NanoCAD UCLA

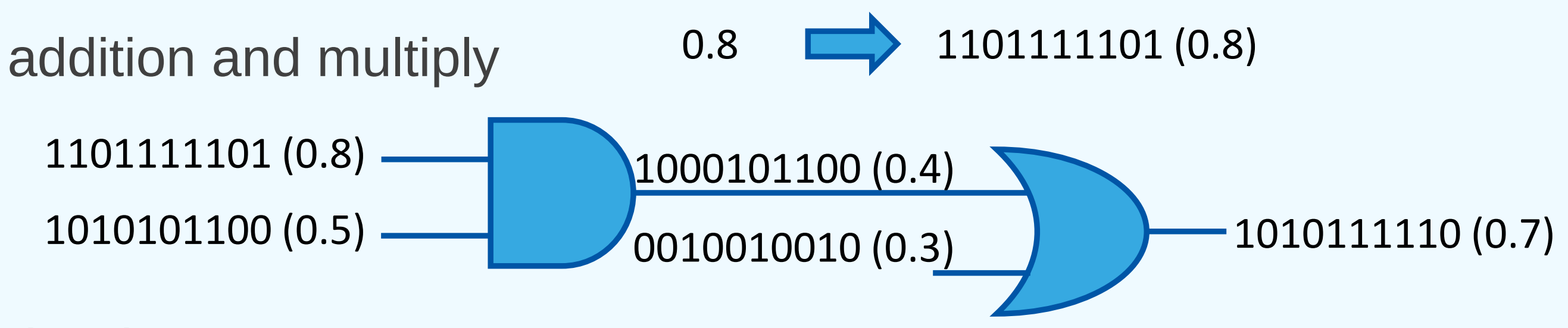
Software Simulation of Stochastic Computing Machine Learning Accelerators

Tristan Melton, Tianmu Li, Wojciech Romaszkan and Puneet Gupta

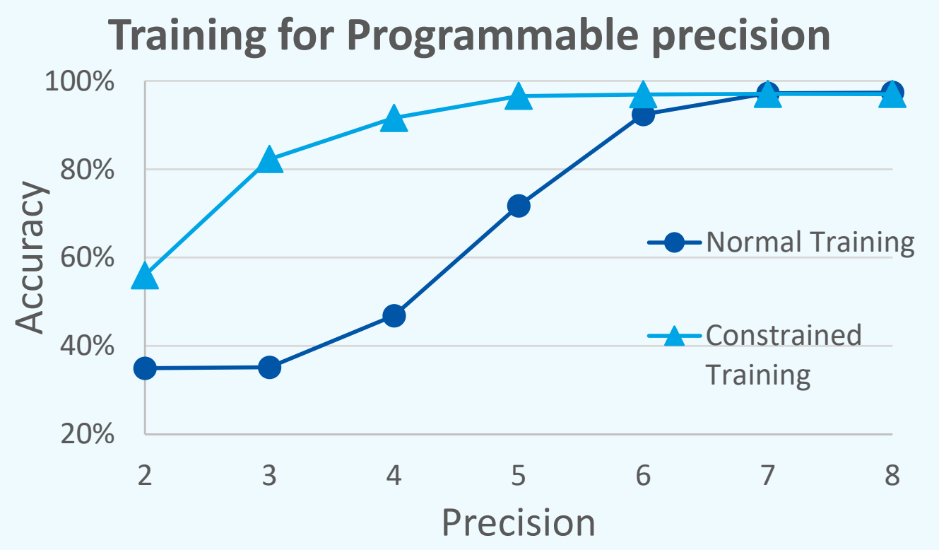
Department of Electrical and Computer Engineering, University of California, Los Angeles

Stochastic Computing

- Represent numbers using proportion of 1’s in a randomly generated bit stream
- Single gate for computation of addition and multiply
  - **Massive parallelism**

