SN Computer Science (2021) 2:363
https://doi.org/10.1007/s42979-021-00756-9

ORIGINAL RESEARCH

Designing Efficient NoC-Based Neural Network Architectures for Identification of Epileptic Seizure

Ayut Ghosh[1] · Arka Prava Roy[1] · Ramapati Patra[1] · Hemanta Kumar Mondal[1]

Received: 20 April 2021 / Accepted: 18 June 2021 / Published online: 30 June 2021

Abstract

Artificial Neural Networks (ANNs) mirror the analytical functions of human neural networks. The performance of smart healthcare systems has been limited to the increasing size and intricacy of information. Several ANN architectures help in the analysis of EEG signals for the identification of epileptic seizures. However, real-time performance needs to be accurate and very quick. Consequently, it is important to design efficient ANN models without compromising the feasibility of hardware realization. Since, CPUs and GPUs are based on conventional bus-system architectures, processing large complex datasets decreases the efficiency, scalability and versatility of the systems. To counter the bottlenecks of the bus-based architectures, Network-on-Chip has been efficient for complex computations. In this paper, we develop NoC-based feed-forward neural network and convolutional neural network models for the identification of epileptic seizure by analysis of continuously monitored EEG signal. The trained neural network models are mapped onto the Network-on-Chip to increase the throughput, power efficiency, parallelism and scalability of the architecture. The performance of all models is thoroughly explored in terms of throughput, energy, latency and identification accuracy of an epileptic seizure.

Keywords Network-on-Chip · Feed-forward neural network · CNN · Epileptic seizure · Classification · Accelerator

Introduction

With rapid advancement in science and technology, large sets of data are being processed. Processing such complex datasets reflects the bottlenecks and inefficiency of present computing systems. A versatile and efficient computing framework is required to process such 'big data' on a real-time platform.

Artificial Neural Networks (ANNs) have led to the advancement by computing large datasets efficiently at a smaller footprint. It has found various real-world applications in speech recognition, image processing, etc. [1]. ANN can be efficiently used in disease classification and pattern recognition by complex non-linear modeling between inputs and outputs [2].

In this paper, we propose in designing an efficient NoC-based ANN platform for classification of epileptic seizures. According to World Health Organization, Epilepsy is an ongoing central nervous system disorder affecting the life of over 50 million individuals around the world [3–7]. It is a fast, capricious, and temporary change in the electrical activity of the brain that influences functions of human beings of all age groups [8–10]. It might be a partial occurrence in the left or right hemisphere of the brain or could affect both of them.

Brain wave patterns can effectively be tracked and recorded with the help of Electroencephalogram (EEG). These EEG records are then examined and analyzed thoroughly by neurologists for detection and then categorization of epilepsy diseases [6]. The EEG assessment is a visual

This article is part of the topical collection "Hardware for AI, Machine Learning and Emerging Electronic Systems" guest edited by Himanshu Thapliyal, Saraju Mohanty and VS Kanchana Bhaaskaran.

✉ Ayut Ghosh
ayg9895@gmail.com

Arka Prava Roy
arkapravaroy98@gmail.com

Ramapati Patra
patraramapati@gmail.com

Hemanta Kumar Mondal
hemanta.mondal@ece.nitdgp.ac.in

[1] Department of Electronics and Communication Engineering, National Institute of Technology, Durgapur, India

cycle and long time is required to inspect and analyze recording of even small time.

These urge the researchers to develop epileptic seizure recognition system based on machine-learning methodologies, utilizing epileptic multi-channel EEG signals including EEG signal procurement, pre-processing, feature extraction and classification [4, 7]. The majority of the proposed frameworks depend on feature extraction process for differentiation of normal and epileptic EEG signals. Performance of such systems is influenced by discriminative feature selection [8]. Deep neural network shows exceptional capabilities of learning on the dataset without any domain information necessary for feature set construction.

Deep learning enables multilayered computational models to learn inherent information with different abstraction levels directly from the available data. Models computing such large datasets would require higher number of nodes, which challenges the hardware implementation of ANN. Hence, the design of an ANN accelerator becomes difficult. Mostly, ANN is simulated over CPUs, GPUs and FPGAs. The traditional architecture of CPUs and GPUs limits the computations of such large datasets. The bottlenecks of the conventional architectures lead to increased power consumption and traffic within the system, which results in an inefficient computation [11]. FPGAs get restricted from using its reconfigurable feature, due to its limited logic and storage resources while mapping large nodes of ANN. Hence, an efficient platform is required to process complex real-time activities.

In this paper, we propose an efficient and low-cost NoC-based ANN platform to classify epileptic seizure using EEG signals. We employ Network-on-Chip (NoC) to improve the computational flexibility. Within heterogenous multi-core systems, Network-on-Chip is proven to be efficient to process complex data, providing higher throughput, scalability and parallelism over any conventional architectures [12, 13]. In this work, we map feed-forward and convolutional neural network over NoC and report in terms of classification accuracy, throughput and latency. The models are first trained and then mapped onto NoC for real-time classification. The convolutional neural network is flattened after training and then mapped onto NoC. We utilize time slicing mechanism and power gating technique within the architecture to reduce complexity and static power dissipation of the architecture, respectively. The nodes of the neural network model are clustered and mapped within the processing elements (PEs) of the NoC architecture, where various computations are performed and outputs are packetized which are stored within temporary memory and are retrieved when required. The Artificial Neural Network (ANN) efficiently processes the large complex sets of data used for classification of epileptic seizure and Network-on-Chip (NoC) provides the versatility and reliability in communicating to various nodes within the architecture.

The salient contributions of this paper are:

- Development of an efficient and low-cost feed-forward and convolutional neural network models for accurate epileptic seizure classification.
- Designing a low-power and efficient Network-on-Chip-based ANN accelerator for processing real-time continuous activities.
- Utilizing time slicing and power gating technique within the architecture to reduce complexity and power consumption of the accelerator.
- Exploring the performance of the accelerator in terms of classification accuracy, throughput, latency and energy.

The rest of the paper is organized as follows—the next section reviews the background and related works on designing neural network accelerators. The third section gives the dataset description for classification of epileptic seizures using EEG signals. In the fourth section, we discuss the proposed NoC-based ANN architecture. The fifth section gives the details of the performance of the architecture explored in terms of classification accuracy, latency, throughput and energy. The last section is the conclusion.

Background and Related Works

Different works that map Artificial Neural Networks (ANNs) on Network-on-Chip Architecture are thoroughly investigated. In the SpiNNaker project [14], each hub is made of 18 ARM9 centers and loads are put away in DDR SDRAM. Aim of this project is parallel simulation of neurons on NoC Architectures. [15] clarifies how an ANN can be emulated on FPGA based on NoC framework. SyNapse [16], a project by IBM for wide range of cognitive and sensory applications, uses ANN having 2D arrangement of neurons.

