Graph Partitioning as Quadratic Unconstrained Binary Optimization (QUBO) on Spiking Neuromorphic Hardware

International Conference on Neuromorphic Systems

ICONS 2019, Knoxville, TN

Susan M. Mniszewski

smm@lanl.gov

July 23-25, 2019

Presented by: Forrest C. Sheldon

National Nuclear Security Administration

Managed by Triad National Security, LLC for the U.S. Department of Energy's NNSA

LA-UR-19-26947

Neuromorphic Computing

- Brain-inspired computers, devices, and models
- Architecture of synthetic neurons and synapses
- Characterized by:
  - Collocation of memory and compute
  - Massively parallel processing
  - Large connectivity
  - Spiking codes for communications between components
  - Event-driven computation
  - Low power consumption
  - Stochastically firing neurons for noise
  - Scalability
- Viable choice as co-processors in hybrid computing systems

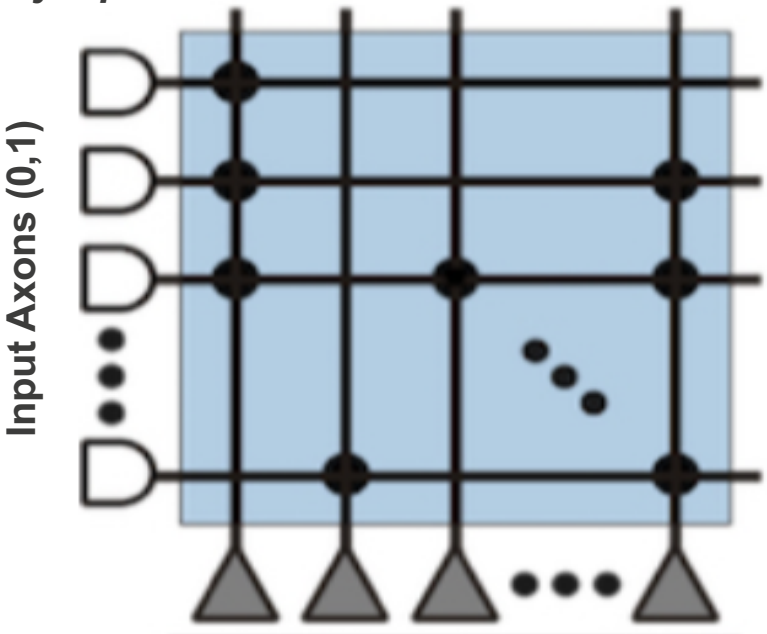
Spiking Neuromorphic Hardware and Applications

- Provides a more biologically realistic neuron model
- Allows for spiking, sparse, low precision communication
- Example hardware
  - IBM TrueNorth
  - Intel Loihi
  - SpiNNaker
  - BrainScaleS
- Applications
  - Model neuroscience theories
  - Solve challenging machine learning problems
  - Classical computing non-neural network challenges
    - Matrix operations
    - Combinatorial optimization problems

IBM TrueNorth (TN) Neuromorphic Spiking Architecture

- Brain-inspired computing
- Each core contains electronic implementations of neurons, synapses, and axons
  - Perform computation, memory, and communication
  - 256 neurons, 256 axons, and matrix of 65,536 synapses (crossbar)
- Thousands of cores tiled into a single chip
- Multiple chips tiled together into large scalable networks
- Neuron behavior – integrate and fire
  - Integration of connected axon weights, spike when exceeds threshold α
  - Gradual decay or leak, can be stochastic
  - Crossbar defines connectivity between axons and neurons
  - Axon types and weights limited to 4, multiple axons can have same type
  - 23 configurable parameters per neuron type

Synapse matrix or crossbar for connectivity



Neurons spike when threshold is exceeded

TN Core

Quadratic Unconstrained Binary Optimization (QUBO)

