OAK RIDGE National Laboratory

Device architectures to meet neuromorphic computing challenges

Catherine Schuman

Research Scientist

Oak Ridge National Laboratory

February 28, 2020

ORNL is managed by UT-Battelle, LLC for the US Department of Energy

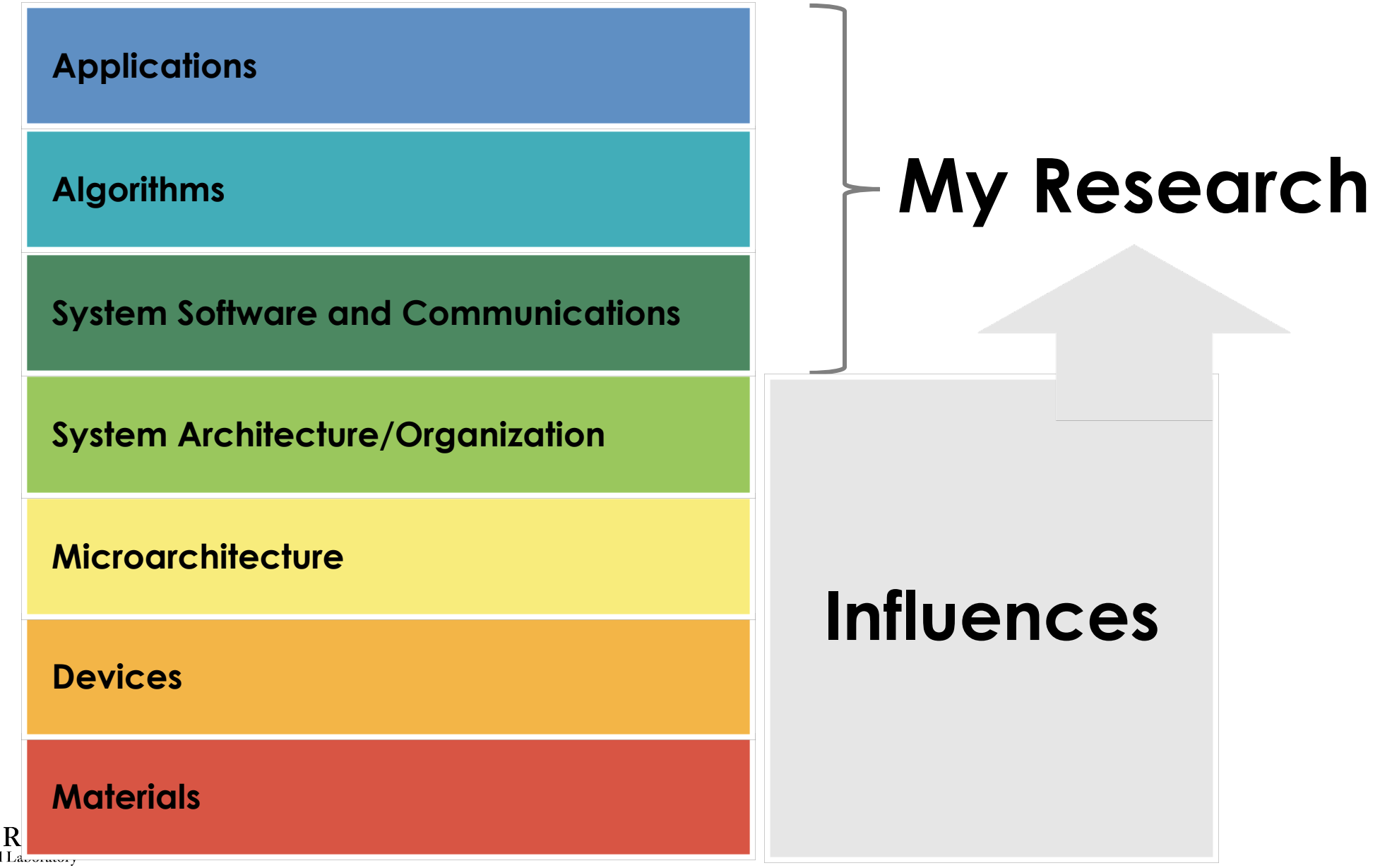
U.S. DEPARTMENT OF ENERGY

My Background

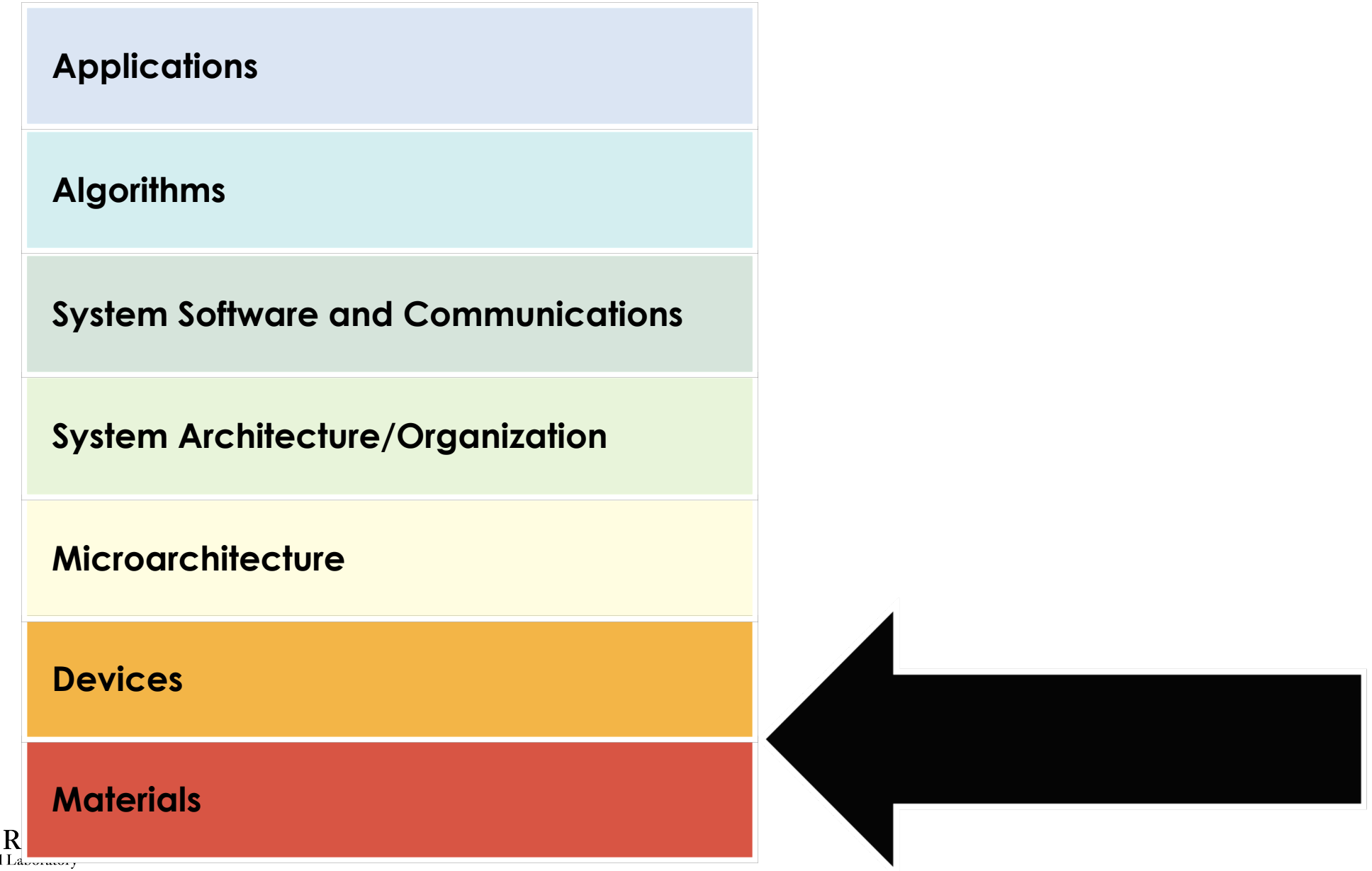
- Ph.D. in Computer Science from the University of Tennessee
  - National Science Foundation Graduate Research Fellowship to study evolutionary algorithms and spiking neural networks
- Joined ORNL in 2015 as a Liane Russell Early Career fellow
  - Project: Programming and Usability of Neuromorphic Computing
- 45+ publications in spiking neural networks and neuromorphic computing, 6 patents
  - A Survey of Neuromorphic Computing and Neural Networks in Hardware
- Joint faculty with the Department of Electrical Engineering & Computer Science at the University of Tennessee
- Co-founder of the TENNLab
- Department of Energy Early Career Award in 2019

TENN LAB
NEUROMORPHIC