While EMBRACE [17] has come up with FPGA execution of Spiking Neural Network communication inside a NoC system Application, an NoC-based interconnection of Spiking Neural Network is examined by the DhyANA [18]. An analog neuromorphic computing platform suitable for implementation of any type of neural network is proposed by the FACETS [19]. NoC-based ANN models for specific applications have likewise been proposed, for example, Kakoulli et al. likewise planned an ANN architecture dependent on NoC for hotspot prediction [20] inside the framework to maintain sustainability in performance of NoC architecture; Wang et al. planned an ANN-dependent admission controller [21], capable of predicting the packet injection rate at every hub for productive communication.

There has been no significant work till now that has formulated any NoC-based ANN framework for identification or characterization of any health disfunction. Certain NoC-based ANN [22], CNN [23] and DNN [24] test systems have been proposed for simulating of NoC-based Neural Networks. In this paper, we would examine about the techniques for mapping different types of ANN to the NoC framework, development of the ANN models capable of classification of EEG signals and identification of different stages of epileptic seizure and look into the performance parameters of the proposed architectures.

Dataset Description

The various labels and their significance are described in Table 1. Different types of signals present in the original dataset are depicted in Fig. 1 and the distribution of seizures and other non-seizure signals in the dataset is shown in Fig. 2.

The original dataset [25] consists of 5 different folders, each folder has 100 files, and each folder represents a separate subject/person. Each entry is a count of 23.6 s of brain activity. When sampling the time series, there are 4097 data points. Each of them is a measure of EEG recorded at different points in time. This means that 500 people have 4097 data points, and data points for each person are registered in 23.5 s. These 4097 data points are split and mixed into 23 blocks. Each block contains 178 data points per second,

Table 1 Present classes Of EEG signal

Original class representation	Description
1	Recording of seizure activity
2	Recording of the tumor location
3	Identify tumor location and EEG recording from healthy part of brain
4	Eyes closed during recording
5	Eyes open during recording

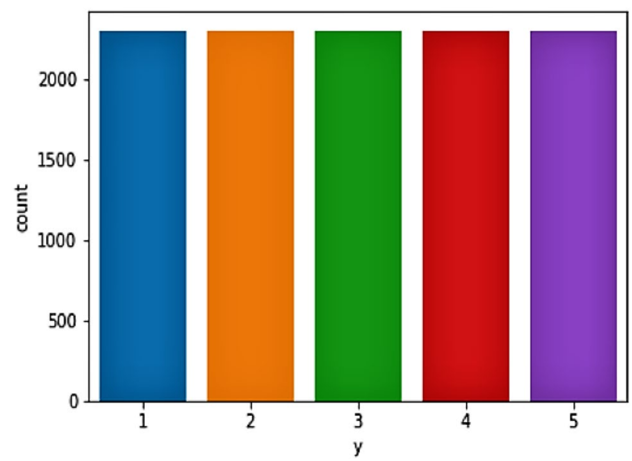

Fig. 2 Distribution of various classes within the dataset

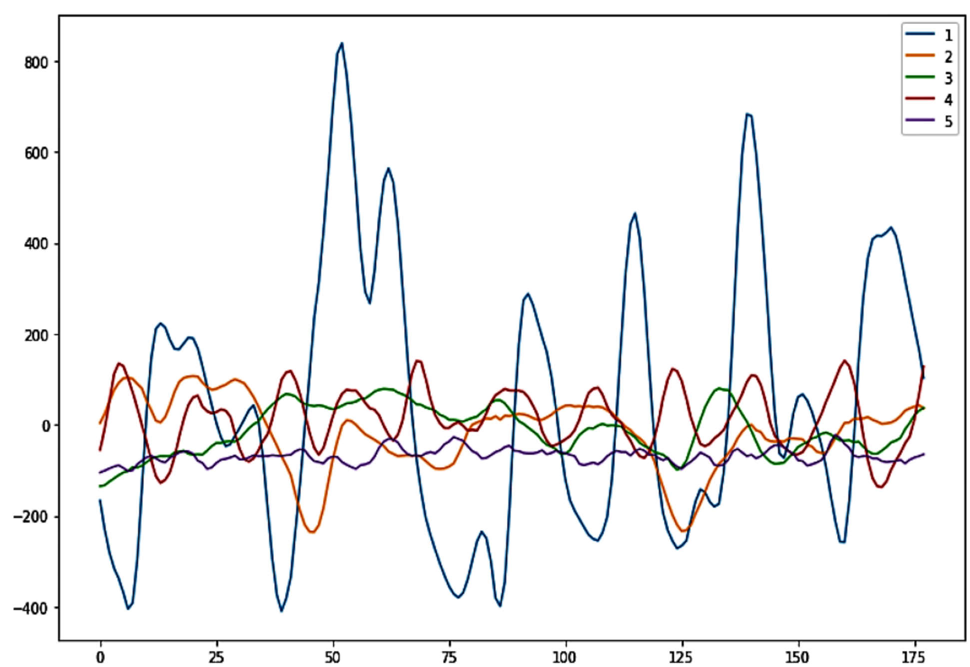

Fig. 1 EEG signal representation for different classes

and each data point is an estimate of the EEG record at a different point in time. This complete operation provides $23 \times 500 = 11{,}500$ information bits (rows), each row contains 178 data points in 1 s (column), and the last column gives the label. The different categories are represented by integers. 1 means epileptic seizure record. All other classes have no seizures. However, our goal is to complete the task of classifying multiple categories so that other health diseases and seizures can be identified.

NoC-Based Neural Network Architecture

In this following section, we describe our proposed NoC framework for ANN computation. An efficient and low-cost architecture is designed for ANN. The framework consists of nodes which act as processing elements and NoC is used to facilitate communication among the nodes within the architecture. We have used a 20 X 20 mesh-based NoC framework for mapping ANN. Figure 3 describes the design methodology for generating NoC-based ANN framework.

The section is divided into various segments: Network-on-Chip Architecture, Computational Algorithms for feed-forward and convolutional neural networks, the time slicing mechanism, clustering and mapping of ANN onto NoC and the routing technique.

A. Network-on-Chip architecture

Network-on-Chip act as a communication infrastructure for the processing elements (PEs) which are interfaced via network interfaces and routers. The router is the key unit within NoC architecture. It consists of five bi-directional ports, where four ports: North, South, East and West ports are used for communicating with the neighboring routers interfacing the neighboring PEs. The fifth local port is used to communicate with the PE. The router contains various components: arbiters, crossbar and virtual channels (VCs). Arbiters solves the issue when multiple input ports demand a same output port for communication [26]. A group of buffers form a virtual channel, which is used to store incoming packets coming as input when the output links are occupied. The arbiters and VCs increase throughput and parallelism within the NoC framework. The crossbar switch connects the input and output ports. The network interface (NI) acts as an interface between PE and router. It also decodes the incoming packets [27].

