Minimising Impact of Local Congestion in Networks-on-Chip Performance by Predicting Buffer Utilisation

Aqib Javed*, Jim Harkin, Liam McDaid and Junxiu Liu
School of Computing, Engineering and Intelligent Systems,
Ulster University, Magee Campus, Derry, Northern Ireland, United Kingdom
*Contact: Javed-a@ulster.ac.uk

Abstract—**Networks-on-Chip (NoC) were designed to enhance the communication performance of Multi-processor Systems-on-Chip (MPSoC). NoCs are equipped with buffered input channels which queue incoming data and minimise routing stress especially under uneven traffic distributions. Buffer utilization of a router node provides an early indication to potential local congestion. In this work we propose a novel Spiking Neural Network (SNN) based congestion prediction model to predict input buffer utilization as a congestion parameter to minimize impact of potential local congestion. Router-level and Network-level models are proposed in predicting congestion at each NoC router node. Results show that the router and network models can predict buffer utilization patterns with an average accuracy of 91.89% and 93.76%, respectively.**

Keywords— ***Networks-on-Chip; congestion prediction; Spiking Neural Networks***

I. Introduction

The number of processing cores on a single chip has increased to support parallel multi-processor Systems-on-Chip (SoCs) that can support complex computation and dense communication demands [1]. These SoCs face bandwidth, scalability and latency issues if using existing bus-based interconnect systems [2]. To meet the communication challenges in multi-core systems, scalable network interconnect paradigms have been proposed including communication topology, routing schemes, arbitration, switching, and flow control [3], [4]. Networks-on-Chip (NoC) was designed to address existing scalability and latency issues. Furthermore, it enhances communication bandwidth by providing multiple parallel paths to boost data transmission between processing cores [5].

Congestion is an important factor in NoC performance degradation, and it occurs when large levels of traffic data is routed through specific router nodes [6]. These nodes under high traffic loads start to cause delays in data transmission. NoCs can distribute traffic uniformly across networks to avoid possible congestion [7], and such traffic flow depends on the routing algorithm, application mapping and network topology. These parameters influence traffic flow and cause non-uniform traffic distribution in NoC network. Quality of Service (QoS) is the notable measure to determine network congestion and can be maximized by avoiding uneven traffic load in NoC [8].

Neural networks are inspired from biological neurons to process information and perform human like activities i.e., decision, identification, classification etc. [9]. Over the years, they have emerged as powerful tools for classification and prediction. Spiking Neural Networks (SNN) use a more realistic neural behaviour by incorporating spikes for learning and encoding statistical information. SNNs are comprised of complex mathematical equations to process information in a spatio-temporal domain [10]. Nowadays, SNNs can be implemented on hardware with low area and power overhead [11] enabling new application areas.

NoC traffic patterns are temporal in nature and these temporal patterns can be used by SNNs to identify and predict potential NoC congestions [12]. This work proposes a novel SNN using Spike-Response-Model (SRM) neurons that can predict traffic congestion in NoCs. Work investigates two congestion models based on the level of abstraction: 1) router model and 2) network model. Both models are trained and tested on temporal traffic patterns to predict NoC congestion at each node. Both models can predict congestion 30 clock cycles in advance of it occurring. The predicted output can be used by congestion handling mechanisms or adaptive routing algorithms to minimize network latency by bypassing router nodes with trending congestion hazards. The overall goal of this work is to analyse prediction performance of the proposed SRM based SNN prediction and to identify efficient, low-cost prediction solution for NoC congestion.

Section II reports background on SNN networks and existing NoC congestion solutions. Section III introduces the proposed SNN-based NoC hotspot prediction methodology, and the established experimental setup along with simulation results are discussed in section IV. Section V presents conclusion and outlines future work.

II. Background

This section explains the cause of congestion in NoC and provides brief overview of existing congestion prediction and handling techniques for NoC systems.

