Journal of Basic and Applied Engineering Research
p-ISSN: 2350-0077; e-ISSN: 2350-0255; Volume 11, Issue 1; January-March, 2024, pp. 1-7
© Krishi Sanskriti Publications
http://www.krishisanskriti.org/Publication.html

A Novel Hypergraph Numerical Neural-like P System for Brain Tumor Segmentation

Swathika P[1], Manju Bala S[2] and Amirthavarshini B[3]

*[1]Assistant Professor(Sr.G), Department of Artificial Intelligence and Data Science
Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India
[2]Department of Artificial Intelligence and Data Science
Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India
[3]Department of Artificial Intelligence and Data Science,
Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India
E-mail: [1]swathikap@mepcoeng.ac.in, [2]smanjubala18ai@mepcoeng.ac.in, [3]vivek561997_ai@mepcoeng.ac.in*

Abstract—*Spiking neural P systems are parallel and distributed computation devices which are inspired by the neuro-physiological behavior of biological neurons. They are regarded as advanced-generation models in the realm of neural networks. While biological neurons exhibit intricate structures, classical neural-like P systems offer simplifications by representing these structures and their associated mechanisms as either two-dimensional graphs or communication patterns based on tree structures involving firing and forgetting. In this research paper, we introduce a novel numerical neural-like P system referred to as the Hypergraph-based Numerical Neural-like (HNN) P system, which incorporates five distinct types of neurons(Elementary, Parent, Hyperparent, Hyperchild, Neighbor)aimed at capturing high-order correlations within neuron structures. Additionally, we present three innovative communication mechanisms(V-rule, E-rule, H-rule) among these neurons designed to handle numerical variables and functions. Leveraging this newly devised neural-like P system, we have developed a model for segmenting tumors in MRI brain images. Our experimental findings demonstrate that our proposed models surpass the performance of existing state-of-the-art methods when applied to BRATS dataset. Its result affirm the effectiveness of the HNN P system in accurately segmenting tumors.*

Keywords: *Hypergraph, Brain Tumor Segmentation, HNN P System*

INTRODUCTION

Membrane computing models that incorporate the properties of spiking neural network encoding for information over time and membrane computing processing in parallel are known as neural-like P systems. Because neural-like P systems take the notion of spiking neurons into account, they may be thought of as third-generation neural network models. The building blocks of neural-like P systems are always two-dimensional graph structures, where the vertices stand in for individual neurons and the edges for the appropriate synapses between them. These neuron architectures cannot be calculated hierarchically and can only communicate with their connected siblings. Different cell and neuronal systems, notably P systems, have been developed to address this issue. Hypergraphs have hyperedges that connect any number of vertices and can enable more complex relationships between data. Thus, extending membrane structures to a high-dimensional and more comprehensive nonlinear space using hypergraph structures could improve the learning capabilities of P systems.

P systems primarily consist of objects and rules in addition to structures. There are only spikes of firing and forgetting rules to compute numbers in each neuron of a traditional neural-like P system, which also restricts the model's capacity for learning. Extensions of neural-like P systems, such as nonlinear neural-like P systems, inhibitory neural-like P systems, and generalised neural-like P systems, have been proposed to overcome this issue. A numerical neural-like P system with mathematical functions and numerical variables.

In that they use synapses to convey signals from one neuron to another, neural-like P systems' frameworks resemble CNNs. While standard neural networks require 886 sigmoid function-based processors, neural-like P systems can save more computing resources and achieve Turing universality with just 10 neurons. Digital image segmentation, power supply system fault detection, robot control, and other areas have seen some effective approaches. However, in comparison to their theoretical accomplishments, implementations of P systems have advanced slowly due to their limitations on learning capacity and framework. A deep learning model was integrated into spiking neural-like P systems. In order to handle medical imaging data, integrated the automatic extraction of feature information from convolution neural networks (CNNs) into P systems. Particularly, the discrete information processing capability of P systems can convert picture pixels and their local information into objects. The distributed parallel mechanism used by P system rules to implement the automatic information selection of CNNs can successfully select distinct features from various images or regions of images at once, further enhancing the effectiveness and accuracy of brain tumor segmentation. Current research, however, relies on conventional P system topologies and protocols, making it unable to fulfil specialised transmembrane communication goals like multiple feature fusions of MRI brain images.

In this paper, we introduce a novel approach known as the Hypergraph-Based Numerical Neural-Like (HNN) P system, which incorporates five distinct types of neurons to capture the intricate relationships within neuron structures. Additionally, we put forth three innovative communication mechanisms for facilitating interactions among neurons, particularly concerning numerical variables and functions. This framework serves as the foundation for developing a brain tumor segmentation model tailored for tumor and organ detection.

Within this neural-like P system, we have devised specific rules within the neurons to execute both down-sampling and up-sampling feature selections. To further enhance the delineation of boundaries and the characterization of tumor or organ content, we conduct feature fusion operations at various levels, including up-down, forward-backward, and inter-neuron interactions, employing rules that apply to these new neuron types.