B. Computational algorithm

Artificial Neural Network has made it possible to develop efficient deep-learning architectures capable of doing complicated tasks such as identification of seizure activity from EE signals accurately. Here, we have deployed a feed-forward neural network model and a 1D-CNN based model. In both the experiments, our focus has been building efficient neural network architectures which at the same time can easily be implemented on NoC.

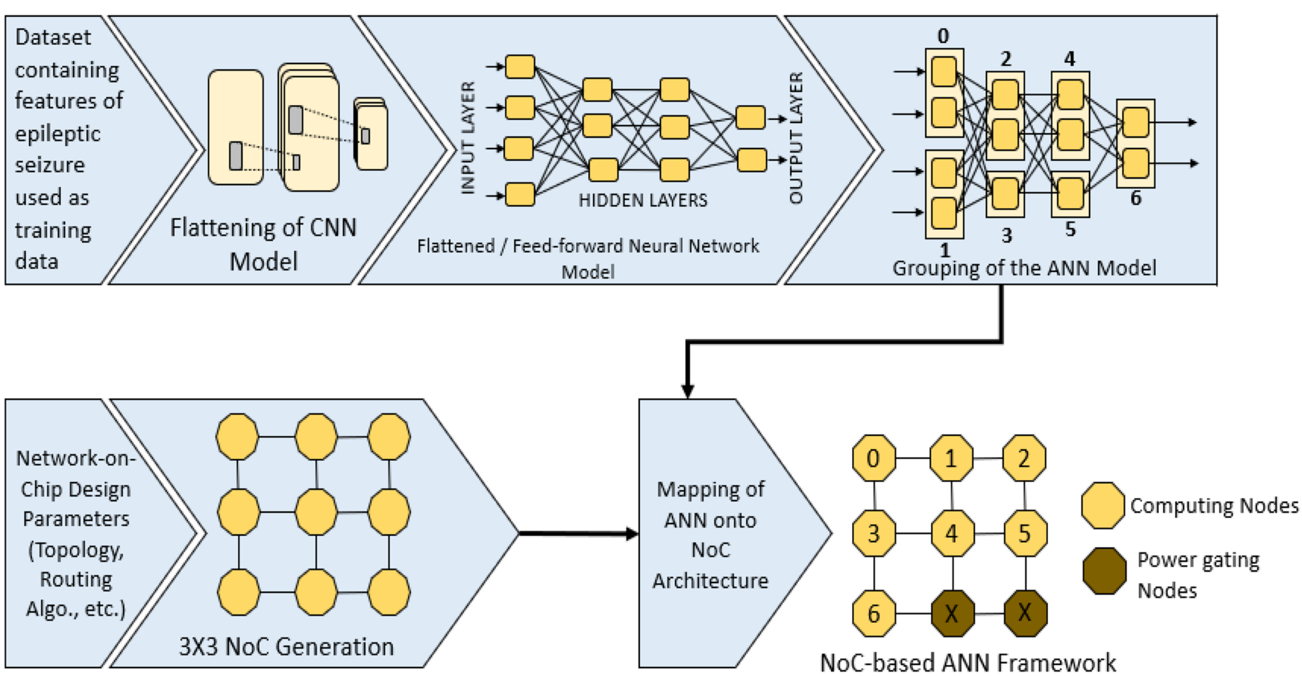

Fig. 3 Design process for NoC-based ANN framework generation

Feed-Forward Neural Network: In a feed-forward neural network, every neuron of a layer is connected to every neuron of its previous and next layer. Multilayer feed-forward neural networks have one input layer, one output layer and one or more hidden layers [28]. This type of networks is trained by back propagation algorithm.

The deployed model is a fully connected feed-forward neural network consisting of the 5 hidden layers. The input layer consists of 256 nodes, followed by a hidden layer having 128 nodes. The detailed architecture is shown in Fig. 4. Input and all hidden layers have ReLU (rectified linear unit) activation function [29]. For 5 class classification purpose, the output layer has 5 nodes with SoftMax activation function. Categorical cross-entropy loss function has been used for calculating loss along with Adam optimizer.

ReLU does not alter the input if it is positive and provides zero output otherwise. This is not only supposed to produce better result but also make computation simple.

Convolutional Neural Network: Since we have a sequence of records as each training sample, a deep 1-dimensional convolutional neural network [30] has been employed for this 5-class classification task. Conventional 2D CNN performs extremely well for images and similar 2D data. They are modified to make 1D CNN which for forward and backward propagation instead of matrix operations, require array operations. Being comparatively shallow and having less computational requirements than 2D CNN, 1D CNN are suitable for real-time and low-cost applications especially handheld devices or mobiles.

For this application, a deep 1D CNN network has been built with the output layer being a dense layer with 5 neurons. It has 6 1D CNN layers including the input layers. The 6th layer is followed by 1D max pooling layer and the output is flattened before passing through output layer. Output dense layer has SoftMax activation function. The loss function used is categorical cross-entropy along with Adam optimizer function. The complete model is depicted in Fig. 5.

C. Time slicing mechanism

To map a larger ANN over NoC for complex computations, either the size of NoC should be enlarged or the processing capability of the architecture should be increased. Enlarging NoC will increase the number of nodes within the architecture and enhancing the processing efficiency will allow processing elements to handle computations of bigger neuron sizes. Both the methods are possible, but not feasible while real-time identification of epileptic seizures.

To address the computational bottleneck for larger ANN, time slicing mechanism is adopted [24]. The mechanism is shown in Fig. 6, where the computations are performed layer-by-layer at various time instants. At an instant, all nodes of a layer or more layers should be mapped over NoC and computed. The output of the nodes is converted to data packets and are stored within temporary memory, which are retrieved at next time instant. The clustering and mapping of the nodes for feed-forward neural network over NoC utilizing the slicing method is depicted in Fig. 7. We use a controller for the mapping and slicing operations within the NoC architecture. The algorithm of the controller is shown in Algorithm 1.