A. Literature review

NoCs have become an essential paradigm for Multi-processor System on Chip (MPSoC) in communicating and sharing data between processing nodes [13]. NoCs are widely adopted in MPSoC's to satisfy the need in achieving minimize latency and high bandwidth requirements. NoC performance is influenced by several network parameters i.e. routing algorithm, topology, application mapping etc. [14]. These factors define the overall traffic flow and influences behaviour of NoC traffic patterns. Ideally, the NoC was designed to achieve even traffic distribution across many nodes by transmitting data via multiple paths. But in reality, NoCs face congestion problem due to uneven traffic distribution caused by network parameters [15]. NoC congestion is not an instant phenomenon, it happens in phases before it occupies a whole or part of a network. Fig.1 illustrates different phases of congestion in NoC routers [13]. Congestion occurs inside the router when incoming data packets from different channels are trying to compete to route towards the same output channel (as shown in Fig 1-a). Among all competitors, one input channel will be allocated to route data through the output channel while the rest of the incoming data packets are set to queue in the input channel buffers.

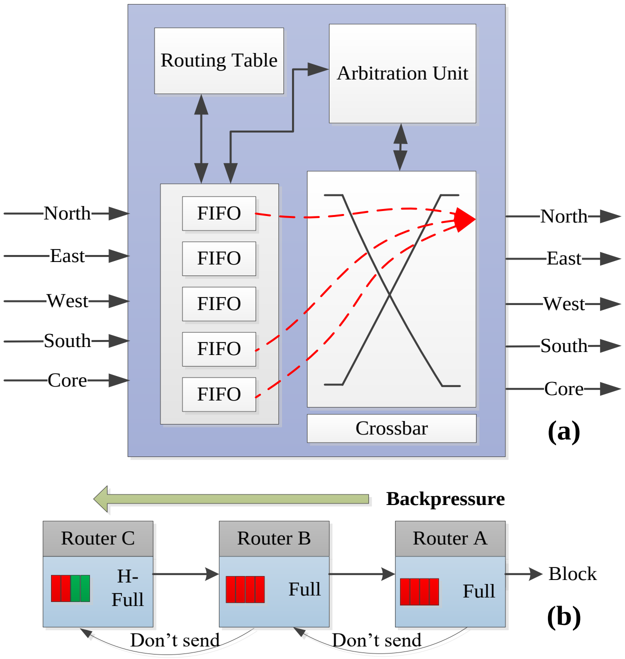


Fig. 1 (a). Switch contention (dashed lines show multiple switch request to the north output). (b). Effect of congestion and Backpressure.

Since routers have limited buffer slots, arrival of many data packets in a short period can cause local congestion. A congested router will stop receiving data packets from neighbouring nodes and start developing back pressure towards adjacent nodes by queuing data at the input buffers of neighbouring nodes. The Backpressure effect is shown in Fig 1(b). These neighbouring nodes stop receiving data from their neighbours and hence causes backpressure towards their adjacent nodes until the whole network becomes congested. Research suggests that congestion handled at local levels improves overall performance of NoC system by minimizing the impact of backpressure [7].

In 2D mesh based NoCs, routing algorithm plays an important role in traffic distribution and transmission across NoC [14]. In recent years, number of routing algorithms are developed e.g. XY, Odd-Even etc. to provide solution for traffic distribution in NoC systems [3]. These routing systems are static in nature and cause flow of traffic towards specific nodes leading to congestion. To overcome non-uniform traffic distribution problems adaptive routing algorithms are proposed. These routing algorithms react on on-path congestion level and change course of incoming data packets through optimal alternative path towards destination node. Dynamic AD (DyAD) [16], Dynamic XY (DyXY) [17] etc. are most commonly used adaptive routing algorithms in NoCs. Adaptive routing algorithms minimize the effect of potential congestion hazard and maximize network throughput. These routing algorithms have low network visibility (having information of adjacent nodes only) and respond to on-path congestion by selecting least congested on-path node as next hop. This adaptive selection often leads to misjudgement problem and routes data towards highly congested node which may cause further congestion [18].

Congestion-aware adaptive routing algorithms are introduced to supress misjudgement problem caused by adaptive routing algorithm. These routing algorithms require additional network information and processes multiple data simultaneously i.e., selection function, Output Buffer Length (OBL), arbitrator usage etc., to find optimal path for routed data packets. Selection function based on OBL uses occupancy information of on-path routers to identify least congested router for incoming data packets [19]. Selection function used Neighbour-on-Path (NoP) algorithm process buffer occupancy level of all neighbouring nodes before forwarding data towards next hop. Some CAAR utilize switch information for adaptive routing decisions. A Path-Congestion Aware Adaptive Routing (PCAR) algorithm requires on-path buffer occupancy information as well as crossbar demand of each output channel to identify optimal on-path router [13]. Similarly, a modified Odd-Even routing algorithm depends on switch contention and the neighbour's occupancy to deflect data packets through the least congested path [20]. A Congestion control scheme utilizes dynamic input arbitration and an adaptive routing path selection is proposed to intelligently balance traffic distribution to enhance NoC performance by 70% with cost of 6% area overhead [21].

