.8649, (0.9409, 1)]$. The output proposition neuron σ_{39} fires since it satisfies the firing condition $[(0.7396, 0.81), 0.8649, (0.9409, 1)] \geq M$ at $t = 5$. Therefore, $X_r(5) = [0]$, the stopping condition is satisfied, the algorithm terminates and the reasoning result is obtained. The fault event corresponding to the output proposition neuron σ_{39} occurs, i.e., the motor is faulty, since $[(0.7396, 0.81), 0.8649, (0.9409, 1)] \geq FH$.

5.3.2. Backward Reasoning

After the induction motor is determined to be faulty, the computation results of the FRNSN P reasoning algorithm are used to perform backward reasoning to find out the fault event, fault source and the fault propagation path. The backward reasoning model is shown in Figure 12.

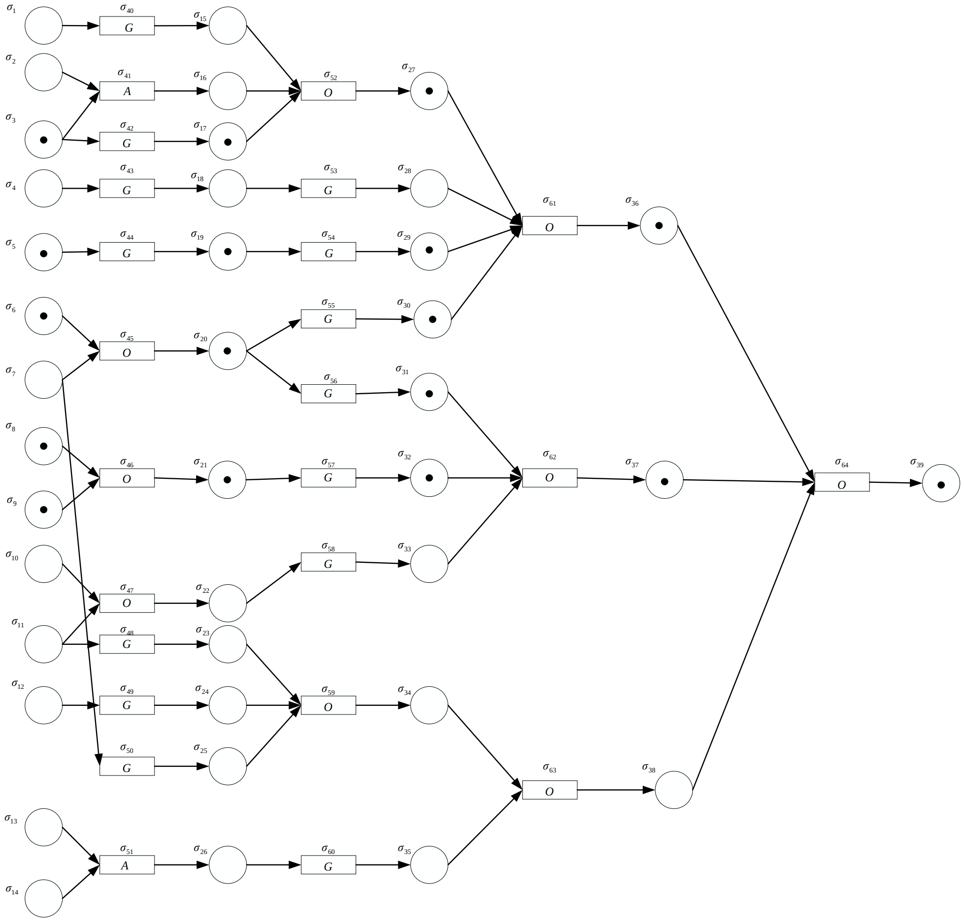


Figure 12. The backward reasoning model for induction motor fault diagnosis using the FRNSN P system.

The immediate cause of the motor failure can be determined from the threshold conditions. Since the confidence factors of propositions p_{36}, p_{37} and p_{38} are $[(0.7396, 0.81), 0.8649, (0.9409, 1)] \geq FH$, $[(0.6278, 0.711), 0.7626, (0.8148, 0.90)] \geq FH$ and $[(0.2838, 0.324), 0.4092, (0.4462, 0.52)] \leq FL$, respectively, the fault events “Winding insulation burnt” and “Bearing damage”, but not “Broken rotor bar”, are determined to have occurred.