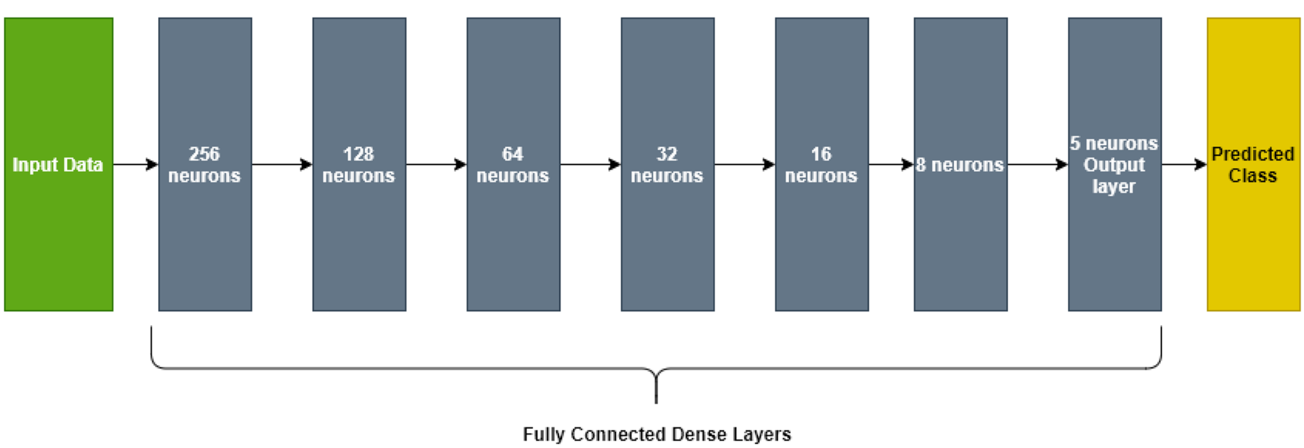

Fig. 4 Feed-forward neural network model

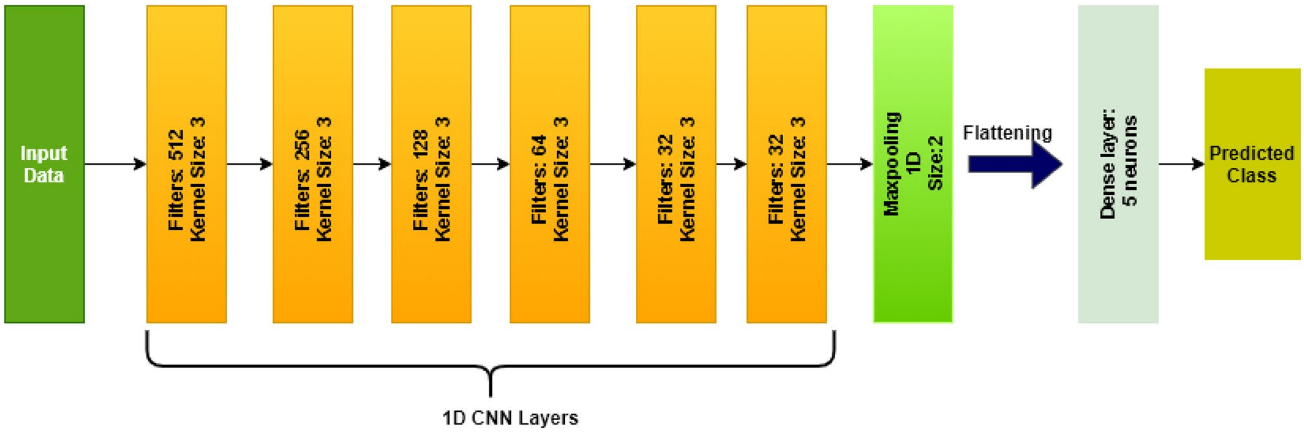

Fig. 5 Convolution neural network model

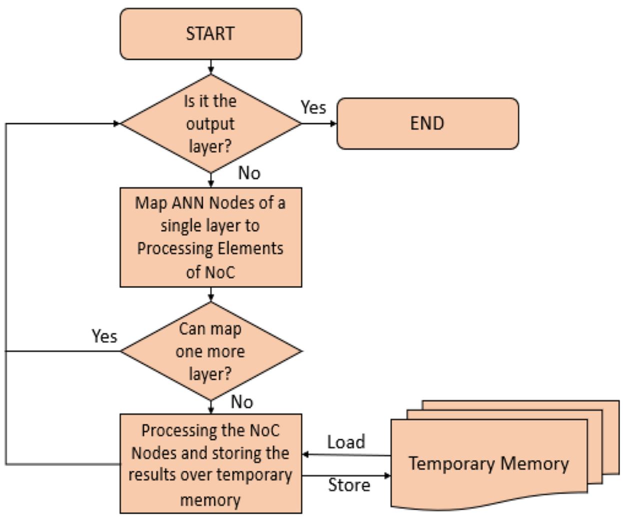

Fig. 6 Flowchart for time slicing mechanism

At any instant of time, the unused nodes of the NoC framework utilize power-gating technique (Fig. 7) to reduce static power consumption [31]. In power gating technique, PMOS switches are utilized within the architecture to switch off the unused routers and PEs to reduce the static power consumption [32]. A gating controller is used to switch off the unused nodes at any slicing instant. As depicted in Fig. 8, it utilizes PMOS transistors in the pull-up part of the network, which is switched on/off by the gating controller. Upon switching off the PMOS transistor, the unused router gets turned off, hence, minimizing the static power consumption. The algorithm of the gating controller is shown in Algorithm 2.

Algorithm 1: Algorithm for the Mapping & Slicing Controller

Initialization: $K \leftarrow$ *Total Nodes* $(M * N)$; $I \leftarrow$ *Time Slice Instants*
Input(s): $L \leftarrow$ Number of Layers, $A^q \leftarrow$ Number of nodes in each layer
do
  if *Last Layer?*
    Map the existing nodes onto NoC
  end
  else
    Map the nodes of the q^{th} Layer onto NoC
    if *can map more layers?*
      continue;
    else Start processing for the *time slice instant I* and *store the outputs* within the *temporary memory*
    end
end