In addition, flow control mechanisms, switching and task mapping also contribute in NoC congestion. Mostly NoC are equipped with Warm-hole Flow Control (WFC) mechanism to divide data packets into small chunks called flits [22]. These flits are added with head and tail flits to identify starting and ending of data transmission. Routing function establishes routing channels as soon as it receives header flit. Once the header flit is received, the communication channel starts receiving data packets until tail flit arrived. On arrival of the tail flit, routing function cancels the reservation of a channel and allocates it to the next incoming header flit. The problem arises when these routing flits are stuck in on-path congestion and stop receiving incoming data flits. Thus, keeping the dedicated channel path and host router occupied and busy. This queuing of data flits causes backpressure towards neighbouring nodes which leads to global congestion. Virtual Channels (VC) with additional buffering space can be used to overcome the backpressure effect caused by traffic flow mechanisms [23]. NoC congestion has a devasting effect on network performance and requires timely action on local congestion to avoid its spread across the network [12], [13]. Proximity Congestion Awareness (PCA) technique used switching behaviour of neighbouring nodes (called stress values) to make switching decisions in order to avoid network congestion and increase network load by 20 times [24].

SoCs require task mapping to execute multiple tasks simultaneously and meet real-time design constraints. Applications are mapped on processing elements to execute tasks. Tasks mapped randomly during run-time execution causes communication delays and latency issues. Task allocation controls net traffic flow in NoC interconnects and requires optimization to spread traffic across a network. Congestion-Aware Task Mapping (CATM) are proposed to dynamically allocate tasks on NoC-based MPSoCs to minimize effects of congestion in NoC and maximize network throughput [25]. Congestion-aware dynamic mapping heuristics are proposed to evaluate effects of dynamic task allocation in NoC infrastructure and have shown 78% reduction in NoC congestion. Research investigated the performance of mapping heuristics on NoC-based SoCs with dynamic workloads to identify best task mapping application to minimize NoC congestion. Work showed that path load mapping was effective in NoC congestion reduction and significantly reduced execution time by 19.3 % [26]. Task allocation requires additional time for processing, decision and allocation. Research shows that dynamic task allocation time can be neglected in applications with large execution time but for small execution times, application task mapping adds into latency and throughput issues [15], [27].

All the above techniques are reactive to congestion and activates when congestion has already occurred in the network. These techniques can significantly reduce the impact of congestion but not fully supress the potential hazard of congestion. NoCs require a solution to predict congestion in advance to improve overall system performance. NoC congestion prediction is an on-going research problem and limited work is available to avoid the impact of congestion. The Traffic-Based Routing Algorithm (TBRA) is a hybrid routing algorithm that use switch contention and on-path router occupancy level to switch between Odd-Even and XY routing algorithm to predict and ditch on-path congestion [18]. An efficient runtime Congestion-Aware Scheduling (CWS) based on the link utilizations is proposed to predict traffic pattern in reconfigurable NoC systems[28]. The proposed model shows up to 66% improvement in average network latency and 32% in average throughput. A traffic flow predictor [4] is proposed to control packet injection rate of each node in order to regulate steady number of packets in the network to avoid congestion. Prediction-based Flow Control showed an average 49% reduction in global delay. NoC traffic prediction provides solution for smooth uniform traffic flow across network but also helps to minimize energy consumption by efficient resource allocation. Router with built-in low-power Application Driven Traffic Pattern Table (ATPT) model is proposed to record traffic flow and use historical data to predict incoming data [29]. This predicted data flow is then used to optimize voltage frequency of a router to save up to 86% of dynamic power.