ARCHITECTURES. LEARNING. APPLICATIONS.

Neuromorphic Computing "Stack"



Neuromorphic Computing "Stack"



Memristive Devices and Materials

- Memristive materials are being used to implement both neurons and synapses
- Basic functionality of the component depends on the materials and devices utilized
- All aspects of the compute stack are influenced by the materials/devices used at the lowest level

Kim, Sun Gil, et al. "Recent advances in memristive materials for artificial synapses." *Advanced Materials Technologies* 3.12 (2018): 1800457.

Metal Oxide Memristors Materials

- Circuit components fabricated with different materials can have different behaviors:
  - Number and type of resistance states
  - Switching speeds
  - Endurance
  - Stability
  - Reliability
  - Cost
  - Tunability

Material	TE/BE	V_{SET}/V_{RESET}	Switching Speed	Retention Time	Endurance
ZnO	Ag/Cu	1.2V/-1.25V	-	-	>500 cycles
TiO_2	TiN/Pt	+1V/-1.5V	1uS	10^4 s	
LaO	ITO/$SrTiO_3$	5V/-1.6V	-	>4x10^4 s	2000 cycles
TaO_x	W/Pt	-	-	>10 years	10^4 cycles
NiO	Pt/Pt	>10V/<-10V	-	>10^4 s	-