Furthermore, we explore the possibility of integrating multiple segmentation models with varying sets of variables into an ensemble learning framework, maximizing the system's parallelism to enhance accuracy and efficiency. To evaluate the effectiveness of our proposed HNN P system, we employ one dataset-the public BRATS dataset for tumor segmentation(targeting multiple brain metastases). Our experiments showcase that the HNN P system achieves state-of-the-art performance in segmentation tasks.

The primary contributions of our work can be summarized as follows:

1. Introduction of a Hypergraph-Based Numerical Neural-Like P system, harnessing the high-order correlation inherent in hypergraphs to enable P systems to perform computations and communications across hierarchical, planar, and transmembrane levels.

2. Design and incorporation of five distinct neuron types, specifically tailored to capture the intricate high-order relationships within neuron structures. We introduce three novel communication mechanisms to enable the handling of numerical variables and functions, thereby enhancing the learning capabilities of HNN P systems and expanding their applicability in real-world scenarios.

3. We have created a MRI brain image tumor segmentation model using the HNN P system. This model seamlessly combines the exceptional accuracy of deep learning functions typically found in semantic segmentation with the strong parallelism and resilience inherent in P systems.

III. PROPOSED SYSTEM

A. Numerical Neural-like P Systems

The components that follow is the structure of a numerical neural-like P system [11] made up of $M > 1$ neurons:

$$\tau = \vartheta_1 + \vartheta_2 + +\vartheta_n + synapse + inputN \; + \; outputN \quad (1)$$

where the dimension of each neuron ϑ_i $(1 \leq i \; \leq M)$ is $\vartheta_i = \rho_i(0), \rho_i, P_i$.A collection of spike variables acquired in neuron ϑ_i, or vector $C_{1,i}, C_{2,i}, \ldots, C_{N_{i,}i} \; \in \; \mathbb{R}^n$, is represented as $\rho_i = \{C_{n,i} \; | 1 \leq n \leq N_i\}$. The collection of starting values in neuron ϑ_i is denoted by $\rho_i(0)$. A collection of production functions linked to ϑ_i are called P_i. There are two sorts of these modes: 1) nonthreshold mode, $f_{t,i}(C_{1,i}, C_{2,i}, \ldots, C_{N_{i,}i})$, and 2) threshold mode, $f'_{t,\mathrm{i}}(C_{1,i}, C_{2,i}, \ldots, c_{N_{i,}i}) \mid Q_t$, where Q_t is the threshold. At stage $t \in \mathbb{N}$, $f_{t,i}$ compute a value $P(s) = f_{t,i}(s)$ and transmits $P(s)$ directly to postsynaptic neurons ϑ_j with $(i,j) \in synapse$. The collection of synapses, represented by the two-dimensional graph structure $synapse \; \subseteq \; \{1,2, \; \ldots,M\}$, is denoted by $synapse$. For every tandem $(i,j) \in synapse$, $1 \leq i,j \; \leq \; M$ and $i \; = \; j$, the input and output neurons are, respectively, $inputN, outputN \subseteq \{1,2, \; \ldots, \; M\}$.

The configuration of the numerical neural-like P system τ at each step t is given by $A_t =< C_{1,i}(t), \ldots, C_{M_{1,}1}(t), \ldots, C_{1,N}(t), \ldots, C_{M_{N,}N}(t) > with \; C_{m,i} \in \mathbb{R} \, for \; 1 \leq i \leq M \; and \; 1 \leq n \leq N_i$, values of variables in all neurons. The process of calculating τ may be represented as a finite sequence, $A_0 \; \Rightarrow \; A_1 \; \Rightarrow \; A_2 \; \Rightarrow \ldots \Rightarrow \; A_k \; where \; k \in \; N$. The numerical neural-like P system τ halts, i.e., the ultimate configuration is attained, if it reaches A_k where there is no function $f_{t,i}$ or $f'_{t,\mathrm{i}}$ modify variables or values.

B. Brain Tumor Segmentation

The dataset $Y = y_1 + y_2 + \cdots + y_n$ presents a binary classification situation, with n being the number of samples in Y. Specifically,$\beth(y)$,which translates $\{y_1 + y_2 + \cdots + y_n\}$ to the two established classes, is the goal function that will be learnt. The three most often used methods for automated classification are k-nearest neighbors, naïve Bayes, and support vector machines.

With differing levels of convolutional and subsampling layers and various implementation modes, several deep networks, such as AlexNet, VGG-16, and ResNet, have recently made significant gains in MRI brain images segmentation.

The convolution, pooling, activation, deconvolution, and loss operations that collectively make up a deep network often possess the following properties.