B. *Neural Network and prediction:*

Neural Networks are abstract and simplified mathematical counterparts of biological neurons to perform brain like decision and classification functions. NNs are comprised of neurons (computational nodes) and synapses (communication links) and the Artificial Neural Network (ANN) is most widely used NN model. These neurons are connected in form of layers and transfer information to subsequent layers to generate numeric in the output layer. NNs access information during the learning process and store information in form of weights between neurons. After training of NNs, statistical inputs are then passed through neural layers with stored information to perform prediction, classification and recognition tasks. The values generated by output layer neurons are deemed to classify nonlinear and dynamic behaviours of systems. Spiking neural networks mimics closely to biological neurons and are termed as third generation NN models. SNNs encodes and process information in the form of temporal spikes. Contrary to ANNs, outputs generated by SNNs depend on the time between spikes. Recently, ANNs have been employed to cope with congestion problems in NoC systems. A state-of the art multi-layered ANN based hotspot prediction model was proposed for mesh-based NoCs that uses buffer occupancy level of each network node to predict location of potential congestion [22]. The ANN mode showed an average accuracy of 62-92%, however it lacks scalability for higher NoCs and causes latency issues. Another NN based prediction technique used a hamming network to compute the link buffer utilization in identifying the worst congestion node and re-routing data from that node to minimize congestion hazards [30].

All known pervious neural network approaches used ANNs to predict NoC congestion using different constraints. NoC interconnect generates and transfers information in the form of digital data. NoC traffic patterns are temporal in nature and can be used by SNN for training and testing to predict NoC congestion. A recent study shows SNNs as computationally more reliable and exhibits a low hardware and power requirements [11]. In this work we considered temporal traffic patterns as an input for SRM based SNN model to predict local congestion in NoC system.

C. *SNN prediction model:*

Spiking neurons transmit and process statistical information by a series of firing times called spikes. NoC generates temporal traffic patterns which can be by SNN to analyze system behavior more accurately. This work proposed Spike Response neuron Model SRM with Spikeprop as a learning algorithm to predict NoC congestion.

Spike response model is generalized version of Integrate and Fire (I&F) neuron model, where membrane potential explicitly relay on pre-synaptic and post-synaptic spikes time[31]. The membrane potential ‘u’ of cell ‘i’ at time ‘t’, $u_i(t)$ is defined as:

$$u_i(t) = \eta(t - \dot{t}) + \int_0^\infty \varepsilon(t - \dot{t}, s) I(t - s) ds, \qquad (1)$$

where ε (also called linear filter of membrane) is linear response to input current $I(t - s)$. ‘ṫ’ is the time of last spike of neuron i. The time dependency in membrane potential enables the refractoriness in neuron. Kernel ‘η’ is response of neuron to its own spike.

Bothe’s Spikeprop (aka spike propagation [32]) is an error-backpropagation training algorithm designed for Spiking neurons. Spikeprop is designed to minimize the error between actual firing times t_j^a at output neuron j and desired firing time t_j^d at output neuron j, using the following equation:

$$E = \frac{1}{2} \sum\nolimits_{j \in J} (t_j^a - t_j^d)^2. \qquad (21)$$

III. METHODOLOGY

NoC congestion occurs with the concentration of data traffic at specific nodes. This can be prevented by adopting precautionary measures i.e., uniform traffic distribution across the network and by avoiding potential hotspot nodes. Research shows that local congestion has a devastating effect on NoC performance [7], [29]. Congestion handled locally can avoid its spread across the network through backpressure. It also helps to re-route data through alternative paths thus improving overall NoC latency and throughput performance. This work proposes a congestion prediction model to aid congestion handler or congestion aware adaptive routing algorithms, by bypassing data traffic from potential hotspot nodes. This benefits NoC routers by avoiding the build-up of congested pathways and thus improves overall throughput performance.

NoC routers are equipped with buffers at input channels and these buffers queue incoming traffic to supress effect of local congestion towards neighbouring node. Congested nodes cause backpressure effect once these input buffers channels are completely occupied. Since these buffer utilizations provide early indication for congestion that will affect overall NoC. Therefore, this work proposed the prediction of input buffer utilization to indicate the potential hotspot threat. In NoC interconnect system, each router generates different utilization values. These values depend on behaviour of traffic flow pattern towards routing node defined by routing algorithm and mapped application. Buffer utilization values are temporal in nature and can be used directly by SNN for training and testing of congestion prediction model.

A. Prediction models

We proposed two SRM based congestion prediction models: Router-level model and Network-level model to predict congestion for each NoC node. Both spiking neuron-based models use utilization values from each router and process temporal information to predict local congestion (the prediction is 30 clock cycles in advance). The predictive utilization values are then forwarded to adaptive routing algorithm/ congestion handler to take appropriate routing actions.

i. Router-level Prediction Model.