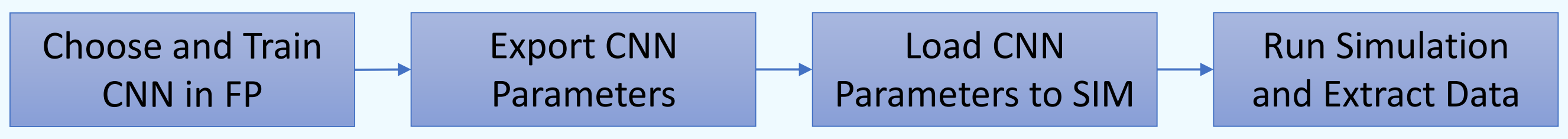


- Variable precision in the same hardware
- Single error only introduces $\pm\frac{1}{N}$ error



The Case for Stochastic Simulation

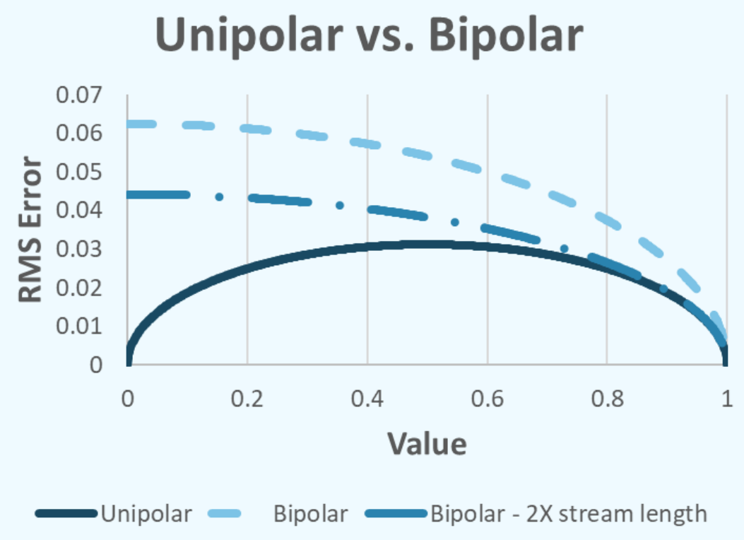
- Training networks to account for SC is time-consuming
- Difficult to integrate SC into existing ML architectures
- Utilizing a separate simulator to test SC properties acts much more efficiently

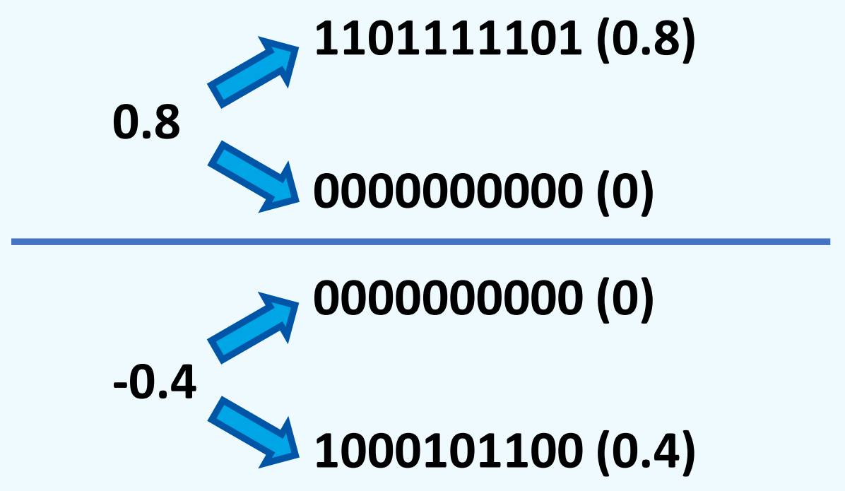


Choose and Train CNN in FP → Export CNN Parameters → Load CNN Parameters to SIM → Run Simulation and Extract Data