The convolution operation is defined as follows:

$$B_{j+1} = \beta_{j+1} \; \times \; y_j + \varphi_{j+1} \qquad (2)$$

$$y_{j+1} = f\left(\beta_{j+1}\right), 0 \; \leq j \; \leq n \qquad (3)$$

where$j(0 \leq j \leq n)$ is the layer index, y_j is the jth layer's output (y_0 is the input data), and β_{j+1} and φ_{j+1} are the (j+1)th layer's weight and threshold, respectively. f is an activation function that has several different forms, including the rectified linear unit (ReLu) function and sigmoid function. In order to reduce the spatial size of the maps of features, pooling layers are introduced in after the convolutional layers. Two prevalent approaches for sampling down features by figuring out their optimum and average values are max pooling and average pooling.

Binary cross-entropy loss, that can be formulated as follows in Eq. (4), is always employed for determining a classification model's performance for a binary categorization issue.

$$\beth(q,r,b) = -\frac{1}{n}\sum_i a[\ln p + (1-a)\ln(1-p)] \qquad (4)$$

where q represents the pixels of the images, r denotes the labels, and p is the prediction.

IV. HYPERGRAPH-BASED NUMERICAL NEURAL-LIKE P SYSTEMS (HNN P SYSTEM)

A. Structures of HNN P System Neurons

Conventional numerical neural-like P systems are limited by their two-dimensional simple graph structures, which allow information sharing between neurons limited to those that are connected. Their practical applications are hindered by their non-hierarchical structure and incapacity to store intermediate results temporarily. HNN P systems with hierarchical structures are needed to get around this restriction. Hypergraphs provide a way around this issue by letting P systems function in a higher-dimensional, more comprehensive nonlinear space, facilitating specific transmembrane communications and level-by-level operations while performing calculations in the plane. According to hypergraph theory, items with similar characteristics are members of the same set, whereas objects at different levels are members of supersets. More flexible hierarchical relationships for neurons are provided by these supersets than by conventional graphs or tree-based structures because they contain unique logical structures that can be utilized to organize complex relationships between objects.

A matrix denoting the framework of the hypergraph G, as defined below, is given hypergraph $G = (V, E, W)$.

The hypergraph

$$H(v,e) = \begin{cases} 1 \; if \; v \in e \\ 0 \; if \; v \notin e \end{cases} \qquad (5)$$

The hyperedge $e \in E$ in H can link any number of vertices $v \in V$.

Below definitions outline the five distinct types of neurons that make up the HNN P system.

1. An elementary neuron is defined as neuron ϑ_i $(1 \leq i \leq M)$ in the HNN P system if and only if σi is devoid of neurons.
2. If neuron ϑ_j $(1 \leq j \leq M, i \neq j)$ exists inside/outside of ϑ_i, then ϑ_i is referred to as a parent/child neuron of ϑ_j in the HNN P system.
3. The term hyper child neuron refers to neuron ϑ_i $(1 \leq i \leq M)$only if it has more than one parent neuron. Analogously, for two neurons ϑ_i and ϑ_j $(1 \leq i \neq j \leq M)$, both ϑ_i and ϑ_j are referred to as hyperparent neurons if and only if they share at least one child neuron.
4. Neighboring neurons are those that, for any two child neurons ϑ_i and ϑ_j $(1 \leq i \neq j \leq M)$, share a parent neuron if and only if that parent neuron. Neighbor neurons are another term for any two parent neurons.
5. Input and output neurons are the neurons in the HNN P system that are responsible for variable input and output.

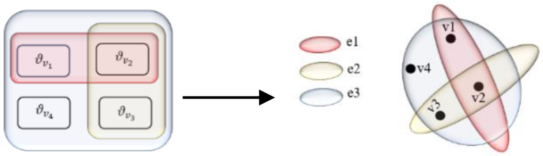


Fig. 1. Example of a hypergraph (a) and it's neuron structure (b)

As an illustration, the HNN P system's basic neuron structure is depicted in Fig. 1. The hypergraph G, as depicted in Fig. 1(a), comprises of V = {$v1$, $v2$, $v3$, $v4$} and E = {$e1$, $e2$, $e3$}, where $e1$ = {$v1$, $v2$}, $e2$ = {$v2$, $v3$}, and $e3$ = {$v1$, $v2$, $v3$, $v4$}. The corresponding neuron structure is displayed in Fig. 1(b). These are the elementary neurons $\vartheta_{v_1}, \vartheta_{v_2}, \vartheta_{v_3}$. ϑ_{v_1} is descended from ϑ_{e_1} and ϑ_{e_3}. ϑ_{v_2} possesses parent neurons ϑ_{e_1}, ϑ_{e_2} and ϑ_{e_3}. ϑ_{v_3} is descended from ϑ_{e_2} and ϑ_{e_3}. As a result, $\vartheta_{v_1}, \vartheta_{v_2}, \vartheta_{v_3}$ are hyperchild neurons. Three neurons are hyperparent: ϑ_{e_1}, ϑ_{e_2} and ϑ_{e_3}.Neighbor neurons include $\vartheta_{v_1}, \vartheta_{v_2}, \vartheta_{v_3}, \vartheta_{v_4}$. Neighboring neurons include ϑ_{e_1}, ϑ_{e_2} and ϑ_{e_3}.