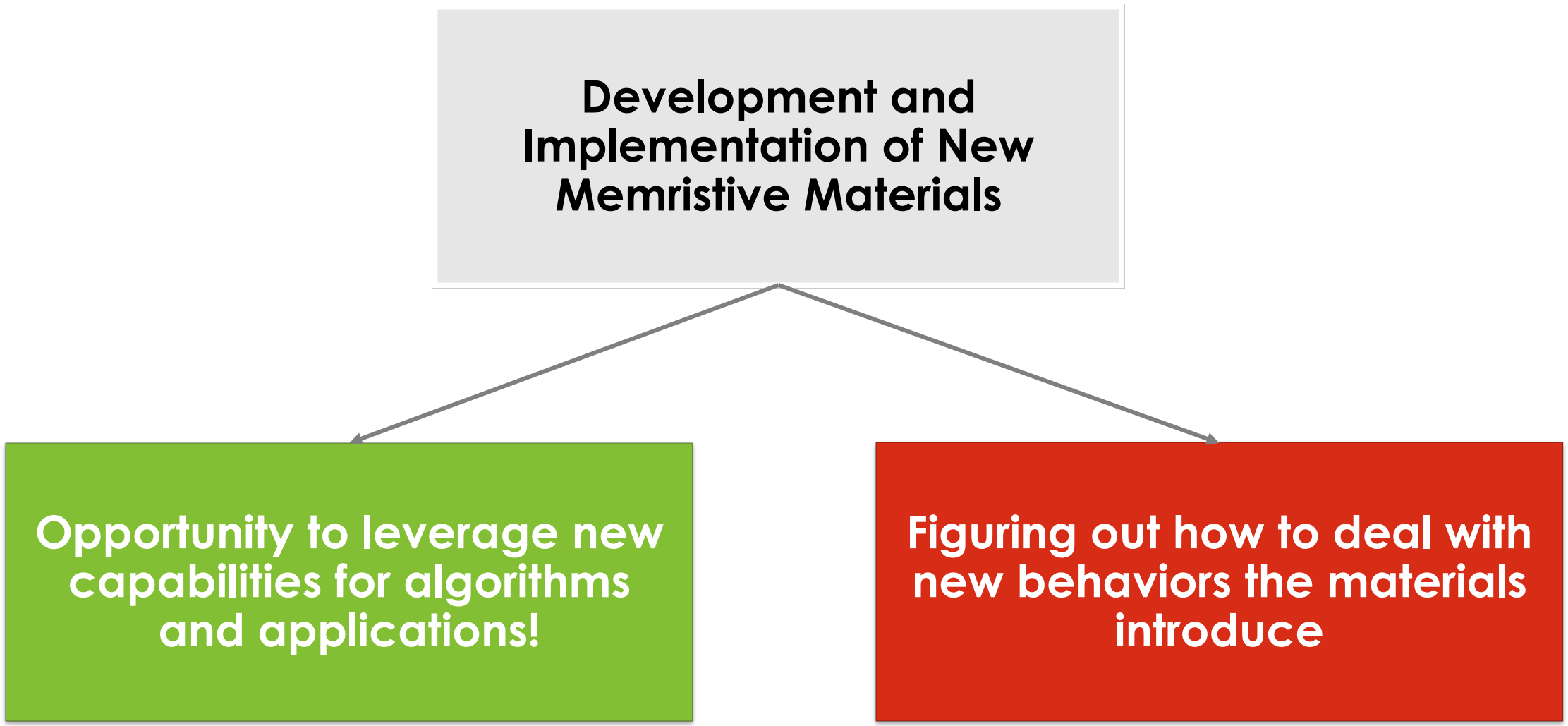
Table: Mohammad, Baker, et al. "State of the art of metal oxide memristor devices." *Nanotechnology Reviews* 5.3 (2016): 311-329.

Challenges (and Opportunities) with Memristive Materials

- Variability from device to device and from cycle to cycle
  - Advantages:
    - Could help compensate for limited weight resolution
    - Source of stochasticity (which can be useful in some neural network implementations)
  - Disadvantages:
    - May require re-training
    - May require on-chip training
- Weight resolution
  - Disadvantage:
    - Limited resistance levels require limited programmable weight values
  - Advantage:
    - May benefit generalization ability of a neural network

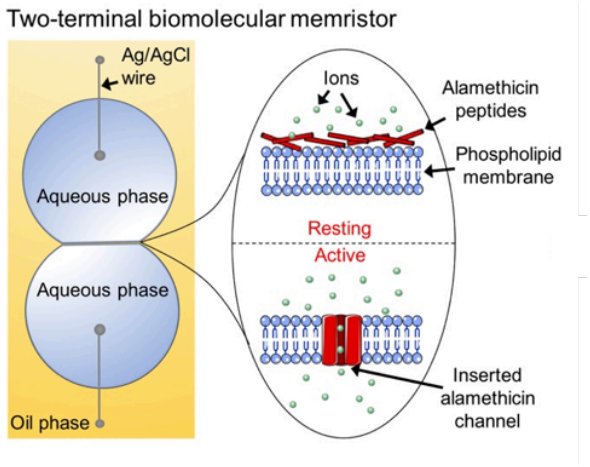
Li, Yibo, et al. "Review of memristor devices in neuromorphic computing: materials sciences and device challenges." *Journal of Physics D: Applied Physics* 51.50 (2018): 503002.

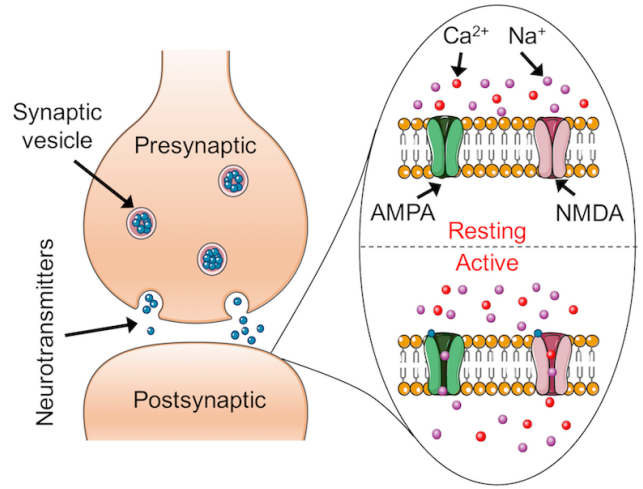
Materials Research from an Algorithms/Applications Perspective