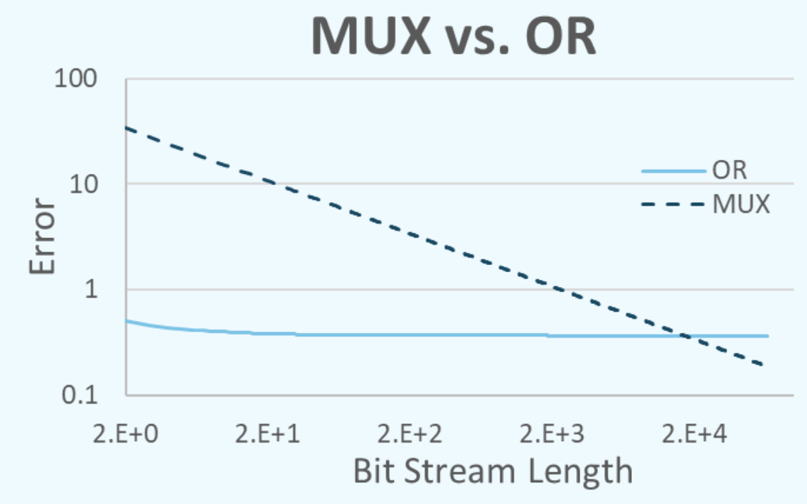
SC and Simulation Optimizations

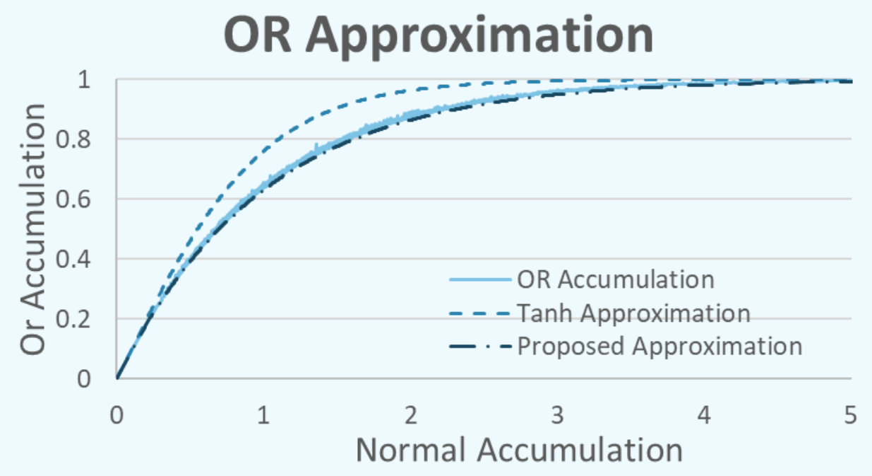
- **Precision**
  - Bipolar representation has low precision.
  - Unipolar representation is limited to [0,1].
  - **Split-unipolar enables high accuracy and negative weights.**



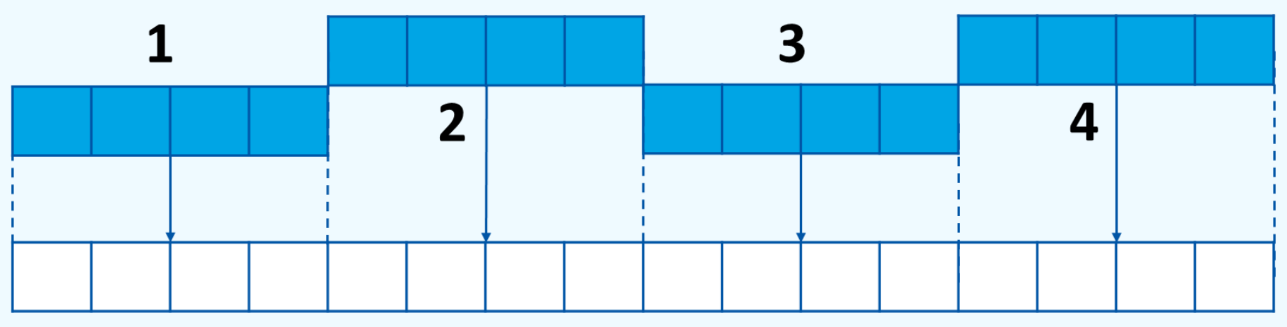


- **Accumulation**
  - Stochastic addition scales down output, degrading precision.