The fault propagation path generally begins with the source of the fault and ends at the immediate cause of the motor fault. In this case, there are six fault propagation paths, including $L_1 = \{\sigma_3 \rightarrow \sigma_{17} \rightarrow \sigma_{27} \rightarrow \sigma_{36} \rightarrow \sigma_{39}\}$, $L_2 = \{\sigma_5 \rightarrow \sigma_{19} \rightarrow \sigma_{29} \rightarrow \sigma_{36} \rightarrow \sigma_{39}\}$, $L_3 = \{\sigma_6 \rightarrow \sigma_{20} \rightarrow \sigma_{30} \rightarrow \sigma_{36} \rightarrow \sigma_{39}\}$, $L_4 = \{\sigma_6 \rightarrow \sigma_{20} \rightarrow \sigma_{31} \rightarrow \sigma_{37} \rightarrow \sigma_{39}\}$, $L_5 = \{\sigma_8 \rightarrow \sigma_{21} \rightarrow \sigma_{32} \rightarrow \sigma_{37} \rightarrow \sigma_{39}\}$ and $L_6 = \{\sigma_9 \rightarrow \sigma_{21} \rightarrow \sigma_{32} \rightarrow \sigma_{37} \rightarrow \sigma_{39}\}$. It can be found that the fault events 2, 10 and 13 cannot ultimately lead to motor failure even if they also occur, i.e., they are not fault sources for motor failure.

Next, the severity of "Winding insulation burnt" and "Bearing failure" are determined by computing the relative preference values for proposition neurons σ_{37} and σ_{38}. Let $A_1 = [(0.7396, 0.81), 0.8649, (0.9409, 1)]$ and $A_2 = [(0.6278, 0.711), 0.7626, (0.8148, 0.90)]$, then $\overline{A} = [(0.6837, 0.7605), 0.81375, (0.87785, 0.95)]$ $T_S^{L-} = \left(t_{sl}^{L-}, t_{sh}^{-}, t_{sr}^{L-}\right) = (0.711, 0.7626, 08148)$, $T_S^{L+} = \left(t_{sl}^{L+}, t_{sh}^{+}, t_{sr}^{L+}\right) = (0.81, 08649, 0.9409)$, $T_S^{U-} = \left(t_{sl}^{U-}, t_{sh}^{-}, t_{sr}^{U-}\right) = (0.6278, 0.7626, 0.9)$, $T_S^{U+} = \left(t_{sl}^{U+}, t_{sh}^{+}, t_{sr}^{U+}\right) = (0.7396, 0.8649, 1)$, $\|T_S^{L+}, T_S^{L-}\| = 0.22445$, $\|T_S^{U+}, T_S^{U-}\| = 0.529$, $\mu_\beta(A_1, \overline{A}) = 0.669$ and $\mu_\beta(A_2, \overline{A}) = 0.331$. Due to the relative preference value $\mu_\beta(A_1, \overline{A}) > \mu_\beta(A_2, \overline{A})$, the fault of "Winding insulation burnt" is more serious. When multiple faults occur in the motor, the introduction of the relative preference relation β can help determine the severities of all the faults.

Finally, the performance of the FRNSN P system was compared with those of other motor fault diagnosis methods. The methods used for comparison were FFPN [37], CLPSO-FPN [38] and rMFRSNPs [40], and the comparison results are shown in Table 3. Fault events, fault symptoms, fault sources and fault cases are represented by corresponding neurons.

For the single-fault cases "Winding insulation burnt" and "Broken rotor bar", FFPN [37], CLPSO-FPN [38], rMFRSNPs [40] and FRNSN P could correctly detect the fault events and obtain the same fault sources and fault causes. The fault event of case 3 was "Bearing damage". Although all four methods could obtain the correct detection results, FFPN [37] and CLPSO-FPN [38] found one more fault source, i.e., neuron σ_9, than rMFRSNPs [40] and FRNSN P did.

For the multi-fault case "Bearing damage and broken rotor bar", CLPSO-FPN [38], rMFRSNPs [40] and FRNSN P gave correct and consistent results, but FFPN [37] could only detect one of the faults, i.e., "Broken rotor bar". For the multi-fault case "Winding insulation burnt and bearing damage" detailed in this subsection, FRNSN P showed certain advantages, i.e., it could correctly detect faults "Winding insulation burnt" and "Bearing damage", but FFPN [37] and CLPSO-FPN [38] could only detect fault "Winding insulation burnt". Although rMFRSNPs [40] could also detect faults "Winding insulation burnt" and "Bearing damage", fault "Broken rotor bar" that did not exist was also detected. In addition, the fault sources were slightly different for each method. Since FFPN and CLPSO-FPN could only detect fault event 36, their fault sources were only associated with event 36. Since rMFRSNPs detected one more fault, it found more fault sources than FRNSN P did. For the same case, it is reasonable and acceptable for different methods to have slightly different fault sources due to different operating mechanisms and parameter settings.