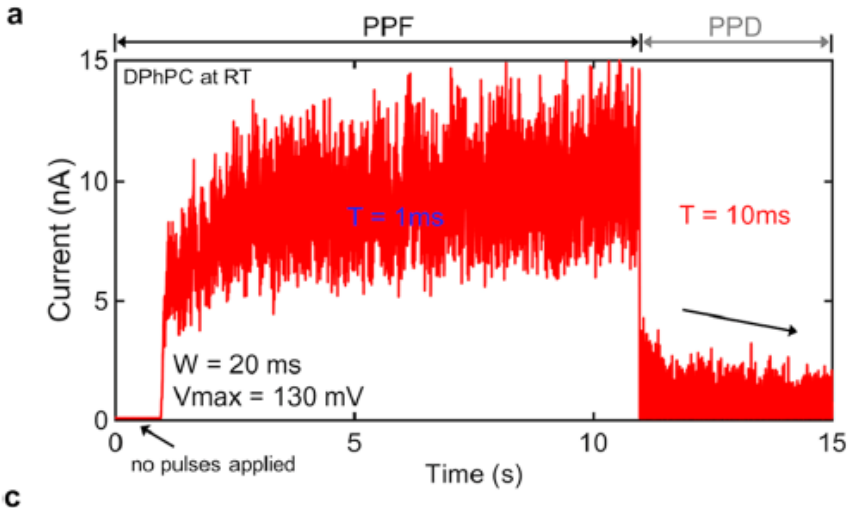


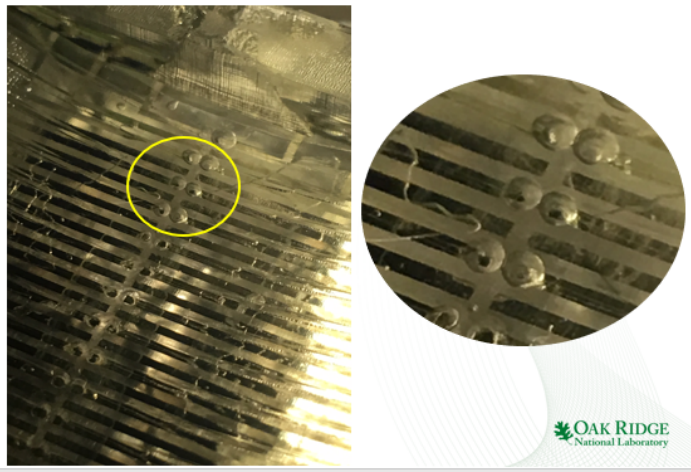
Example Device: Biomimetic Synapses

- Droplet-interface bilayers (DIBs) form the synapses
- DIBs exhibit natural short-term plasticity (PPF and PPD)
- Easily fabricated on a small scale