B. Neuronal Transmission within the HNN P System.

There are only two ways that neurons in the cell and neural like P system can communicate: either with one another or with their offspring. Within the HNN P system, connections between (hyper)child, (hyper)parent, or (hyper)parent and (hyper)child neurons can be made hierarchically. That is to say, the HNN P system offers three levels of communication, in contrast to the two-dimensional graph-based NN P system, where communication is limited to synapses between a single type of neuron. Networks between neurons that are (hyper)parents are one example. Amidst (hyper)child neurons is the other. From (hyper)parent to (hyper)child neurons, there is a third one. Thus, the following definitions apply to three new categories of rules. ti $(ti \geq 1)$ indicates how many times a given variable, ρ_i, is controlled by rules in set R. The first square bracket indicates which rules in the second square bracket will cause the neurons to fire. Additionally, there are three scenarios in the second square bracket:

(1) Rules operate on a single neuron, i.e. (6), (13). Variables $\rho_i(t_i - 1)$ evolve into $\rho_i(t_i)$ according to function P_i.

(2) In neurons belonging to the same class, rules operate between them. (7) and (14). The two neurons communicate with one another through the operations $forward$ and

$backward$. The variables $\rho_i(t_i - 1)$ and $\rho_j\big(t_j - 1\big)$are transformed into $\rho_i(t_i)$ and $\rho_j(t_j)$ by the functions P_i and P_j.Exchange of products between ϑ_{v_i} and ϑ_{v_j} occurs simultaneously. Under (7), (14) there is an exception where one of the neurons has no variables; these are rules (8), (9) and (15), (16).

(3) In distinct classes, rules operate between two neurons, i.e. (10). The two neurons communicate with one another through the operations up and $down$. The variables $\rho_v(t_v - 1)$ and $\rho_e(t_e - 1)$ are transformed into $\rho_v(t_v)$ and $\rho_e(t_e)$ by the functions P_e and P_v. Exchanges of products between ϑ_v and ϑ_e take place simultaneously. When one of the neurons in rule (10) has no variables, it is an exceptional case covered by rules (11) and (12).

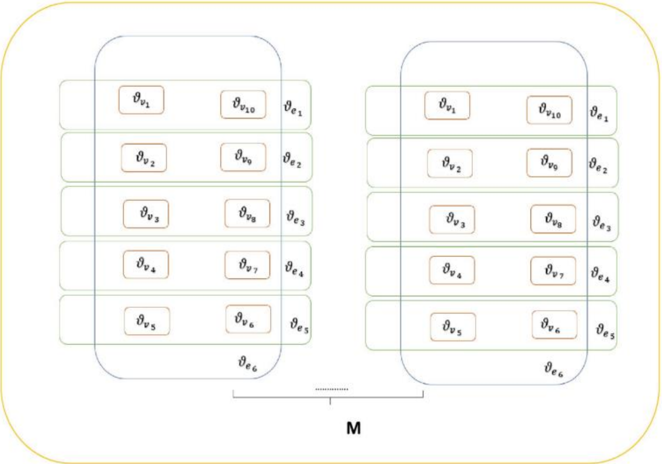


Fig. 2. HNN P system neuron architectures for medical segmentation

Rules for (hyper)child neurons in detail (called V-rules).

$$[\vartheta_{v_i}] : [\ \rho_i(t_i - 1), P_i) \rightarrow \rho_i(t_i)] \qquad (6)$$

Function Pi can be used to compute the variable $\rho_i(t_i - 1)$ for (hyper)child neuron ϑ_{v_i}, resulting in $\rho_i(t_i)$.

$[\vartheta_{v_i}, \vartheta_{v_j}]$:[($\rho_i(t_i - 1)$, P_i , $forward$); ($\rho_j(t_j)$,in)]

→ [($\rho_j\big(t_j - 1\big)$, P_j, $backward$); ($\rho_i(t_i)$,in)] (7)

When (hyper)child neurons ϑ_{v_i} and ϑ_{v_j} contain variables $\rho_i(t_i - 1)$ and $\rho_j\big(t_j - 1\big)$, functions Pi and Pj transform them into $\rho_i(t_i)$ and $\rho_j\big(t_j\big)$. ϑ_{v_i} and ϑ_{v_j} simultaneously exchange their products with each other by operations *forward* and *backward.*

In specific, the V-rule will be framed as follows if ϑ_{v_i} does not include variables and $\rho_i(t_i - 1)$ is present.

$[\vartheta_{v_i}, \vartheta_{v_j}]$:[($\rho_i(t_i - 1)$,P_i, $forward$); (λ,in)]→ [(λ, $backward$); ($\rho_i(t_i)$,in)] (8)

where λ stands for "empty.". Without providing any input, ϑ_{v_i} trades its productions with ϑ_{v_j}.