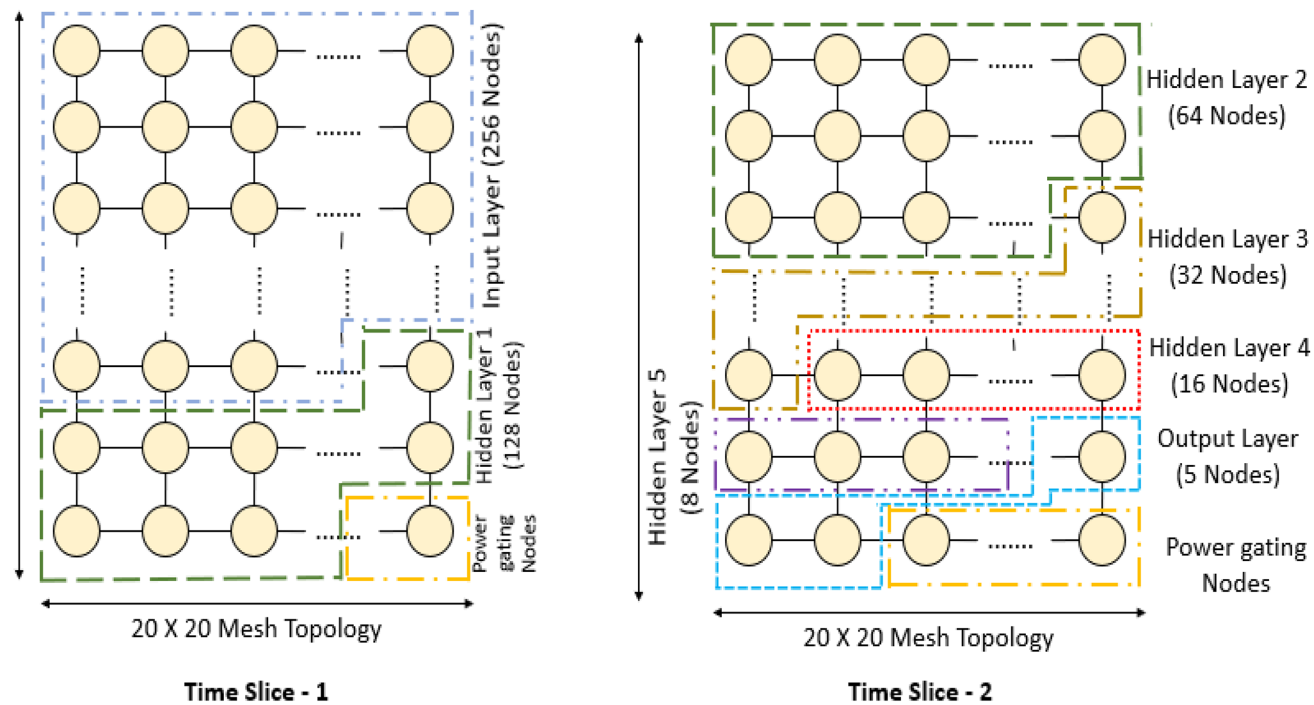

Fig. 7 Time slicing mechanism of feed-forward neural network

D. Clustering and mapping of ANN

For feed-forward neural network, each node of ANN is mapped to a single processing element within NoC framework. At the end of each computation, the outputs of each PE are converted to data packets and further processed. The clustering of nodes depends upon the processing capability of the architecture. As for feed-forward neural network, we have considered two time slices for computation, the traffic is comparatively less than computing with higher number of nodes, thus we have mapped single node to a single PE [33].

Algorithm 2:	Algorithm for Gating Controller
Initialization:	K ← *Total Nodes* (M ∗ N)
Input:	A ← *Number of Active Nodes*
Output:	[K-1 : 0] Z;
while	*At every instant of dynamic slicing* **do**
	if $A \le K$ **then**
	Z [A-1 : 0] ← 0
	Z [K : A] ← 1
	else
	Z [K : 0] ← 1
	end
end	

For convolutional neural network, the network is first flattened using Fig. 9 [24]. After flattening, as it consists of large number of nodes which increases the traffic considerably, hence two nodes are clustered and mapped to a single PE within NoC architecture to reduce the network traffic load.

Depending upon the processing efficiency of the architecture, the slicing, clustering and mapping techniques are carried out to increase the throughput of the system [22, 24, 33]. Further, the throughput can be boosted using a

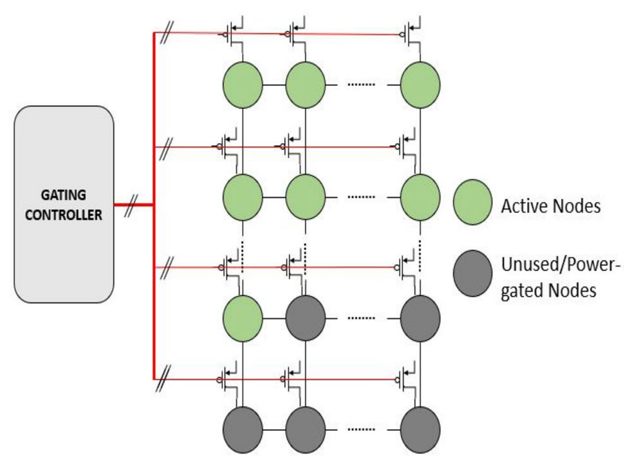

Fig. 8 Implementation of the power gating technique in NoC-based neural network architecture

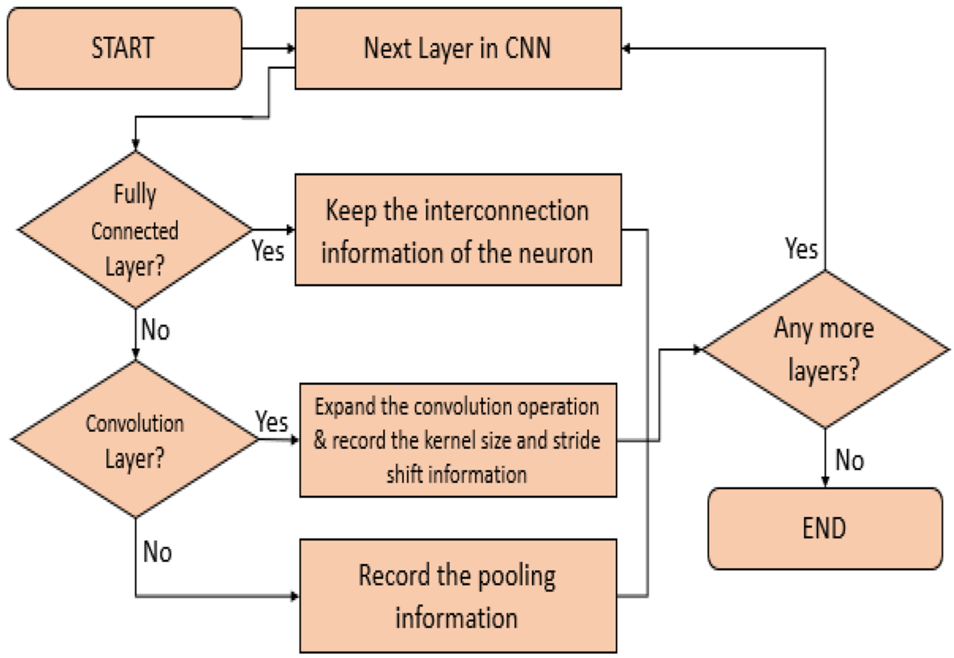

Fig. 9 Flowchart for flattening CNN model

3D-stacked NoC architecture as given in [34], where nodes are connected in a hierarchical fashion [35].