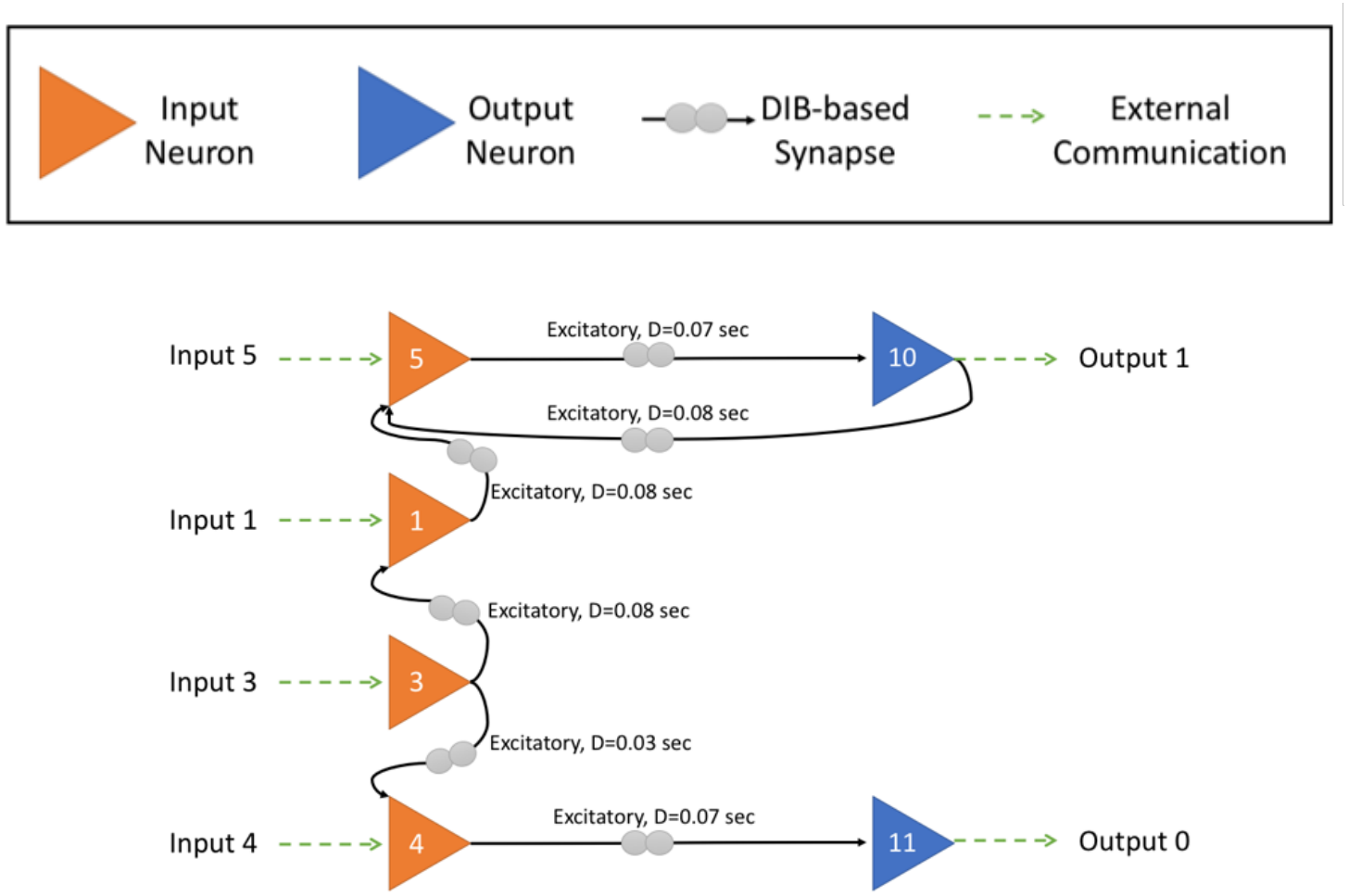




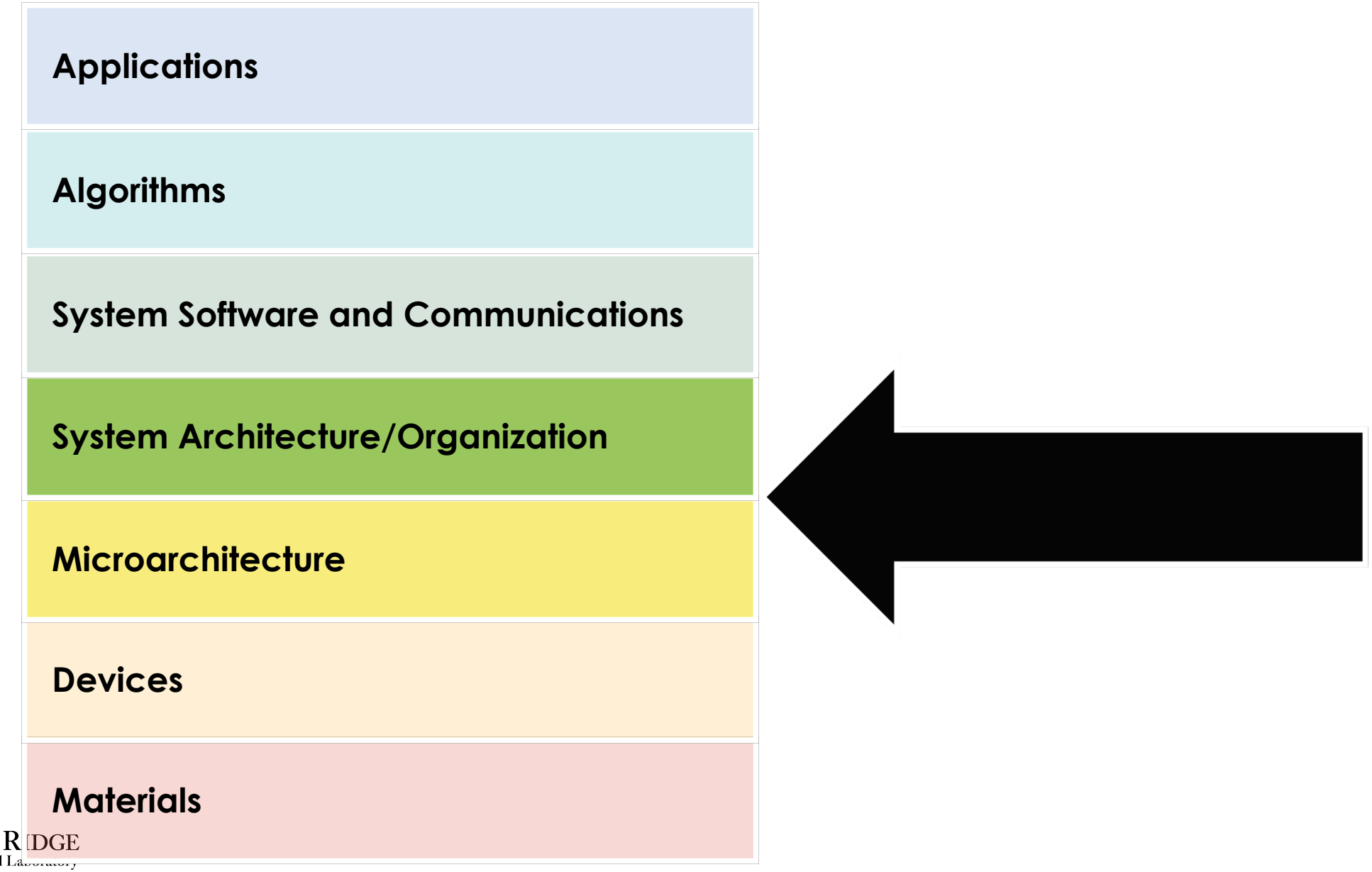
Najem, Joseph S., et al. "Memristive ion channel-doped biomembranes as synaptic mimics." *ACS nano* 12.5 (2018): 4702-4711.

Example Device: Biomimetic Synapses

- Implemented an analytical model of the DIB behavior
- Implemented a software simulation of the DIB network
- Used simulation to train an EEG classifier that achieves 98.25% accuracy on the training and testing set, comparable with other neuromorphic architectures



Neuromorphic Computing "Stack"