6. Conclusions

In this work, the NSN P systems were extended to the FRNSN P systems by introducing IVTFNs. FRNSN P systems can easily model the fuzzy production rules of motor faults. A fuzzy reasoning algorithm based on the FRNSN P system was proposed for motor fault diagnosis. Through the study of single fault and multiple fault cases, the effectiveness and feasibility of the FRNSN P reasoning algorithm were proved for motor fault diagnosis. In addition, the relative preference relationship can be used to estimate the severity of various faults, so that the motor can be repaired in time when a minor fault occurs to prevent the fault from worsening.

Since it is necessary to rely on historical data and expert experience to obtain the probability of occurrence of motor fault symptoms, signal processing technology will be combined with the FRNSN P system to obtain real-time motor fault information in a future study. Specifically, considering that the stator current signal is minimally affected by the external environment and the current sensor is easy to install, the current signal will be used to obtain fault information. According to the fault information, the occurrence probability of some cause events can be obtained early, the fault probability corresponding to the IVTFN can then be estimated, and the FRNSN P reasoning algorithm is finally used for fault diagnosis. Furthermore, other intelligent algorithms can be introduced into the FRNSN P system so as to apply it to other real-world applications including the fault diagnosis of other types of motors.

Author Contributions: Conceptualization, X.Y.; methodology, X.Y.; software, X.Y.; validation, X.Y., M.S. and J.D.; formal analysis, X.Y.; investigation, X.Y.; resources, X.Y.; data curation, X.Y.; writing—original draft preparation, X.Y.; writing—review and editing, X.Y.; visualization, X.Y.; supervision, X.L., M.S., G.Z. and J.D.; project administration, X.Y.; funding acquisition, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research project was funded by the National Natural Science Foundation of China (61876101, 61802234, 61806114), Social Science Fund Project of Shandong Province, China (16BGLJ06, 11CGLJ22), Natural Science Fund Project of Shandong Province, China (ZR2019QF007), Postdoctoral Project, China (2017M612339, 2018M642695), Humanities and Social Sciences Youth Fund of the Ministry of Education, China (19YJCZH244), and Postdoctoral Special Funding Project, China (2019T120607).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, X.; Wang, T.; Ying, R.; Cao, Z. A Fault Diagnosis Method Considering Meteorological Factors for Transmission Networks Based on P Systems. *Entropy* **2021**, *23*, 1008. [CrossRef]
2. Li, X.; Zhang, W.; Ding, Q.; Sun, J.-Q. Intelligent rotating machinery fault diagnosis based on deep learning using data augmentation. *J. Intell. Manuf.* **2018**, *31*, 433–452. [CrossRef]
3. Xiong, G.; Shi, D.; Zhu, L.; Duan, X. A New Approach to Fault Diagnosis of Power Systems Using Fuzzy Reasoning Spiking Neural P Systems. *Math. Probl. Eng.* **2013**, *2013*, 815352. [CrossRef]
4. Wang, T.; Liu, W.; Zhao, J.; Guo, X.; Terzija, V. A rough set-based bio-inspired fault diagnosis method for electrical substations. *Int. J. Electr. Power Energy Syst.* **2020**, *119*, 105961. [CrossRef]
5. Zhao, J.; Hu, T.; Zheng, R.; Ba, P.; Mei, C.; Zhang, Q. Defect Recognition in Concrete Ultrasonic Detection Based on Wavelet Packet Transform and Stochastic Configuration Networks. *IEEE Access* **2021**, *9*, 9284–9295. [CrossRef]
6. Kumar, P.; Hati, A.S. Dilated convolutional neural network based model for bearing faults and broken rotor bar detection in squirrel cage induction motors. *Expert Syst. Appl.* **2022**, *191*, 116290. [CrossRef]
7. Shao, H.; Xia, M.; Han, G.; Zhang, Y.; Wan, J. Intelligent fault diagnosis of rotor-bearing system under varying working conditions with modified transfer CNN and thermal images. *IEEE Trans. Ind. Inform.* **2020**, *17*, 3488–3496. [CrossRef]
8. Toma, R.N.; Piltan, F.; Kim, J.M. A deep autoencoder-based convolution neural network framework for bearing fault classification in induction motors. *Sensors* **2021**, *21*, 8453. [CrossRef]
9. Defdaf, M.; Berrabah, F.; Chebabhi, A.; Cherif, B.D.E. A new transform discrete wavelet technique based on artificial neural network for induction motor broken rotor bar faults diagnosis. *Int. Trans. Electr. Energy Syst.* **2021**, *31*, e12807. [CrossRef]
10. Ibrahim, M.; Saeed, T.; El-Shorbagy, M.A.; Nofal, T.A.; Aamir, N. Implementation of the artificial neural network to predict the effectiveness of the solar system using Cu/water-ethylene nanofluid to save energy. *Eng. Anal. Bound. Elem.* **2022**, *138*, 30–41. [CrossRef]
11. Deng, W.; Zhang, S.; Zhao, H.; Yang, X. A Novel Fault Diagnosis Method Based on Integrating Empirical Wavelet Transform and Fuzzy Entropy for Motor Bearing. *IEEE Access* **2018**, *6*, 35042–35056. [CrossRef]
12. Mejia-Barron, A.; Tapia-Tinoco, G.; Razo-Hernandez, J.R.; Valtierra-Rodriguez, M.; Granados-Lieberman, D. A neural network-based model for MCSA of inter-turn short-circuit faults in induction motors and its power hardware in the loop simulation. *Comput. Electr. Eng.* **2021**, *93*, 107234. [CrossRef]
13. Zhou, N.; Wang, A. Fault diagnosis of transmission circuit based on triangular interval valued fuzzy spike neural network P system. *Energy Rep.* **2022**, *8*, 776–784. [CrossRef]
14. Pang, S.C.; Wang, M.; Qiao, S.B.; Wang, X.; Chen, H.Q. Fault Diagnosis for Service Composition by Spiking Neural P Systems with Colored Spikes. *Chin. J. Electron.* **2019**, *28*, 1033–1040. [CrossRef]
15. Peng, H.; Wang, J.; Ming, J.; Shi, P.; Perez-Jimenez, M.J.; Yu, W.P.; Tao, C.Y. Fault Diagnosis of Power Systems Using Intuitionistic Fuzzy Spiking Neural P Systems. *IEEE Trans. Smart Grid* **2018**, *9*, 4777–4784. [CrossRef]