In the same way, we can derive the V-rule for the case where ϑ_{v_i} contains $\rho_j\big(t_j - 1\big)$ and ϑ_{v_j} contains nothing.

$[\vartheta_{v_i}, \vartheta_{v_j}]$:[($\lambda$, $forward$); ($\rho_j\big(t_j\big)$,in)] →

[($\rho_j\big(t_j - 1\big)$, P_j ,$backward$); (λ, in)] (9)

Using nothing as input, ϑ_{v_j} trades its productions with ϑ_{v_i}.

H-rules: neuronal rules for (hyper)parent and (hyper)child.

$[\vartheta_v, \vartheta_e]$:[($\rho_v(t_v - 1)$, P_v , up); ($\rho_e(t_e)$,in)] → [($\rho_e(t_e - 1)$, P_e ,$down$); ($\rho_v(t_v)in$)] (10)

where the associated functions are P_e and P_v, and the variables are $\rho_v(t_v - 1)$ and $\rho_e(t_e - 1)$ in neurons ϑ_v and

ϑ_e respectively. Brain neuron ϑ_v is the (hyper)child of neuron ϑ_e.

When variables $\rho_v(t_v - 1)$ and $\rho_e(t_e - 1)$ are present in (hyper)child neuron ϑ_v and (hyper)parent neuron ϑ_e, functions Pe and Pv convert them into $\delta v(tv)$ and $\delta e(te)$. After that, via operations up and $down$, ϑ_v and ϑ_e exchange their products with one another simultaneously.

Specifically, the H-rule will be formulated as follows if ϑ_v contains $\rho_v(t_v - 1)$ and ρ_e does not include variables.

$[\vartheta_v, \vartheta_e]$:[($\rho_v(t_v - 1)$) , P_v , up); (λ, in)] →

[(λ,$down$); ($\rho_v(t_v)$,in)] (11)

Likewise, we can derive the H-rule for the case where ϑ_v is empty and ϑ_e is filled with $\rho_e(t_e - 1)$.

$[\vartheta_v, \vartheta_e]$:[($\lambda$, up); ($\rho_e(t_e)$,in)] →

[($\rho_e(t_e - 1)$, P_e , $down$);(λ,in)] (12)

Without providing any input, ϑ_e exchanges its products with ϑ_v.

E rules are a set of guidelines for neurons that display hyperparent behavior.

$$[\vartheta_{e_i}\ ]:[\ \rho_i(t_i - 1), P_i) \rightarrow \rho_i(t_i)] \qquad (13)$$

The variable $\rho_i(t_i - 1)$ may be estimated for (hyper)parent neuron ϑ_{e_i} by function P_i, yielding $\rho_i(t_i)$.

$[\vartheta_{e_i}, \vartheta_{e_j}\ ]$:[$\rho_i(t_i - 1)$, P_i, $forward$); ($\rho_j\big(t_j\big)$,in)] → [($\rho_j\big(t_j - 1\big)$, Pj , $backward$); ($\rho_i(t_i)$,in)] (14)

Functions Pi and Pj convert variables $\rho_i(t_i - 1)$ and $\rho_j\big(t_j - 1\big)$ that are present in (hyper)parent neurons ϑ_{e_i}, and ϑ_{e_j}, into $\rho_i(t_i)$ and $\rho_j\big(t_j\big)$. Exchange of the items between ϑ_{e_i} and $\vartheta_{e_j}$is accomplished by operations *forward* and *backward.*

Analogously, we may derive the E-rule in the case when ϑ_{e_i} contains $\rho_i(t_i - 1)$ and ϑ_{e_j} contains nothing.

$[\vartheta_{e_i}, \vartheta_{e_j}]$:[($\rho_i(t_i - 1), P_i, forward$); ($\lambda$, in)] →
[(λ, $backward$); ($\rho_i(t_i)$,in)] (15)

where λ stands for "empty.". Without providing any input, ϑ_{e_i} trades its productions with ϑ_{e_j}.

Analogously, we may derive the E-rule in the case when ϑ_{e_j} contains $\rho_j\big(t_j - 1\big)$ and ϑ_{e_i} contains nothing.

$[\vartheta_{e_i}, \vartheta_{e_j}]$: [($\lambda$, $forward$); ($\rho_j\big(t_j\big)$,in)] →

[($\rho_j\big(t_j - 1\big)$, Pj, $backward$); (λ, in)] (16)

Furthermore, under the V- and E-rules, if $i = j$, the outcomes of variable ρ_i will be redefined in the neuron ϑ_{v_i}or ϑ_{e_i}, bringing about a change in variable ρ_i.

C. Configurations for the HNN P Systems

The HNN P system τ with $M > 1$ neurons may thus be defined.

