Politecnico di Torino
Dipartimento di Automatica e Informatica
DAUIN

PhD in Computer and Control Engineering
34 cycle

Supervisor
Ernesto Sanchez

Artificial Neural Networks Reliability

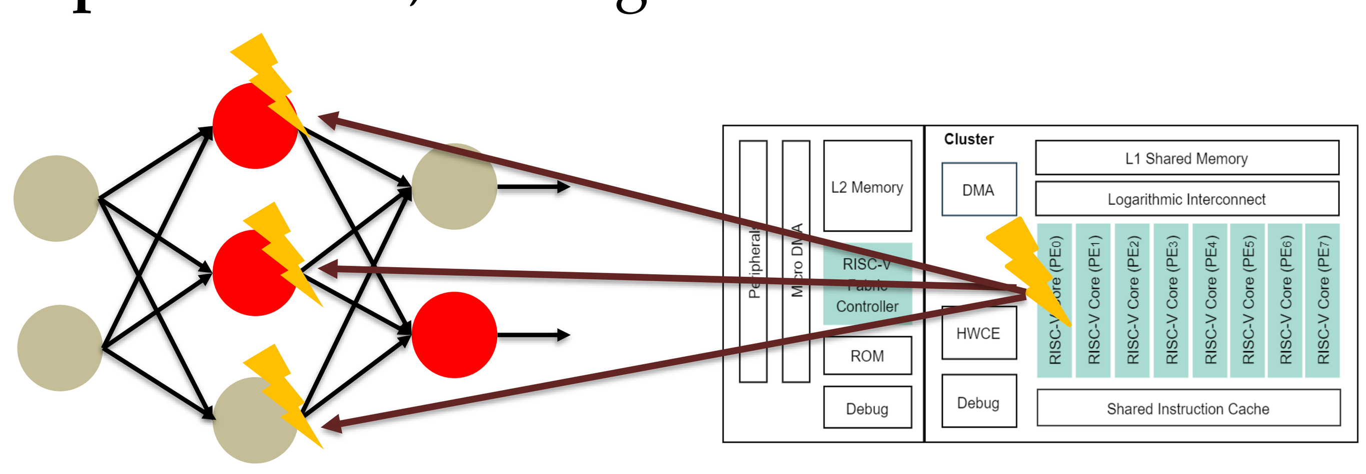
PhD Candidate: *Annachiara Ruospo*

1. Introduction

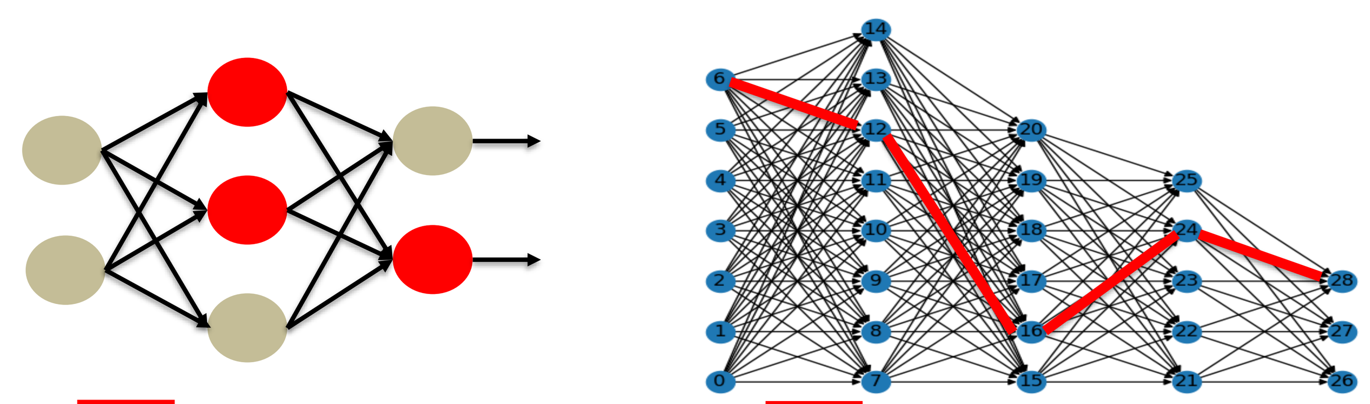
Nowadays, the usage of electronic devices running applications based on Artificial Neural Networks (ANNs) is spreading in our everyday life. To use them safely in human contexts, there is a compelling need for assessing their reliability.

2. Motivation

Artificial Neural Networks are often considered intrinsically robust for being brain-inspired and redundant computing models. However, when they are deployed on resource-constrained hardware devices, **single physical faults** might jeopardize the activity of **multiple neurons**, leading to unwanted outcomes.



Moreover, in the literature it is claimed that neurons exhibit different fault tolerance and resilience levels.



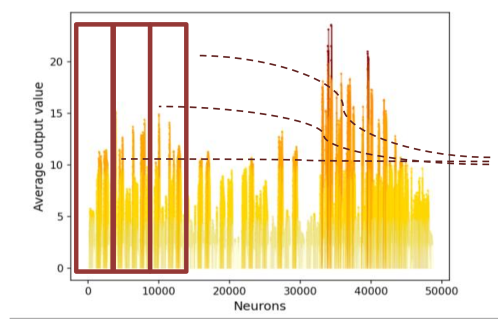
3. Principal Contributions

1. Methodology to identify the most **critical neurons** of a neural network by assigning resilience values to each of them.
2. Reliability-oriented Integer Linear Programming (ILP)-based methodology to **uniformly distribute critical neurons** among the available Processing Elements (PEs) of a MPSoC.