In router model, every router has its own SNN and input layer of SNN is connected to input channel buffer values (as shown in Fig. 2(a)). Number of nodes in NoC reflects number of required SNNs. These SNNs can read buffer occupancy values directly from input channels to predict router congestion. The size of each SNN depends on router location i.e., inner nodes have 5 input channels and corner nodes have 3-4 channels.

ii. Network-level Prediction Model.

This model proposes one SNN for the whole NoC network. Buffer utilization values from each channel is unified as a single router utilization value. These router utilization values are fed into SRM based congestion prediction model to predict local congestion for each NoC node. Fig. 2(b) illustrate proposed network model.

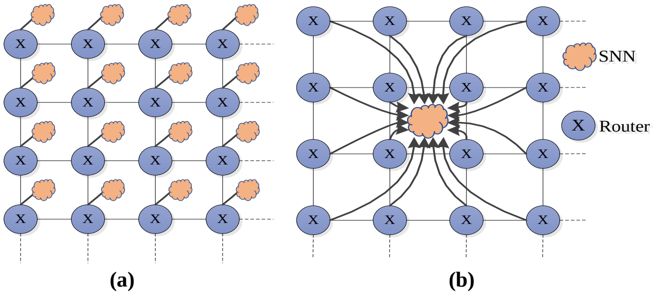


Fig 2. Proposed SNN prediction models (a) Router level and (b) Network level

B. Congestion Criteria:

NoC local congestion can be defined as

"*A router is deemed congested if the accumulated value of buffer occupancy levels is more than 60% of the total buffering slots in one router, and at least one buffer channel is fully occupied."*[12]

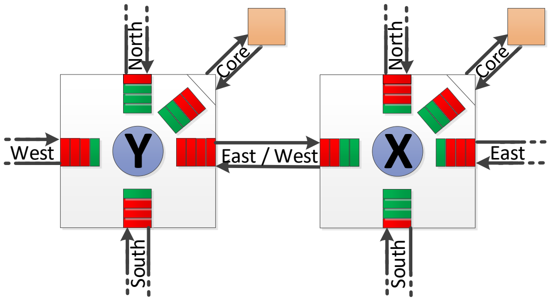


Fig. 3. Buffer Utilization model with 4-buffer slots for each input (Green are free slots; Red are occupied slots)

Consider the example shown in Fig. 3 where router-X and router-Y are receiving routing data from four neighbouring nodes (north, west, south and east) and processing core through 5 input channels. Data routed through router-X and router-Y generates (3, 2, 1, 3, 2) and (1, 3, 3, 4, 2) buffer utilization patterns, respectively. These patterns are directly input to the router-level SNN, whereas in the network-level these utilization values are unified before been forwarding to the network-level SNN. Router-X patterns shows 55% occupancy as compared to 65% buffer occupancy in router-Y. According to the congestion definition, router-Y is labelled as congested and traffic patterns that lead a router into congestion are called '*congestion causing'* patterns.

IV. Experiments

This section describes the experimentation procedure conducted in performing simulation, implementation and analysis of the proposed congestion prediction model.

A. Modelling and Analysis

The experimental setup was established to measure prediction accuracy of proposed SRM based congestion prediction models. Experiment was carried out on synthetic and real-time multimedia application traces. Noxim [33], a cycle accurate NoC simulator, is used to map synthetic and multimedia application on NoC system to generate buffer utilization patterns at each NoC node. Input buffer utilization values depend on mapped application, routing algorithm and Packet Injection Rate (PIR). For evaluation, we used standard XY routing algorithm and set PIR at 0.5 to generate local congestion at each node. Performance is evaluated on six traced based traffic patterns, four synthetic (transpose-1, transpose-2, butterfly and shuffle) and two real-time multimedia (MPEG-4 and MMS) applications. These applications are mapped independently for generation of data traffic across NoC to fetch buffer utilization patterns.

Noxim simulation is executed for 2000 clocks cycles with 1000 warmup clocks. Buffer utilization data values are fetched at each clock to obtain 1000 clocked samples for each network node. These samples are then classified as congested or non-congested according to congestion criteria explained in section III-B. Our research aim to predict local congestion with 30 clocks in advance. SRM based prediction algorithm are modelled and simulated in MATLAB. Utilization dataset generated from NoC simulator are fed into SRM for training and validation.