E. Routing algorithm

The feed-forward and flattened convolutional neural networks are mapped onto NoC architecture interconnected in a 2D-mesh topology [35]. Packet-based transmission protocol is adopted for communication. Within the packet-based approach, a packet is composed of three parts: head, tail and the payload. The message's source and destination addresses are constituted within the header. The output of a node is constituted within the payload of the corresponding packet. The end of the packet is designated as the tail.

The proposed mesh NoC-based ANN architecture utilizes XY routing algorithm for communication [36]. In XY algorithm, the packet moves first in X-direction and then in Y-direction to reach to the destination node.

Performance Analysis and Experimented Results

In this section, we describe the simulation setup and discuss the evaluated parameters explored in terms of latency, energy and throughput of the NoC-based ANN framework. The classification accuracy and evaluation metrices of ANN are also discussed.

A. Simulation setup

Python and Python based libraries are utilized for experimental purpose and data processing. Deep-learning model has been developed with Keras and Tensorflow 2.0 [37]. Experimental data have been divided into training and testing data in 80:20 ratio. Training process is carried out for 500 epochs with batch size being 100.

Training dataset is further divided into training and validation data in 80:20 ratio. Since there is imbalance in the dataset the distribution of different classes in the training dataset is calculated and accordingly class weight parameter is provided to address the imbalance. When class weight is assigned, model gives more importance in accurately identifying signal types which are less common. This helps improve the performance of the deployed model.

The NoC communication infrastructure of the proposed architecture was simulated using NOXIM [38] upon Transaction Level Modelling (TLM) [22]. The traffic pattern and the node parameters of the ANN models is mapped with the NoC topology [27]. We have considered a 20 X 20 mesh-based NoC topology for our simulation. One node of feed-forward neural network and two nodes of convolutional neural network were mapped to a single processing element within the architecture to maintain the complexity. The various simulated parameters are described in Table 2.

B. Evaluation metrics

The considered task is 5-class classification. While distributing training, testing and validation data, it may happen that some types of signals are present more and some are present less in the training dataset. Therefore, deployed model is not equally exposed to all types of signals. Considering the difficulty of 5 class classification task, it is evident that model cannot identify all classes of signals with equal efficiency. Performance of models in such cases cannot be judged only based on accuracy. Therefore, some evaluation metrics are calculated to get idea of the performance of the deployed model in identification of individual classes.

Some standard evaluation metrics are utilized for performance evaluation. Among the retrieved instances, the relevant fraction of instances is precision. The fraction which is retrieved among all the relevant instances is recall. F1 score is the harmonic mean of precision and recall.

$$\text{Precision} = \frac{[(\text{Relevant Instances}) \cap (\text{Retrieved Instances})]}{\text{Retrived Instances}}.$$

$$\text{Recall} = \frac{[(\text{Relevant Instances}) \cap (\text{Retrieved Instances})]}{\text{Relevant Instances}}.$$

C. Performance analysis of neural models

During the training process, employed feed-forward neural network model achieves training accuracy of 88.95% and validation accuracy of 87.15%.

During testing, our model is capable of performing classification with 84.22% accuracy. From Table 3, it is evident that our model can identify different types of EEG signals. The class of signal having higher number of samples present in training data is easier to be identified accurately. Performance of the model is equally well in identification of all signal. Evaluation metrics are shown in the table.For 1D CNN model training and validation, accuracy is 91.15% and 89.45%, respectively. Testing accuracy for this model is 87.32% which is better than that of feed-forward neural network model. Values of evaluation metrics for each class are shown in Table 4. It shows overall better performance than feed-forward neural network model in identifying each class of signals. However, the classes which were difficult to identify for feed-forward neural network, have also proven to be challenging to recognize for the 1D CNN model.

Table 2 Simulation setup

Topology	20 X 20 Mesh NoC
Routing algorithm	XY
Cluster size	1 for Feed-Forward NN and 2 for Convolutional NN
Packet size	8 flits
Flit size	16 bits
Clock frequency	2.5 GHz
Buffer depth	4 for each router input
Pipeline depth	4 stages
Workload type	Real

Table 3 Evaluation metrics for feed-forward neural network

Label	Precision	Recall	F1 score
1	0.99	0.64	0.78
2	0.45	0.37	0.41
3	0.30	0.30	0.30
4	0.75	0.56	0.64
5	0.56	0.64	0.60

D. Latency analysis

The delay in transmitting data packets from source node to destination node due to increased traffic is referred as latency. The variation of normalized latency with packet injection rate for time-slice 1 and time-slice 2 is depicted in Figs. 10a and 11a, respectively.

The latency for CNN model is higher than feed-forward neural network model as CNN encompasses higher rate of computation at the nodes of the architecture which increases the traffic. Further, with increase in rate of injection, traffic increases within the system, forcing the latency to increase exponentially [39].

E. System throughput

The efficiency and reliability of the architecture in transmitting data packets is measured by throughput. The variations of normalized global throughput with injection rate for time-slice 1 and time-slice 2 are represented in Figs. 10b and 11b, respectively. Throughput is measured in terms of flits/cycle/IP.

Table 4 Evaluation metrics for convolution neural network

Label	Precision	Recall	F1 score
1	0.95	0.72	0.82
2	0.52	0.52	0.52
3	0.55	0.20	0.30
4	0.68	0.76	0.72
5	0.74	0.74	0.74

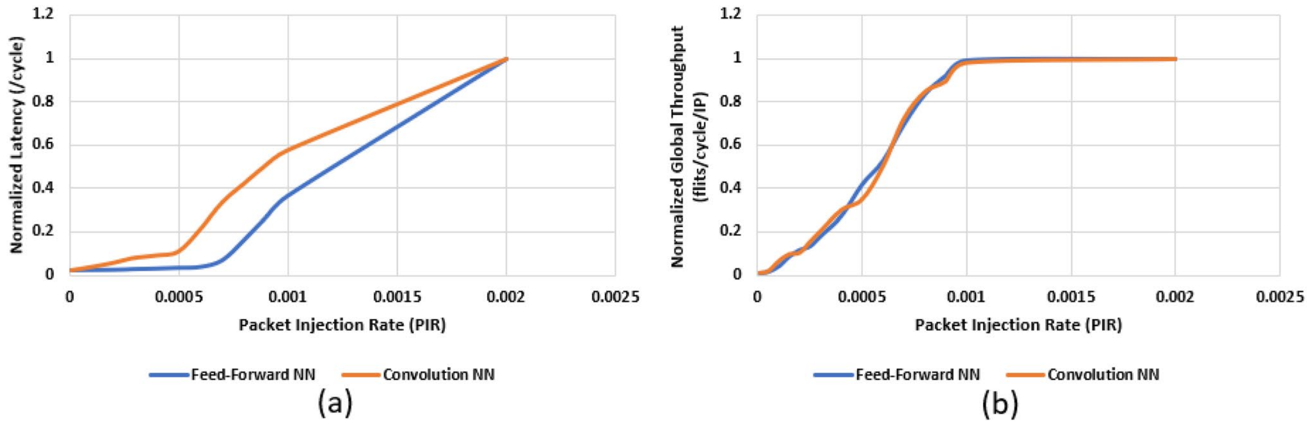

Fig. 10 Variation of latency and throughput for time-slice 1. **a** Depicts the variation of global latency with PIR and **b** depicts the variation of global throughput with PIR