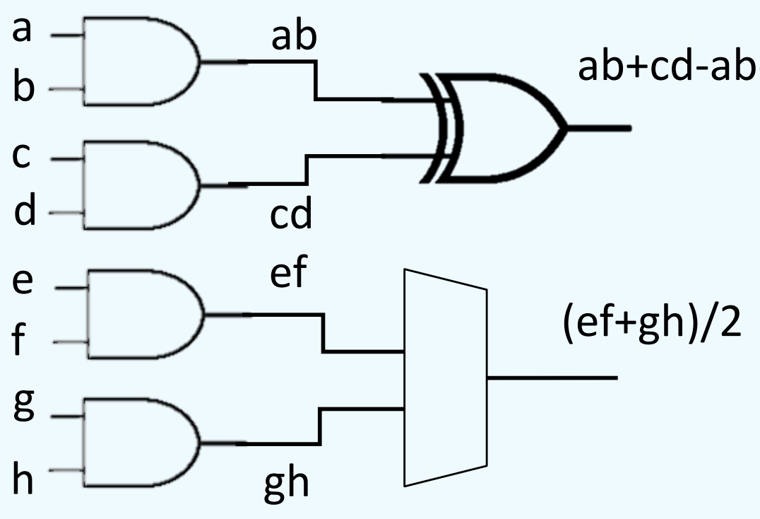
  - Use **OR gate** for scaling-free accumulation.
  - Novel approximation as **activation function.**
- **Other operations**
  - **Max pooling** is expensive in SC.
  - Use **average pooling**. Enables **computation skipping.**



- **SC Generation:**
  - C++ rand() function too slow, caused bottleneck
  - Utilizing an **xor-shift RNG** dramatically reduces generation time.

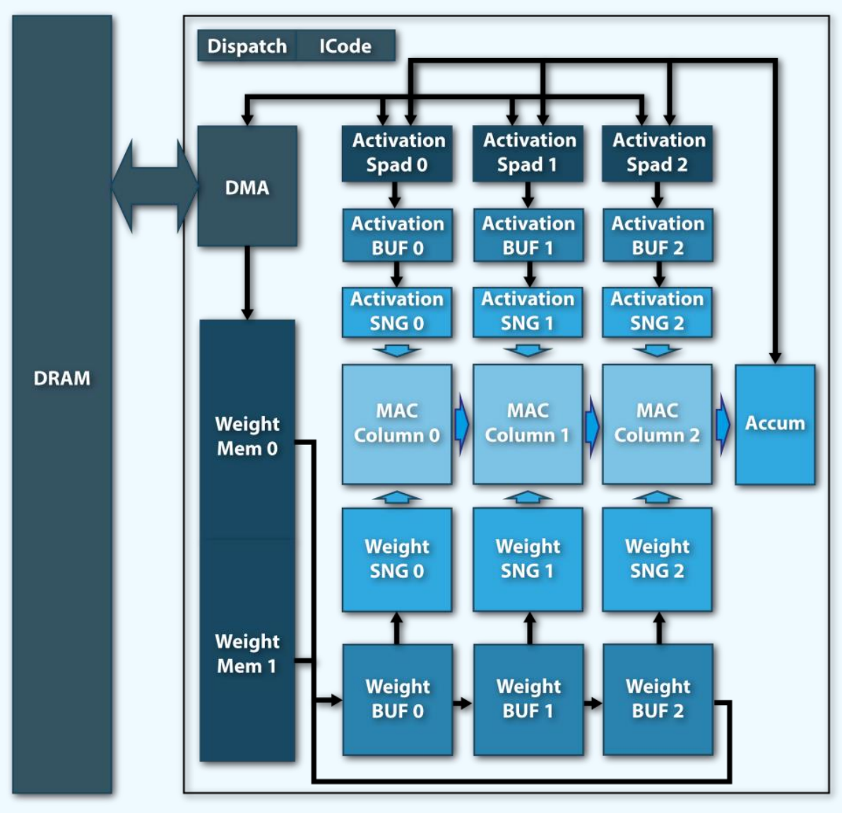
Simulator Design

- SC Simulator developed to mimic computations done in architecture design (ACOUSTIC)
- Provides empirical evidence for this use case of SC