$\tau = \vartheta_1, \vartheta_2, \dots, \vartheta_n, R, synapse, inputN, outputN$ (17)

where p neurons in HNN P system τ are represented by ϑ_1, ϑ_2, ... The form of each neuron ϑ_i (where 1≤ i ≤ M) is ϑ_i = (ρ_i, P_i), where δi denotes the set of variables and P_i represents the operations in neuron ϑ_i. The starting value of ϑ_i is $\rho_i(0)$. R refers to the three distinct types of rules in τ -V-, H-, and E-rules that let neurons exchange information with one another and with (hyper)child, (hyper)parent, and (hyper)child neurons. The neurons of τ can be identified by $synapse$. The input and output neurons are presented by $inputN, outputN$ ⊆ {1,2, …, M}, respectively.

V. HNN P SYSTEM TAILORED FOR MRI BRAIN IMAGE SEGMENTATION

One of the primary objectives of this study is to create an HNN P system tailored for MRI brain image segmentation. Drawing inspiration from the commonly utilized down- and up-sampling architecture found in deep learning models, we have devised the neuron structure for our HNN P system (depicted in Figure 3), consisting of nine hyperchild neurons (namely, $\vartheta_{v_1} - \vartheta_{v_9}$)and five hyperparent neurons ($\vartheta_{e_1} - \vartheta_{e_5}$). This configuration is specifically designed to facilitate MRI brain image segmentation tasks, where the goal is to identify and delineate features of tumors or organs present in MRI brain images. To achieve this, we have introduced three distinct types of new rules within the HNN P system, enabling the selection of relevant tumor features from the image data.

A. Initialization

To makes sure the achievement of precise segmentation of tumor, the HNN P system is tasked with both learning distinctive features and categorizing each pixel within a given image slice. To accomplish this, we represent the initial values of the variable δ as a 2-dimensional vector, encompassing the pixels of a slice along with their associated labels, denoted as ρ (C_{11}, C_{12},… , C_{1h}; …, C_{i1}, C_{i2}, … ,C_{ih}; … , C_{g1}, C_{g2}, … , C_{gh}) .Initially, the pixels from the images and the computational parameters are assigned to variable $\delta v1$ (0) within the input neuronϑ_{v_1}, while the label values are assigned to $\delta e5$ (0) within neuron ϑ_{e_5}.Once all images and labels are initialized in this manner, the HNN P system commences its segmentation process

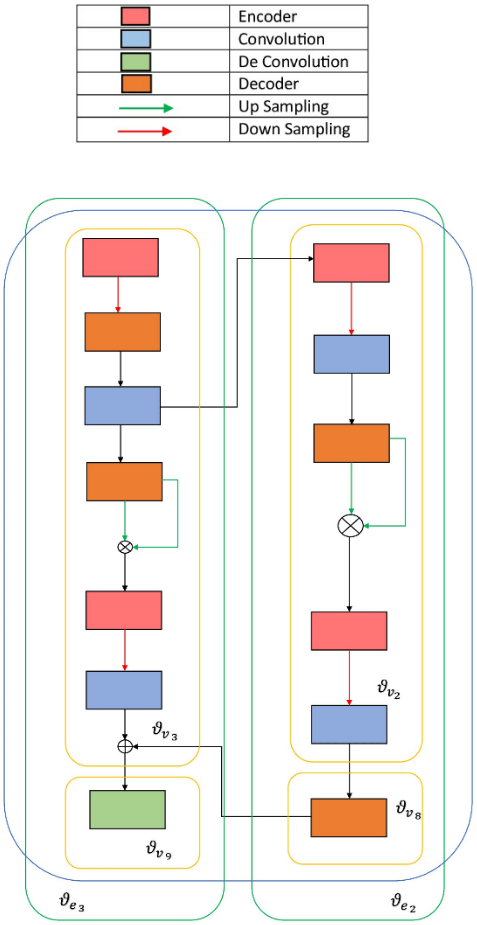


Fig. 3. In the hyperparent neurons ϑ_2 *and* ϑ_3, which make up the partial neuron framework of the multilevel fusion attention mechanism, their objective is to incorporate the up-down features from adjacent neurons in ϑ_2 *and* ϑ_3 with the forward-backward features from neurons in ϑ_2 *and* ϑ_3 , along with their corresponding counterparts in ϑ_8 *and* ϑ_9.

B. Working Mechanism

The neurons responsible for the downsampling and upsampling paths are divided into two groups: $\vartheta_{v_1} - \vartheta_{v_5}$ and $\vartheta_{v_6} - \vartheta_{v_9}$.Images are initialized within neuron ϑ_{v_1}. Subsequently, neurons $\vartheta_{v_1} - \vartheta_{v_5}$ and$\vartheta_{v_6} - \vartheta_{v_9}$perform the downsampling and upsampling operations, respectively, utilizing V-rules inspired by the UNet model.