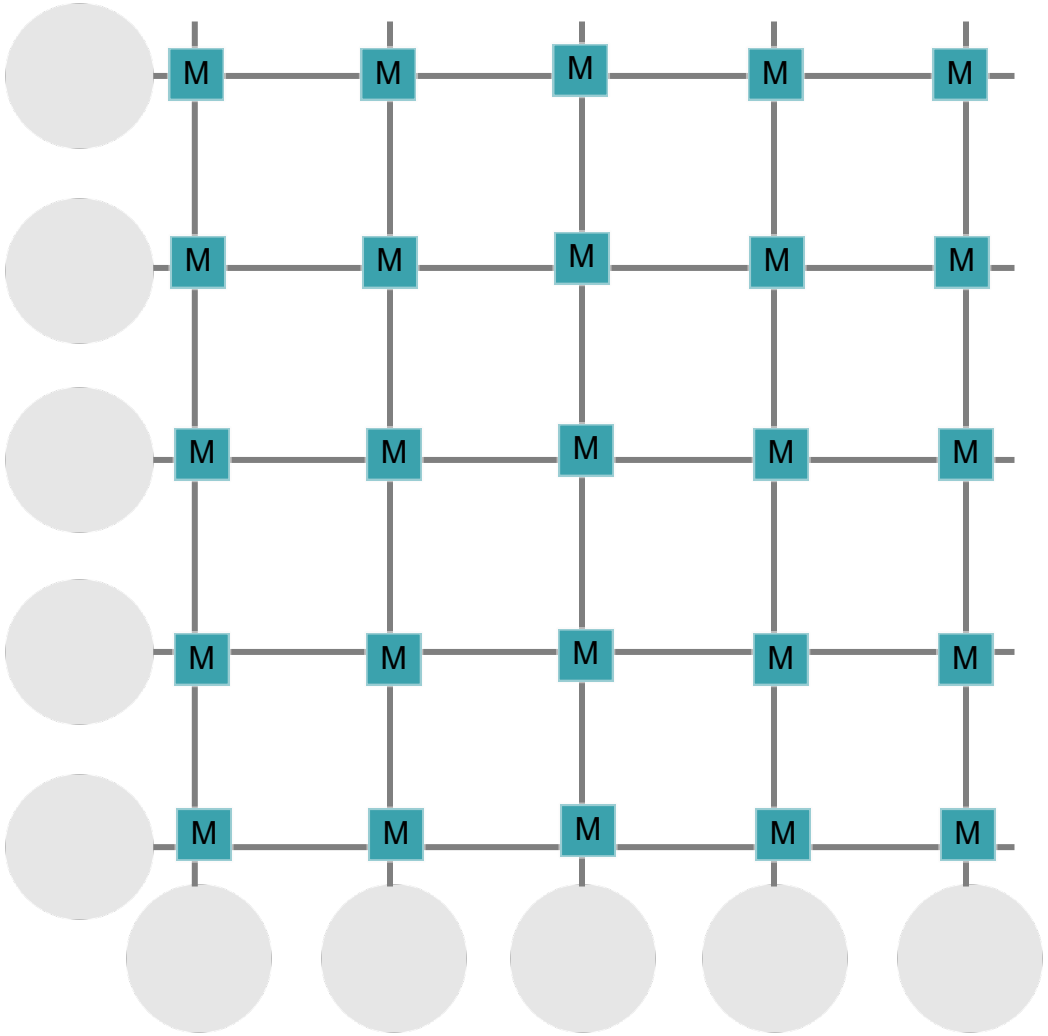


Architectures for Memristive Neuromorphic Systems

- Architecture includes additional components (often implemented in CMOS), including:
  - Neurons (though they may be implemented with memristors as well)
  - Learning circuitry
- Architectural organization influences for algorithms and applications:
  - Possible network structure to implement
  - On-chip learning mechanisms
  - Performance characteristics

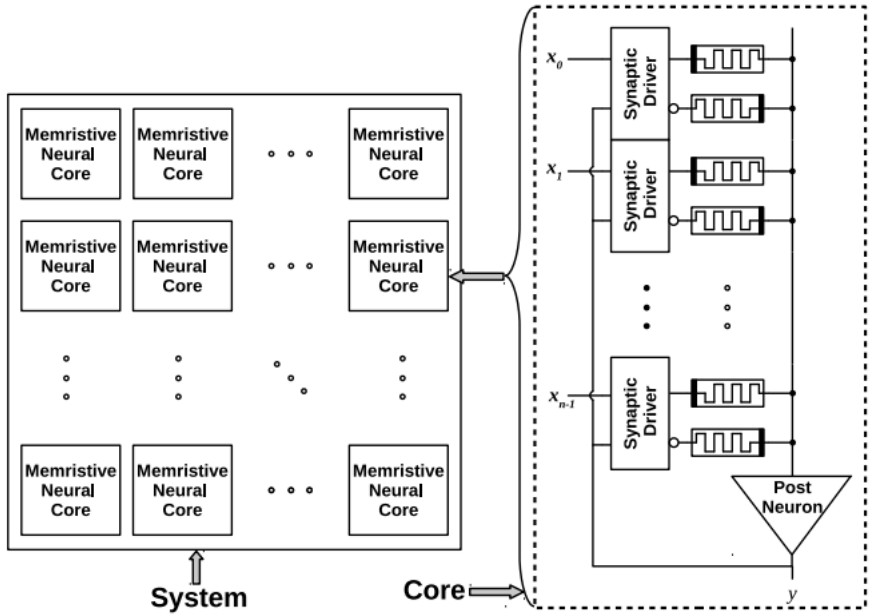
Architectures: Crossbars

- Advantages:
  - High density
  - Low power
- Disadvantages:
  - Potential limitation on achievable structures (number of layers, recurrent connectivity, etc.)
  - Sneak paths updating memristor values
  - May not be well-suited to sparse structures