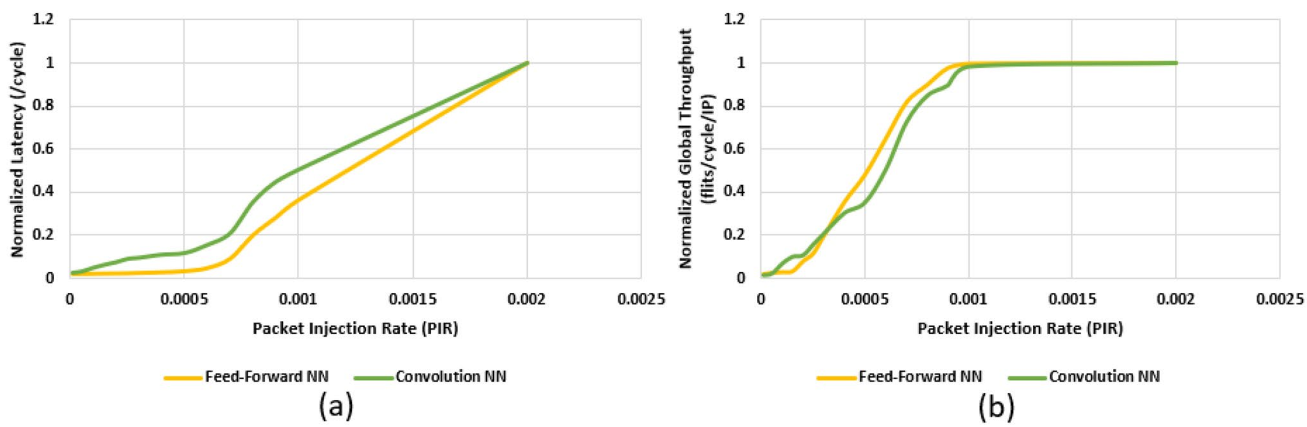

Fig. 11 Variation of latency and throughput for time-slice 2. **a** Depicts the variation of Global Latency with PIR and **b** depicts the variation of global throughput with PIR

The variation of throughput almost remains same for both type of models, indicating efficient and reliable utilization of the architecture. With increase in injection rate, the throughput increases linearly, but after a point it saturates indicating utmost utilization of the architecture [40].

F. Energy variation

With variation in injection rate, the static energy of the architecture remains constant, however, the dynamic energy and total energy varied linearly with the changing PIR. Figure 12 **s**hows the variation of static and dynamic energy of the network for feed-forward and convolutional neural network models. The energy is measured in micro-Joules (μJ).

Upon using power-gating technique, we alleviate 26.24% and 24.47% of static energy for feed-forward neural network and 1D-Convolution Neural Network, respectively, as compared to using the conventional architecture.

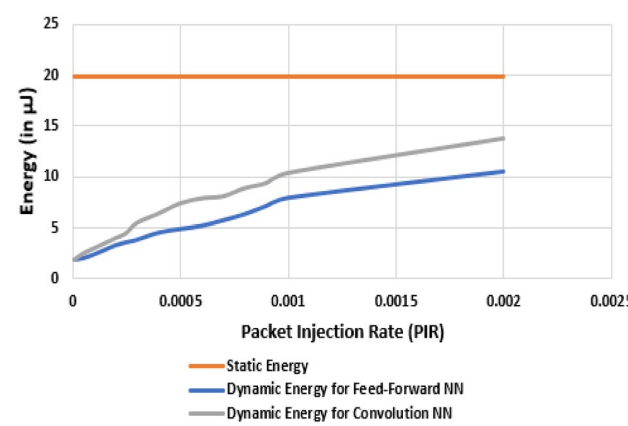

Fig. 12 Variation of static and dynamic energy with PIR

Conclusion

The proposed work utilizes NoC as a communication infrastructure for real-time classification of epileptic seizures. We have used a feed-forward and 1D convolutional neural network models for 5-class classification task. Both the models performed comparably with 1D CNN model slightly produce better results. The neural network models are mapped onto NoC for real-time classification of epileptic seizures. The architecture utilizes various techniques to reduce its complexity and static power dissipation. Further, the use of NoC increases the reliability and versatility of the architecture. The performance parameters for NoC-based neural network models are comparable. Hence, NoC-based CNN models can also be used to process complex data efficiently on a real-time platform.

All the analysis in this paper emphasizes the fact that of NoC has an unparalleled ability to be used as an alternate platform for acceleration of neural network models. Although our models performed well, considering potential application in health-care domain, improvement in classification will allow the real-time platform to classify diseases with greater accuracy. Further, utilizing more techniques and mechanisms in designing the NoC framework will allow more complex models to be mapped for efficient computation. Our work can be considered as an important step towards designing low-cost, reliable and efficient NoC-based neural network accelerators for health-care domain.

Declarations

Conflict of interest On behalf of all the authors, the corresponding author states that there is no conflict of interest.