To enhance both the boundary and content features of tumors, we have devised a multilevel fusion attention mechanism (illustrated in Figure 4) within the hyperparent neurons, specifically $\vartheta_{e_1} - \vartheta_{e_4}$. This mechanism combines the up-down features from adjacent neurons in $\vartheta_{v_1} - \vartheta_{v_5}$and the forward-backward features of neurons in $\vartheta_{v_1} - \vartheta_{v_5}$with their corresponding neurons $\vartheta_{v_6} - \vartheta_{v_9}$ using a combination of V-rules and H-rules.

C. Termination and Output

Upon the completion of calculations by all neurons, including: $\vartheta_{v_1} - \vartheta_{v_9}$and $\vartheta_{e_1} - \vartheta_{e_5}$, a spike denoted as α is generated. The HNN P system undergoes iterative updates until a predefined number of α spikes is reached. Once the HNN P system concludes its operation, the values of variables within neuron ϑ_0 are regarded as the ultimate results extracted from the HNN P system.

VI. EXPERIMENT AND DISCUSSIONS

A. Dataset

"The Multimodal Brain Tumour Image Segmentation Benchmark" is referred to as BRATS. Evaluating objectively the performance of various state-of-the-art brain tumour image segmentation techniques is a challenging task. Furthermore, the implementation of a commonly recognized benchmark for automatic brain tumour segmentation the BRATS benchmark has made it possible to compare different glioma segmentation techniques objectively using this shared dataset. 3064 T1-weighted, contrast-enhanced images of three different types of brain tumors—meningioma (708 slices), glioma (1426 slices), and pituitary tumor (930 slices)—from 233 patients make up this brain tumor dataset.

B. Evaluation Metrics

Dice Coefficient

A measure of similarity between two sets of data, typically sets of pixels, is called the dice coefficient. Typically, it is employed in image processing and machine learning applications to ascertain the degree of correspondence between two images or data sets.

$$\text{Dice coefficient} = 2 \mid U \cap V \mid / (\mid U \mid + \mid V \mid) \qquad (18)$$

where U and V are the two sets of data being compared and |U| and |V| represent the count of elements in each set. This definition of the dice coefficient is the ratio of the intersection of two sets of data to their union.

Precision

An assessment metric called precision counts the frequency of positive predictions the model made that came true. It is computed by dividing the total number of false positives and true positives by the frequency of true positives.

$$\text{Precision} = TP / (FP + TP) \qquad (19)$$

Where, TP is True Positive, FP is False Positive

Recall

The evaluation metric known as recall quantifies the proportion of positive class samples in the dataset that the model correctly identified. True positives are divided by the total of false negatives and true positives to arrive at this calculation. To calculate recall, use the following formula:

$$\text{Recall} = TP / (FN + TP) \qquad (20)$$

Where, TP is True Positive, FP is False Positive, FN is False Negative

F1 Score

A model's precision and recall scores are combined to create the F1 score, an assessment metric. The precision of a model is the percentage of positive predictions that came true. The frequency of positive class samples in the dataset that the model correctly identified is measured by recall. An F1 score reaches its best value at 1 and worst score at 0, and it can be understood as a harmonic mean of the precision and recall. Recall and precision both contribute equally to the F1 score. The F1 score can be calculated as follows:

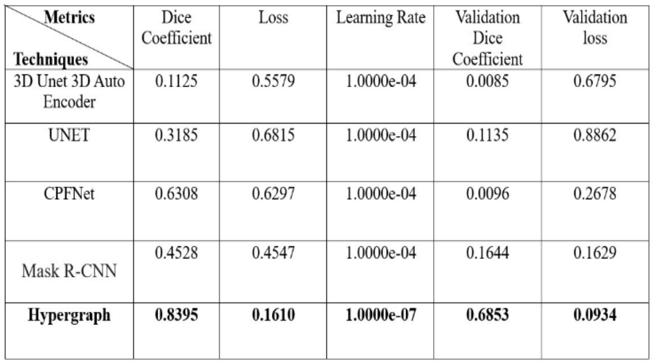
$$F1 = 2 * \text{Precision} + \text{Recall} / (\text{Precision} * \text{Recall}) \; (21)$$

C. Comparisons with Cutting Edge Techniques for the Segmentation of Multiple Brain Metastases

Next, as briefly described below, we analyze the HNN P system with four cutting-edge techniques that P systems have applied on the BRATS datasets.

Mask R-CNN for semantic segmentation and classification. The batch size, iteration stages, and initial learning rate are 16,400 and 0.002.

Table 1 shows the Dice Coefficient, Loss, Learning Rate, Validation Dice Coefficient, Validation Loss of five techniques applied to BRATS dataset. The approaches that are effective are highlighted in bold.