B. *Performance criteria:*

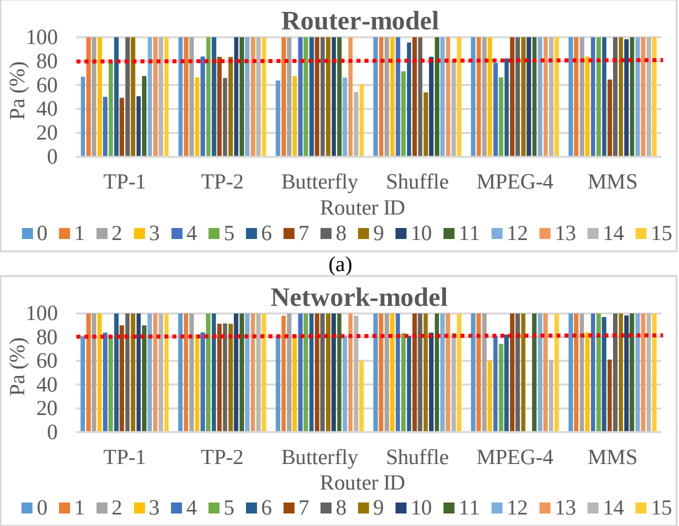
Utilization patterns generated by Noxim are split into two for training and validation. To analyse prediction performance of the proposed prediction model, SRM is fed with 60% of the dataset to train the neural network and validates on 40% of the unseen utilization patterns. Simulation performance of each router is evaluated in the form of prediction accuracy P_a defined by

$$P_a = \frac{(\sum TP + \sum TN)}{\sum(P+N)} \tag{1}$$

where congested patterns are labelled as positive (P) and non-congested patterns are termed as negative (N). TP and TN expresses true prediction of (P) and (N) patterns, respectively. A threshold of 80% prediction accuracy was used as a benchmark in order allow comparison of performances across network nodes node. Overall performance of spiking prediction models is evaluated on average prediction performance.

C. *Simulation results*

Prediction accuracy of proposed prediction models are validated on unseen 40% dataset. Prediction accuracy of each node in network model is shown in Fig. 4. It is depicted that network model outperformed in prediction of local congestion in every synthetic traffic scenario. Specifically, in transposed traffic, the network model predicted congested/non-congested values above the 80% accuracy benchmark at each NoC node. For MPEG-4 traffic application, router-level SRM prediction model predicted congestion above performance benchmark in more network nodes then network model. Whereas in MMS application, both model showed uniform prediction accuracy above performance metrics.



(a)

(b)

Fig 4. Prediction accuracy for each router using (a) Router model and (b) Network model

Table 1 shows average prediction accuracies of router-level and network-level models on synthetic and real-time traffic applications. It is depicted that for transpose-1, transpose-2, bufferfly and shuffle synthetic traced based applications, network model shows an average network prediction of 95.39%, 96.33%, 93.86% and 95.73% as compared to 85.28%, 92.72%, 88.28% and 92.91% router-level SRM based prediction model accuracy. Furthermore, network model has shown more than 90% prediction accuracy across all synthetic applications.

TABLE 1 PREDICTION ACCURACY [%]

	R-Model	N-SRM
Transpose-1	85.28	95.39
Transpose-2	92.72	96.33
Butterfly	88.28	93.86
Shuffle	92.91	95.73
MPEG-4	95.45	84.95
MMS	96.69	96.28
Average	91.89	93.76

Router-level prediction model has proven an effective congestion prevention technique for multimedia applications. It shows 95.45% and 96.69% average prediction accuracy as compared to 84.95% and 96.28% average accuracy in MPEG-4 and MMS applications, respectively.

D. *Comparison with existing neural predicion models*

The proposed router-model and network-model SRM based predictors show 85.28-96.69% and 84.95-96.33% accuracy, respectively. The existing ANN based congestion prediction model [22] showed 62-96% accuracy on synthetic and multimedia application traffics. The proposed SRM based spiking neural prediction model has outperformed existing neural prediction techniques.

V. CONCLUSION

In this work we proposed an SRM based spiking neural to predict local congestion in 2-D NoCs, where two prediction models of router-level and network-level were used to predict congestion at each node. Both were able to predict 30 clocks in advance of any actual congestion occurring. Simulation results shows that network model is more accurate than the router-level model because of its visibility towards varying traffic patterns across all network nodes. Furthermore, the network model exhibits lower area overheads for hardware implementation.