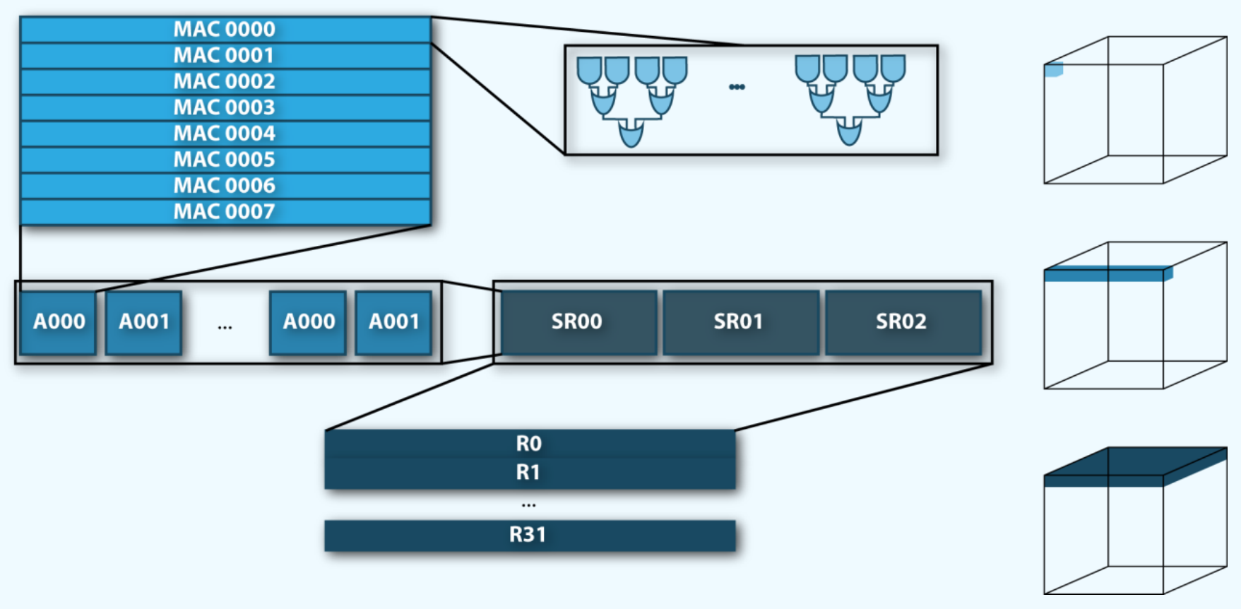
SC Simulation

- Supports OR accumulation, split unipolar representation and stochastic pooling.
- **Configurable layers** allows for testing of various stochastic properties
- Optional tool to measure switching of bit stream to **estimate this source of energy usage**