Metrics / **Techniques**	Dice Coefficient	Loss	Learning Rate	Validation Dice Coefficient	Validation loss
3D Unet 3D Auto Encoder	0.1125	0.5579	1.0000e-04	0.0085	0.6795
UNET	0.3185	0.6815	1.0000e-04	0.1135	0.8862
CPFNet	0.6308	0.6297	1.0000e-04	0.0096	0.2678
Mask R-CNN	0.4528	0.4547	1.0000e-04	0.1644	0.1629
Hypergraph	**0.8395**	**0.1610**	**1.0000e-07**	**0.6853**	**0.0934**

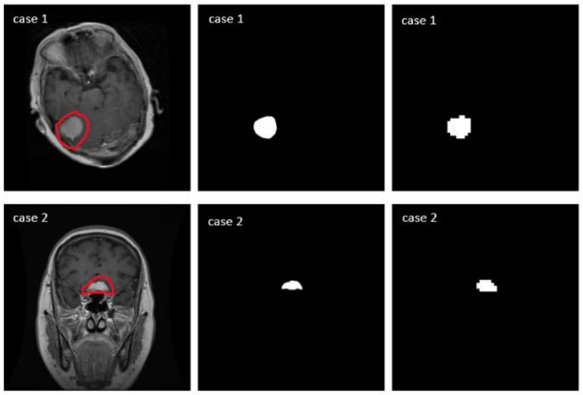


Fig. 4. Outcomes of the segmentation of multiple brain metastases for cases 1 and 2. The raw MRI is displayed in the first column, the manual segmentation is presented in the second column, and the HNN P system prediction is presented in the final column.

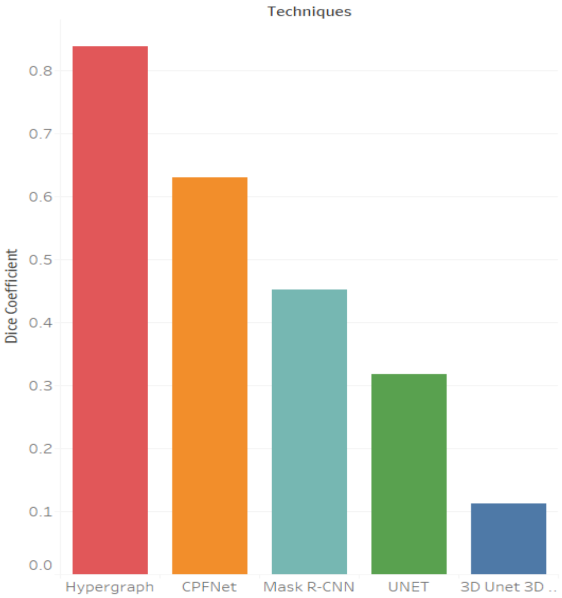


Fig. 5. A quantitative assessment of the performance parameters for each approach is provided by comparing the Dice Coefficient scores obtained from various techniques.

VII. CONCLUSION

In this paper, we have introduced a novel approach called the hypergraph-based numerical HNN P system, which incorporates various types of neurons to capture the complex relationships within neuron structures. This system is specifically designed to address the challenges of Brain tumor segmentation in MRI brain images. By defining communication rules for different types of neurons and leveraging a multilevel fusion attention mechanism, we have developed a segmentation model that enhances both boundary and content features of tumors. Our experiments on Brain Tumor dataset have demonstrated that the HNN P system outperforms existing algorithms by achieving Recall upto 82%.

Fig. 6. A quantitative assessment of the performance parameters for each approach is provided by comparing the Loss scores obtained from various techniques.

REFERENCES

[1] Jie Xue, Liwen Ren, Bosheng Song, Yujie Guo, Jie Lu, Xiyu Liu, Guanzhong Gong, Dengwang Li. Hypergraph-based Numerical Neural-like P Systems for Medical Image Segmentation,2023.

[2] Păun G. Computing with membranes. Journal of Computer and System Sciences, 2000.

[3] Ionescu M, Păun G, and Yokomori T. Spiking neural P systems. Fundamenta informaticae, 2006.

[4] Wu T, Paun A, Zhang Z, et al. Spiking Neural P Systems with Polarizations. IEEE Transactions on Neural Networks and Learning Systems, 2018.

[5] V Asha, S P Sreeja, Binju Saju et al. Brain Tumor Detection using Mask RCNN, 2023 7th International Conference on Computing Methodologies and Communication (ICCMC).

[6] Yashwant Kumar Chandra, Anupam Agrawal.Deep learning based brain tumor segmentation and classification using MRI images,2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT).

[7] Maisha Farzana et al. Semantic Segmentation of Brain Tumor from 3D Structural MRI Using U-Net Autoencoder, 2020.

[8] Shuanglang Feng et al. CPFNet: Context Pyramid Fusion Network for Medical Image Segmentation, 2020.

[9] Arkapravo Chattopadhyay at al. MRI-based brain tumour image detection using CNN based deep learning method, 2020.

[10] Peng H, Li B, Wang J, et al. Spiking neural P systems with inhibitory rules. Knowledge-Based Systems, 2020.

[11] Wu, T, Pan L, Yu ., et al. Numerical Spiking Neural P Systems. IEEE Transactions on Neural Networks and Learning Systems, 2021.