References

1. Sze V, Chen Y, Yang T, Emer JS. Efficient processing of deep neural networks: a tutorial and survey. Proc IEEE. 2017;105(12):2295–329. https://doi.org/10.1109/JPROC.2017.2761740.
2. Shokry A, Espuña A. The ordinary kriging in multivariate dynamic modelling and multistep-ahead prediction. Comput Aided Chem Eng. 2018;43:265–70.
3. Ullah I, Hussain M, Aboalsamh H. An automated system for epilepsy detection using EEG brain signals based on deep learning approach. Expert Syst Appl. 2018;107:61–71.
4. Usman SM, Usman M, Fong S. Epileptic seizures prediction using machine learning methods. Comput Math Methods Med. 2017;2017:1–10.
5. Hussein R, Palangi H, Ward R, Wang ZJ. Epileptic seizure detection: a deep learning approach. Electr Eng Syst Sci. 2018;16:53.
6. Acharya UR, Oh SL, Hagiwara Y, Tan JH, Adeli H. Deep convolutional neural network for the automated detection and diagnosis of seizure using EEG signals. Comput Biol Med. 2018;100:270–8.
7. Yuan Y, Xun G, Jia K, Zhang A. A multi-view deep learning method for epileptic seizure detection using short-time fourier transform. In: Proceedings of the 8th ACM international conference on bioinformatics, computational biology, and health informatics. 2017; pp. 213–22.
8. Paul Y. Various epileptic seizure detection techniques using biomedical signals: a review. Brain Inf. 2018;5(2):6.
9. AXM Chang, B Martini, E Culurciello. Recurrent neural networks hardware implementation on FPGA. IEEE international symposium on circuits and systems. 2015.
10. Mohammadi M, Pouyan AA, Khan NA, Abolghasemi V. Locally optimized adaptive directional time frequency distributions. Circuits Syst Signal Process. 2018;37(8):3154–74.
11. Wang Q, et al. A statistic approach for power analysis of integrated GPU. Soft Comput. 2019;23(3):827–36.
12. Kim J, et al. Design and analysis of an NoC architecture from performance, reliability and energy perspective. 2005 Symposium on architectures for networking and communications systems (ANCS). IEEE. 2005.
13. "comparison of network-on-chip and busses, White paper, Arteris. 2005
14. Furber S, et al. Overview of the SpiNNaker system architecture. Comput IEEE Trans. 2013;62:2454–67. https://doi.org/10.1109/TC.2012.142.
15. Mand NP, Robino F, Öberg J. Artificial neural network emulation on NOC based multi-core FPGA platform. NORCHIP 2012, Cpenhagen. 2012.
16. Akopyan F, et al. Truenorth: design and tool flow of a 65 mw 1 million neuron programmable neurosynaptic chip. Comput Aided Des Integr Circuits Syst IEEE Trans. 2015;34(10):1537–57.
17. Cawley S, et al. Hardware spiking neural network prototyping and application. Genet Program Evol Mach. 2011;12:257–80.
18. Holanda P et al. DHyANA: A NoC-based neural network hardware architecture. 2016; pp. 177–80.
19. Schemmel J et al. A wafer-scale neuromorphic hardware system for large-scale neural modeling," in Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on, 2010; pp. 1947–50.
20. Kakoulli E, Soteriou V, Theocharides T. An artificial neural network-based hotspot prediction mechanism for NoCs. 2010 IEEE computer society annual symposium on VLSI, Lixouri, Kefalonia. 2010.
21. Wang B, Lu Z, Chen S. ANN based admission control for on-chip networks, 2019 56th ACM/IEEE design automation conference (DAC), Las Vegas, NV, USA. 2019.
22. Chen K, Wang T. NN-Noxim: high-level cycle-accurate NoC-based neural networks simulator. 2018 11th International workshop on network on chip architectures (NoCArc), Fukuoka. 2018.
23. Jimmy Chen K-C, George Wang T-Y, Andrew Yang Y-C. Cycle-accurate NoC-based convolutional neural network simulator. In: Proceedings of the international conference on omni-layer intelligent systems (COINS '19), Association for computing machinery, New York, NY, USA. 2019.
24. Chen KCJ, Ebrahimi M, Wang TY, Yang YC, Liao YH. A NoC-based simulator for design and evaluation of deep neural networks. Microprocessors and Microsystems, 2020.
25. Andrzejak RG, Lehnertz K, Rieke C, Mormann F, David P, Elger CE. Indications of nonlinear deterministic and finite dimensional structures in time series of brain electrical activity: dependence on recording region and brain state. Phys Rev E. 2001;64:061907.
26. Leng X, Xu N, Dong F, Zhou Z. Implementation and simulation of a cluster-based hierarchical noc architecture for multiprocessor soc. In: IEEE international symposium on communications and information technology. ISCIT 2005. 2005; 2: 1203–06.

27. Haowen F, Amar S, De M, Qinru Q. Scalable NoCbased neuromorphic hardware learning and inference. 2018 International joint conference on neural networks (IJCNN), 2018; pp. 1–8.
28. Svozil D, Kvasnicka V, Pospichal J. Introduction to multi-layer feed-forward neural networks. Chemom Intell Lab Syst. 1997;39(1):43–62.
29. Glorot, X, Antoine B, Yoshua B. Deep sparse rectifier neural networks. Proceedings of the fourteenth international conference on artificial intelligence and statistics. 2011.
30. Kiranyaz S, et al. 1D convolutional neural networks and applications: a survey. Mech Syst Signal Process. 2021;151:107398.
31. Lizhong C, Di Z, Pedram M, Pinkston TM. Power punch: towards non-blocking power-gating of NoC routers, high performance computer architecture (HPCA), 2015 IEEE 21st international symposium on.2015; pp. 378–89.
32. Mondal HK, Gade SH, Kishore R, Deb S. Adaptive multi-voltage scaling in wireless NoC for high performance low power applications. 2016 Design, automation and test in Europe conference and exhibition (DATE), Dresden, Germany, 2016; pp 1315–20.
33. Chen K, Ebrahimi M, Wang T, Yang Y. NoC-based DNN accelerator: a future design paradigm. In: Proceedings of the 13th IEEE/ACM International Symposium on Networks-on-Chip, 2019.
34. Jash A, Ghosh A, Noyel N, Patra R, Mondal HK. 3D-NoCNN: NoC based clustered architecture for neural networks. 2020 24th International symposium on VLSI design and test (VDAT), Bhubaneswar, India, 2020.
35. Ghosh A, Jash A, Patra R, Kumar Mondal H. NoCSNN: a scalable interconnect architecture for neuromorphic computing systems. 2020 International symposium on devices, circuits and systems (ISDCS), Howrah, India, 2020; pp. 1–6. https://doi.org/10.1109/ISDCS49393.2020.9263025.
36. Wu R, Wang Y, Zhao D. A low-cost deadlock-free design of minimal-table rerouted XY-routing for irregular wireless NoCs. Networks-on-Chip (NOCS), 2010 Fourth ACM/IEEE International symposium on, Grenoble, 2010; pp. 199–206.
37. Abadi M et al. Tensorflow: large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467; 2016.
38. Vincenzo C, et al. Cycle-accurate network on chip simulation with Noxim. ACM Trans Model Comput Simul. 2016;27(1):1.
39. Mubeen S, Kumar S. Designing efficient source routing for mesh topology network on chip platforms. 2010 13th Euromicro conference on digital system design: architectures, methods and tools, Lille, 2010.
40. Tang M, Lin X. Injection level flow control for networks-on-chip (NoC). J Inf Sci Eng. 2011;27:1.