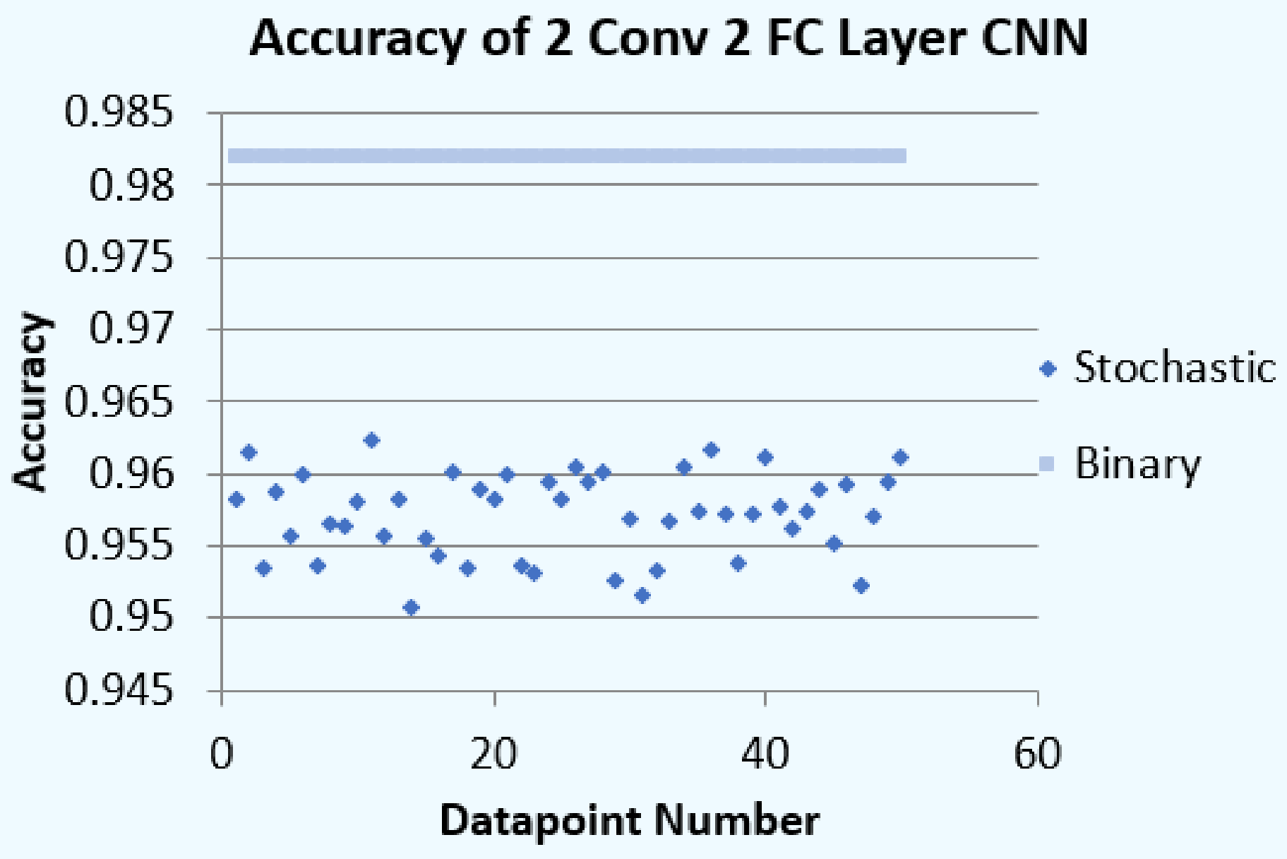


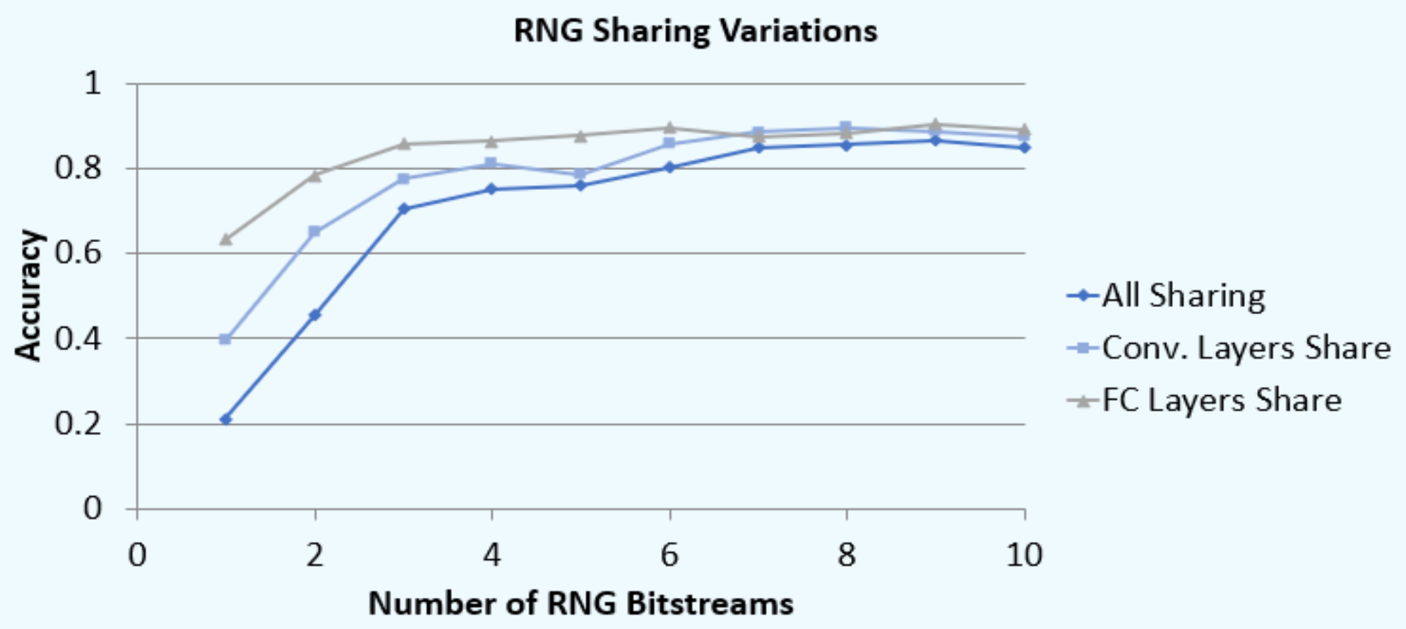
Convolutional and Fully Connected Layers