Architectures: Neural Cores

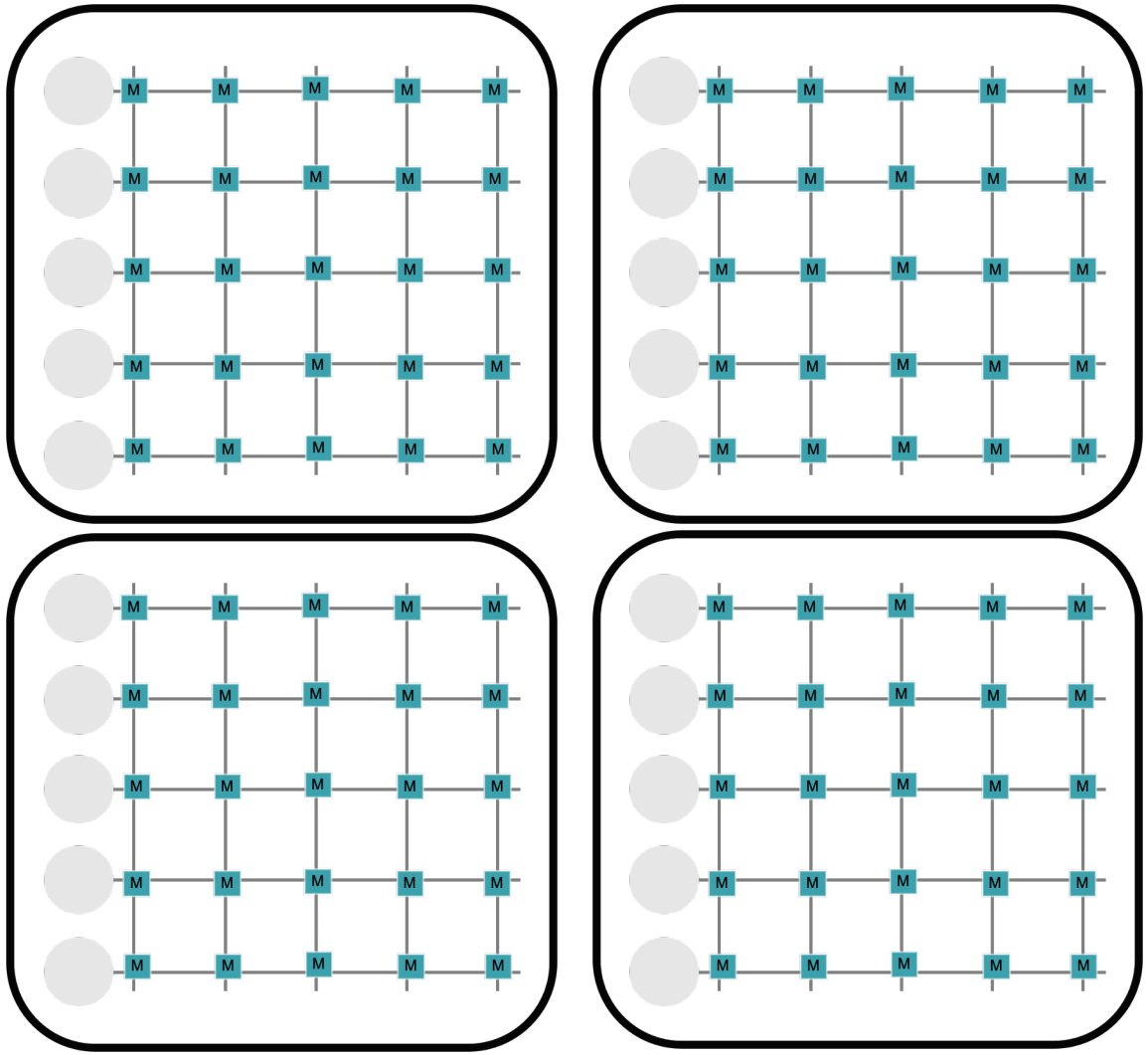
- Advantages:
  - More flexibility in the types of network structures that can be realized
  - Less idle components or unused computation for small, sparse networks
- Disadvantages:
  - Not as space efficient
  - Not as power efficient



Chakma, Gangotree, et al. "Memristive mixed-signal neuromorphic systems: Energy-efficient learning at the circuit-level." *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 8.1 (2017): 125-136.

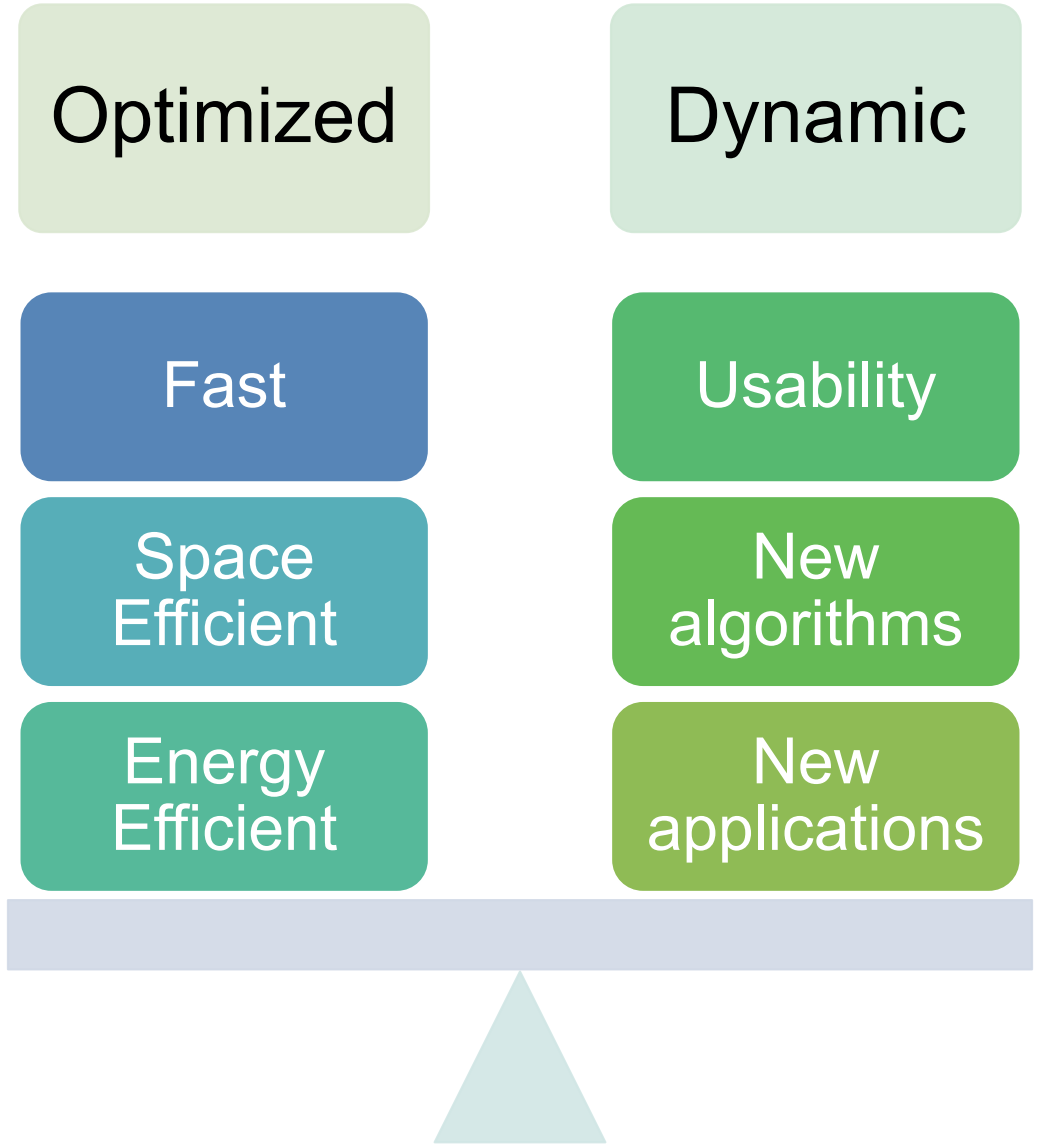
Architectures: Hybrid Crossbar/Neural Core