- Motivated by QUBO-based graph partitioning originally developed for D-Wave quantum annealer (2X and 2000Q)
- Also, shown to work for universal quantum gate model (IBM Quantum Experience) for small graphs
- Now, QUBO on TrueNorth Neuromorphic Spiking Architecture
  - Optimization algorithm as a spiking network implementation
  - Hybrid neuromorphic-classical algorithm using pseudo simulated annealing
- General method that can be applied to NP-hard problems that can be framed as QUBOs
  - Graph-based data decomposition for distributed HPC simulations, mesh partitioning
  - Unsupervised machine learning for community detection and data clustering
- Advantages of Neuromorphic computing
  - Low power
  - Run large problems on scalable architecture

QUBO on D-Wave Quantum Annealer

- Motivated by graph partitioning/clustering methods and implementations developed to run on the D-Wave 2X/2000Q Quantum Annealer
  - *k*-concurrent partitioning all at once
  - *k*-concurrent clustering all at once
- Formulated as a QUBO with penalty constants
- **Minimize/maximize the objective function:** $O(Q,x) = \Sigma Q_{ii} x_i + \Sigma Q_{ij} x_i x_j$
- Demonstrated "proof of principle" results on benchmark graphs, example graphs and electronic structure graphs
- Results were shown to be comparable or out-perform current *state of the art* methods such as METIS and KaHIP

H. Ushijima-Mwesigwa, C. F. A. Negre, S. M. Mniszewski, Graph Partitioning using Quantum Annealing on the D-Wave System, *Proceedings of the 2nd International Workshop on Post Moore's Era Supercomputing,* 2017, 22-29.

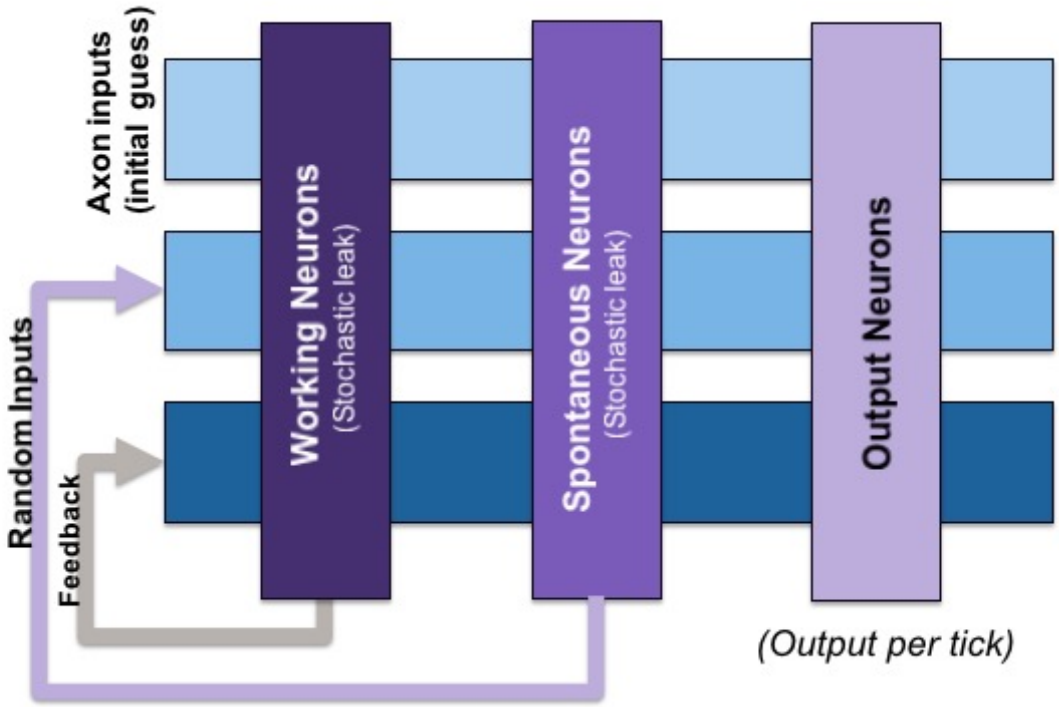
C. F. A. Negre, H. Ushijima-Mwesigwa, S. M. Mniszewski, Detecting Multiple Communities Using Quantum Annealing on the D-Wave System, *arXiv preprint* https://arxiv.org/abs/1901.09756, 2019.