These prediction models can be implemented along with existing congestion handling mechanisms or adaptive routing algorithms. The proposed models provide predictive buffer utilizing values and future work will demonstrate how congestion managing techniques can use this information to take appropriate actions early and prevent congestion occurring.

REFERENCES

[1] M. Amin, M. Shakir, A. Javed, M. Hassan, and S. A. Raza, “Low-cost fault tolerant methodology for real time MPSoC based embedded system,” *Int. J. Reconfigurable Comput.*, vol. 2014, 2014.

[2] J. Liu, J. Harkin, Y. Li, and L. Maguire, “Online traffic-aware fault detection for networks-on-chip,” *J. Parallel Distrib. Comput.*, vol. 74, no. 1, pp. 1984–1993, 2014.

[3] A. Benmessaoud Gabis and M. Koudil, “NoC routing protocols – objective-based classification,” *J. Syst. Archit.*, vol. 66–67, pp. 14–32, 2016.

[4] U. Y. Ogras and R. Marculescu, “Prediction-based flow control for network-on-chip traffic,” *Proc. DAC-44*, pp. 839–844, 2006.

[5] J. Liu, J. Harkin, L. P. Maguire, L. J. McDaid, J. J. Wade, and G. Martin, “Scalable Networks-on-Chip Interconnected Architecture for Astrocyte-Neuron Networks,” *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 63, no. 12, pp. 2290–2303, 2016.

[6] N. Alfaraj, J. Zhang, Y. Xu, and H. J. Chao, “HOPE : Hotspot Congestion Control for Clos Network On Chip,” in *Proceedings of the Fifth ACM/IEEE International Symposium, Pittsburgh, PA*, 2011, no. c, pp. 17–24.

[7] M. Tang, “Analysis on Local Congestion of Network-on-Chip,” no. Iccsee, pp. 2863–2866, 2013.

[8] J. Liu, S. Member, J. Harkin, Y. Li, S. Member, and L. P. Maguire, “Fault-Tolerant Networks-on-Chip Routing With Coarse and Fine-Grained Look-Ahead,” *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 35, no. 2, pp. 260–273, 2016.

[9] J. R. De Oliveira Neto, J. P. C. Cajueiro, and J. Ranhel, “Neural encoding and spike generation for Spiking Neural Networks implemented in FPGA,” *25th Int. Conf. Electron. Commun. Comput. CONIELECOMP 2015*, pp. 55–61, 2015.

[10] A. Mohemmed, S. Schliebs, S. Matsuda, and N. Kasabov, “Method for training a spiking neuron to associate input-output spike trains,” in *IFIP Advances in Information and Communication Technology*, 2011.

[11] J. H. Lee, T. Delbruck, and M. Pfeiffer, “Training Deep Spiking Neural Networks using Backpropagation,” *CoRR*, vol. abs/1, pp. 1–10.

[12] A. Javed, J. Harkin, L. Mcdaid, and J. Liu, “Exploring Spiking Neural Networks for Prediction of Traffic Congestion in Networks-on-Chip,” in *IEEE International Symposium on Circuits and Systems (ISCAS), Seville Spain 2020 (accepted)*, 2020.

[13] E. J. Chang, H. K. Hsin, S. Y. Lin, and A. Y. Wu, “Path-congestion-aware adaptive routing with a contention prediction scheme for network-on-chip systems,” *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 33, no. 1, pp. 113–126, 2014.

[14] M. Yuan, W. Fu, T. Chen, W. Hu, and M. Wu, “CABSR : Congestion Agent Based Source Routing for Network-on-Chip,” in *2014 IEEE Intl Conf on High Performance Computing and Communications, 2014 IEEE 6th Intl Symp on Cyberspace Safety and Security, 2014 IEEE 11th Intl Conf on Embedded Software and Syst (HPCC,CSS,ICESS), Paris*, 2014, pp. 669–676.

[15] E. Carvalho, N. Calazans, and F. Moraes, “Congestion-aware task mapping in NoC-based MPSoCs with dynamic workload,” *Proc. - IEEE Comput. Soc. Annu. Symp. VLSI Emerg. VLSI Technol. Archit.*, no. April, pp. 459–460, 2007.

[16] J. Hu and R. Marculescu, “DYAD - Smart Routing for Networks-on-Chip,” in *DAC 2004, June 7-1 1.2004, San Diego, California, USA*, pp. 260–263.