3. Method

The method bases on two levels of analysis: first, the neuron is viewed as an element of each output class (**class-oriented analysis**); second, the same is interpreted as belonging to the entire neural network (**network-oriented analysis**).

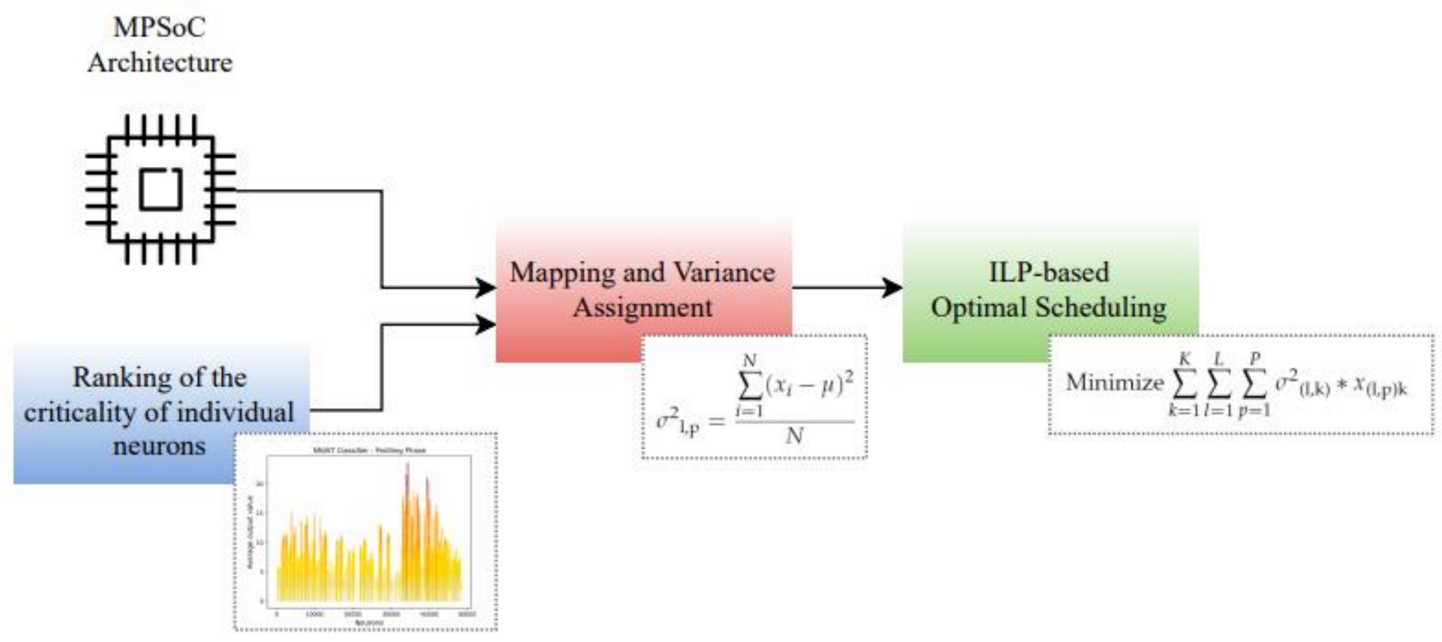
Based on this and on the available PEs of the target AI-oriented MPSoC, a value is given to each chunk of neurons assigned to a single PE.



$$\sigma^2_{l,p} = \frac{\sum_{i=1}^{N}(n_i-\mu)^2}{N}$$

Variance Metric to measure the criticality of group of neurons.

Then, the approach exploits an **integer linear programming** solver to find the optimal and deterministic solution to map ANNs elaborations onto the target hardware architecture.



4. Results

The proposed ILP-based scheduling is able to reduce by 24.74% the neural network wrong predictions (SDC-1%). Overall, it is able to reduce the risk of misbehaviors, producing evidence of faults in the output vector (MSE > 0) but keeping the prediction correct. It leads to a 0.6% *increase* in memory occupation and an *increase* in simulation times of 3.2% at run-time for a single inference cycle.

Fault Injection Results	Static Scheduling		Proposed Scheduling		[%] Variation
	Images	[%]	Images	[%]	
SDC-1	1338	1.63	1007	1.23	−24.74
Hang	71,840	87.61	65,040	79.32	−9.47
Masked, MSE > 0	4910	5.99	9712	11.84	+97.80
Masked, MSE = 0	3912	4.77	6241	7.61	+59.53
Total	82,000	100	82,000	100	

4. Conclusions

This work provides a methodology to improve the reliability of a neural computing system running in a multi-core device. In the future, we will exploit deeper ANNs and more complex datasets, moving the target to GPUs and high-performance architectures.

References

[1] Misra, J.; Saha, I. Artificial neural networks in hardware: A survey of two decades of progress. Neurocomputing 2010, 74, 239–255.

[2] Zhang, J.J.; Gu, T.; Basu, K.; Garg, S. Analyzing and mitigating the impact of permanent faults on a systolic array based neural network accelerator. In Proceedings of the 2018 IEEE 36th VLSI Test Symposium (VTS), San Francisco, CA, USA, 22–25 April 2018