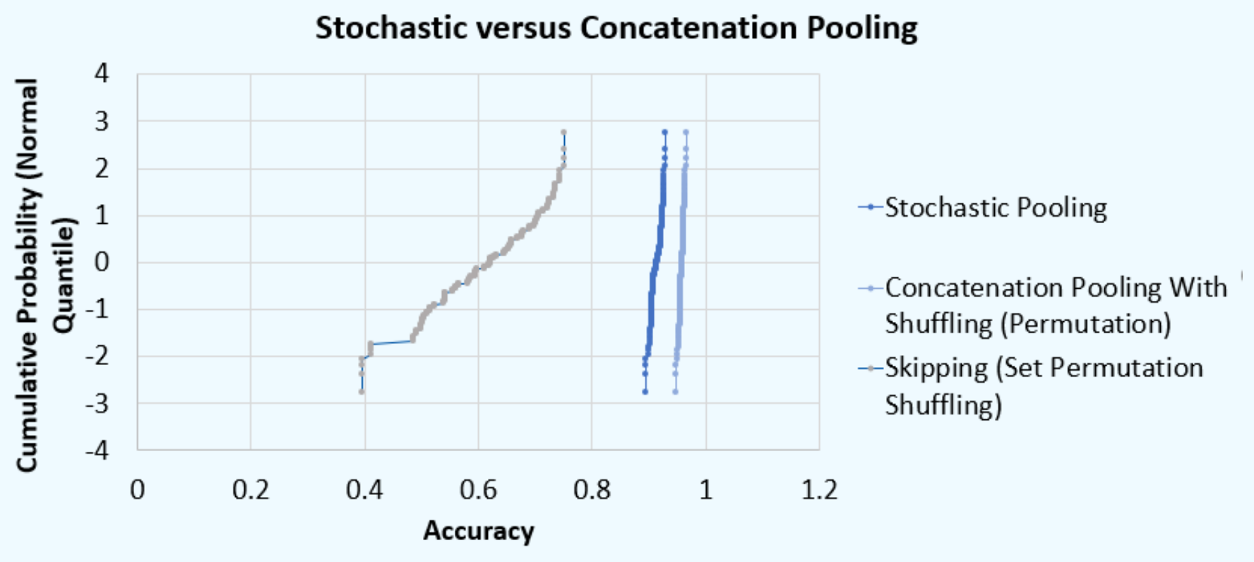
- **All numeric calculations** performed in stochastic with ability to choose OR versus MUX accumulation layers
- Supports **different kernel, padding, stride and pooling sizes**.
- Fully-connected layer support.



Evaluation and Initial Results







Summary

- Simulator allows for the evaluation of SC application in neural networks
- Previous SC problems alleviated through:
  - OR-base accumulation
  - Split-unipolar representation
  - Average pooling with computation skipping

Acknowledgements

Thank you to Professor Pamarti for his feedback and NSF REU and DARPA for their support.