[17] Ming Li, Qing-An Zeng, and Wen-Ben Jone, “DyXY - a proximity congestion-aware deadlock-free dynamic routing method for network on chip,” *2006 43rd ACM/IEEE Des. Autom. Conf.*, pp. 849–852, 2006.

[18] H. Tseng, R. Wu, W. Chang, Y. Lin, and D. Duh, “An Efficient Traffic-Based Routing Algorithm for 3D Networks-on-Chip,” in *Int’l Conf. Embedded Systems, Cyber-physical Systems, & Applications (ESCS’16)*, 2016, pp. 73–79.

[19] G. Ascia, V. Catania, M. Palesi, I. C. Society, D. Patti, and I. C. Society, “Implementation and Analysis of a New Selection Strategy for Adaptive Routing in Networks-on-Chip,” *IEEE Trans. Comput.*, vol. 57, no. 6, pp. 809–820, 2008.

[20] P. Huang and W. Hwang, “An Adaptive Congestion-Aware Routing Algorithm for Mesh Network- on-Chip Platform.”

[21] C. Wang, W. Hu, and N. Bagherzadeh, “Congestion-Aware Network-on-Chip Router Architecture,” in *Proceedings - 15th CSI International Symposium on Computer Architecture and Digital Systems, CADS 2010*, 2010, pp. 137–144.

[22] E. Kakoulli, V. Soteriou, and T. Theocharides, “Intelligent hotspot prediction for network-on-chip-based multicore systems,” *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 31, no. 3, pp. 418–431, 2012.

[23] W. J. Dally, “Virtual-channel flow control,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 3, no. 2, pp. 194–205.

[24] E. Nilsson, M. Millberg, J. Oberg, and R. Robin, “Load distribution with the Proximity Congestion Awareness in a Network on Chip,” in *Proceedings of the Design,Automation and Test in Europe Conference and Exhibition (DATE’03)*, 2003, pp. 11126–11127.

[25] E. Carvalho and F. Moraes, “Congestion-aware Task Mapping in Heterogeneous MPSoCs,” in *2008 International Symposium on System-on-Chip*, 2008, pp. 1–4.

[26] E. Carvalho, N. Calazans, and M. Fernando, “Heuristics for Dynamic Task Mapping in NoC-based Heterogeneous MPSoCs,” in *18th IEEE/IFIP International Workshop on Rapid System Prototyping (RSP’07)*, 2007, pp. 34–40.

[27] T. Chen, W. Fu, B. Xie, and C. Wang, “Packet triggered prediction based task migration for network-on-chip,” *Microprocess. Microsyst.*, vol. 38, no. 4, pp. 316–324, 2014.

[28] H. Chao, Y. Chen, S. Tung, P. Hsiung, and S. Chen, “Congestion-Aware Scheduling for NoC-based Reconfigurable Systems,” in *15th Design, Automation and Test in Europe Conference and Exhibition, DATE 2012 - Dresden, Germany*, 2012, no. March, pp. 1561–1566.

[29] Y. S. C. Huang, K. C. K. Chou, and C. T. King, “Application-driven end-to-end traffic predictions for low power NoC design,” *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 21, no. 2, pp. 229–238, 2013.

[30] H. Cai, Y. Yang, F. Qu, J. Wu, and B. Wang, “Congestion Prediction Algorithm for Network on Chip,” vol. 11, no. 12, pp. 7392–7398, 2013.

[31] R. Jolivet, T. J. Lewis, W. Gerstner, R. Jolivet, T. J. Lewis, and W. Gerstner, “Generalized Integrate-and-Fire Models of Neuronal Activity Approximate Spike Trains of a Detailed Model to a High Degree of Accuracy,” *J. Neurophysiol.*, no. 92, pp. 959–976, 2004.

[32] S. M. Bohte, J. N. Kok, and H. La Poutre, “SpikeProp : Backpropagation for Networks of Spiking Neurons Error-Backpropagation in a Network of Spik- ing Neurons,” *Esann*, no. May, pp. 419–424, 2000.

[33] V. Catania, A. Mineo, S. Monteleone, M. Palesi, and D. Patti, “Noxim : An Open , Extensible and Cycle-accurate Network on Chip Simulator,” *2015 IEEE 26th Int. Conf. Appl. Syst. Archit. Process.*, pp. 162–163, 2015.