16. Peng, H.; Wang, J.; Perez-Jimenez, M.J.; Wang, H.; Shao, J.; Wang, T. Fuzzy reasoning spiking neural P system for fault diagnosis. *Inf. Sci.* **2013**, *235*, 106–116. [CrossRef]
17. Sheng, N.; Zhang, D.; Zhang, Q. Fuzzy Command Filtered Backstepping Control for Nonlinear System with Nonlinear Faults. *IEEE Access* **2021**, *9*, 60409–60418. [CrossRef]
18. Wang, T.; Zhang, G.X.; Zhao, J.B.; He, Z.Y.; Wang, J.; Perez-Jimenez, M.J. Fault Diagnosis of Electric Power Systems Based on Fuzzy Reasoning Spiking Neural P Systems. *IEEE Trans. Power Syst.* **2015**, *30*, 1182–1194. [CrossRef]
19. Huang, K.; Wang, T.; He, Y.; Zhang, G.; Pérez-Jiménez, M.J. Temporal Fuzzy Reasoning Spiking Neural P Systems with Real Numbers for Power System Fault Diagnosis. *J. Comput. Theor. Nanosci.* **2016**, *13*, 3804–3814. [CrossRef]
20. Rong, H.; Yi, K.; Zhang, G.; Dong, J.; Paul, P.; Huang, Z. Automatic Implementation of Fuzzy Reasoning Spiking Neural P Systems for Diagnosing Faults in Complex Power Systems. *Complexity* **2019**, *2019*, 2635714. [CrossRef]
21. Wang, T.; Wei, X.; Wang, J.; Huang, T.; Peng, H.; Song, X.; Cabrera, L.V.; Pérez-Jiménez, M.J. A weighted corrective fuzzy reasoning spiking neural P system for fault diagnosis in power systems with variable topologies. *Eng. Appl. Artif. Intell.* **2020**, *92*, 103680. [CrossRef]
22. Ionescu, M.; Păun, G.; Yokomori, T. Spiking neural P systems. *Fundam. Inform.* **2006**, *71*, 279–308.
23. Tang, X.; Zhang, Q.; Dai, X.; Zou, Y. Neural Membrane Mutual Coupling Characterisation Using Entropy-Based Iterative Learning Identification. *IEEE Access* **2020**, *8*, 205231–205243. [CrossRef]
24. Ionescu, M.; Sburlan, D. Some Applications of Spiking Neural P Systems. *Comput. Inform.* **2008**, *27*, 515–528.
25. Dong, J.; Zhang, G.; Luo, B.; Yang, Q.; Guo, D.; Rong, H.; Zhu, M.; Zhou, K. A distributed adaptive optimization spiking neural P system for approximately solving combinatorial optimization problems. *Inf. Sci.* **2022**, *596*, 1–14. [CrossRef]
26. Song, T.; Pang, S.; Hao, S.; Rodríguez-Patón, A.; Zheng, P. A Parallel Image Skeletonizing Method Using Spiking Neural P Systems with Weights. *Neural Process. Lett.* **2018**, *50*, 1485–1502. [CrossRef]
27. Gou, X.; Liu, Q.; Rong, H.; Hu, M.; Paul, P.; Zhang, X.; Yu, Z. A Novel Spiking Neural P System for Image Recognition. *Int. J. Unconv. Comput.* **2021**, *16*, 121–139.