QUBO on TrueNorth Core

- **QUBO** Formulation for 2-partitioning
  - Derive QUBO with penalty constants (ex. balancing, minimize cut edges)
  - Maximize the objective function: $O(Q,x) = \Sigma Q_{ii} x_i + \Sigma Q_{ij} x_i x_j$
  - Q is a symmetric matrix of QUBO weights
- Hybrid neuromorphic-classical pseudo simulated annealing with temperature schedule
- SA_QUBO Corelet created given Q matrix and temperature
  - **Program network based on QUBO weights**
    - Reorder (Reverse Cuthill-McGee), compress, contract, reuse and overlap weights to fit into axon weight limit – 4 weights/neuron (try weight quantization next)
    - Determine neuron thresholds based on weights
    - Define crossbar connectivity based on presence of weights
  - **Initial guess – N input axons for N node graph**
    - Initial random guess - values of 0 and 1 for 2-partitioning or best result from previous SA_QUBO Corelet
  - **Spiking neurons reinforce through feedback and sent to output**
  - **Decay/leak contributes to neuron membrane potential - causes other neurons to fire**
  - **Spontaneous Stochastic neurons with temperature parameter for exploring sample space**
- Results on up to 30 node graphs show promise – approximate or optimal

SA_QUBO on IBM TrueNorth

- **Synapse matrix or crossbar for axon-neuron connectivity**
- **QUBO weights on axons**
- **Leaky-Integrate-Fire (LIF) neurons**
- **Neurons spike when threshold exceeded**
- ***SNN* noise drives sampling**
  - Spontaneous neurons add nodes to result
  - Gradual leak/decay removes nodes
- **Simulated Annealing metaheuristic**
  - Design or learn annealing schedule
  - Multiple SA_QUBO Corelets - one per temperature
- **Track highest energy results**



SA-QUBO: 3 sets of neurons and 3 sets of axons

Classical Pseudo Simulated Annealing

- Metaheuristic to approximate global optimization in a large search space for an optimization problem
- Requires a temperature variable *T* to vary through the process
- Annealing schedule: Start with high *T* and gradually reduce to a low *T*
- Working neuron stochastic leak and spontaneous stochastic leak serve as *T* (values can be 1-256)
- Corelet does not allow for parameters to change while running
- Separate Corelet required for each temperature value that runs for 100 ticks

Annealing Schedule:

Number	Temp (%)	Temp (count)
1	50.0	128
2	12.5	32
3	3.125	8
4	1.5625	4

Graph Partitioning into 2 Parts - Results

Graph Size (number of nodes)	TrueNorth Energy	TrueNorth Subgraph Size Split	D-Wave Energy	D-Wave Subgraph Size Split
4	4	2/2	4	2/2
6	10	4/4	10	3/5
10 10 10	6 6 6	5/5 6/4 4/6	6	5/5
16 16 16	15 16 17	8/8 9/7 8/8	17	8/8
30 30	35 36	14/16 15/15	37	14/16

Conclusion

- **Summary**
  - This study has demonstrated a ”proof of principle” **neuromorphic-classical** approach using pseudo simulated annealing driving a QUBO solver on the IBM TrueNorth
  - Stochastic noise in the system proved to be useful as a sampling capability
- **Future directions**
  - Extensions for larger graphs over multiple cores
  - Other graph algorithms that can be framed as QUBO formulations
    - *k*-partitioning, community detection, graph coloring, maximum cut, and more
  - Implementation on the Intel Loihi