28. Ma, T.; Hao, S.; Wang, X.; Rodriguez-Paton, A.A.; Wang, S.; Song, T. Double Layers Self-Organized Spiking Neural P Systems with Anti-Spikes for Fingerprint Recognition. *IEEE Access* **2019**, *7*, 177562–177570. [CrossRef]
29. Wang, T.; Wei, X.; Huang, T.; Wang, J.; Peng, H.; Perez-Jimenez, M.J.; Valencia-Cabrera, L. Modeling Fault Propagation Paths in Power Systems: A New Framework Based on Event SNP Systems with Neurotransmitter Concentration. *IEEE Access* **2019**, *7*, 12798–12808. [CrossRef]
30. Wu, T.; Pan, L.; Yu, Q.; Tan, K.C. Numerical Spiking Neural P Systems. *IEEE Trans. Neural Netw Learn. Syst.* **2021**, *32*, 2443–2457. [CrossRef]
31. Wang, Y.-J. Interval-valued fuzzy multi-criteria decision-making based on simple additive weighting and relative preference relation. *Inf. Sci.* **2019**, *503*, 319–335. [CrossRef]
32. Wang, Y.-J. Combining technique for order preference by similarity to ideal solution with relative preference relation for interval-valued fuzzy multi-criteria decision-making. *Soft Comput.* **2019**, *24*, 11347–11364. [CrossRef]
33. Yin, X.; Liu, X.; Sun, M.; Ren, Q. Novel Numerical Spiking Neural P Systems with a Variable Consumption Strategy. *Processes* **2021**, *9*, 549. [CrossRef]
34. Zhao, Y.; Liu, X.; Wang, W. Spiking Neural P Systems with Neuron Division and Dissolution. *PLoS ONE* **2016**, *11*, e0162882. [CrossRef]
35. Wang, J.; Hoogeboom, H.J.; Pan, L.; Paun, G.; Perez-Jimenez, M.J. Spiking neural P systems with weights. *Neural Comput.* **2010**, *22*, 2615–2646. [CrossRef]
36. Pan, L.; Păun, G.; Pérez-Jiménez, M.J. Spiking neural P systems with neuron division and budding. *Sci. China Inf. Sci.* **2011**, *54*, 1596–1607. [CrossRef]
37. Cheng, X.; Wang, C.; Yu, Y.; Yi, L.; Chen, Q. An approach for three-phase asynchronous motor failure analysis based on fuzzy fault petri net. *Trans. China Electrotech. Soc.* **2015**, *30*, 132–139. [CrossRef]
38. Cheng, X.; Wang, C.; Li, J.; Bai, X. Adaptive fault diagnosis of motors using comprehensive learning particle swarm optimizer with fuzzy petri net. *Comput. Inform.* **2020**, *39*, 246–263. [CrossRef]
39. Yang, B.-S.; Jeong, S.K.; Oh, Y.-M.; Tan, A.C.C. Case-based reasoning system with Petri nets for induction motor fault diagnosis. *Expert Syst. Appl.* **2004**, *27*, 301–311. [CrossRef]
40. Huang, Z.; Wang, T.; Liu, W.; Valencia-Cabrera, L.; Pérez-Jiménez, M.J.; Li, P.; Geem, Z.W. A Fault Analysis Method for Three-Phase Induction Motors Based on Spiking Neural P Systems. *Complexity* **2021**, *2021*, 2087027. [CrossRef]