- Reduces the size of the crossbars used, limiting sneak paths
- Allows for more flexible network structures
- Some density and power efficiency benefits from the crossbar approach

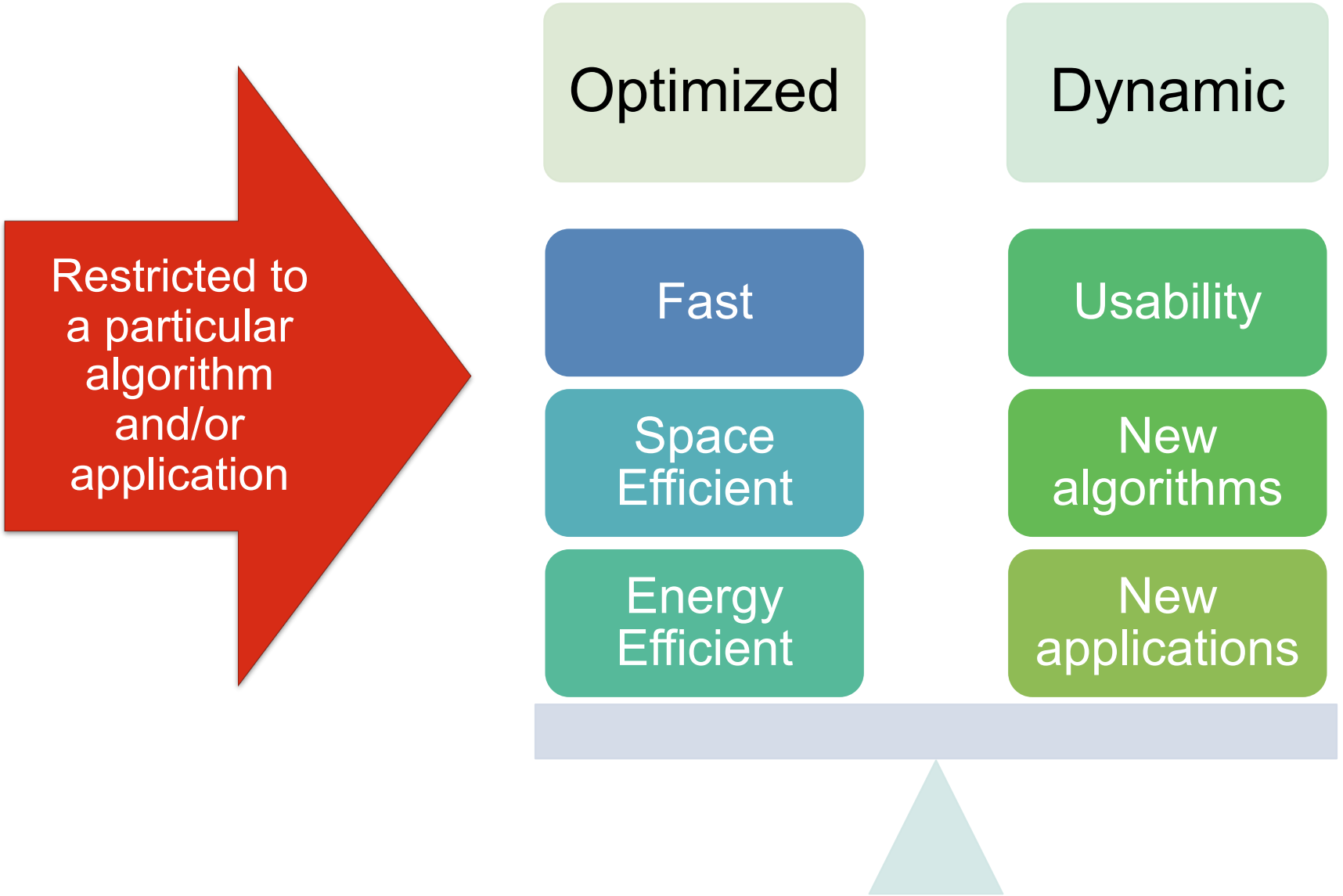


Yakopcic, Chris, and Tarek M. Taha. "Energy efficient perceptron pattern recognition using segmented memristor crossbar arrays." *The 2013 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2013.

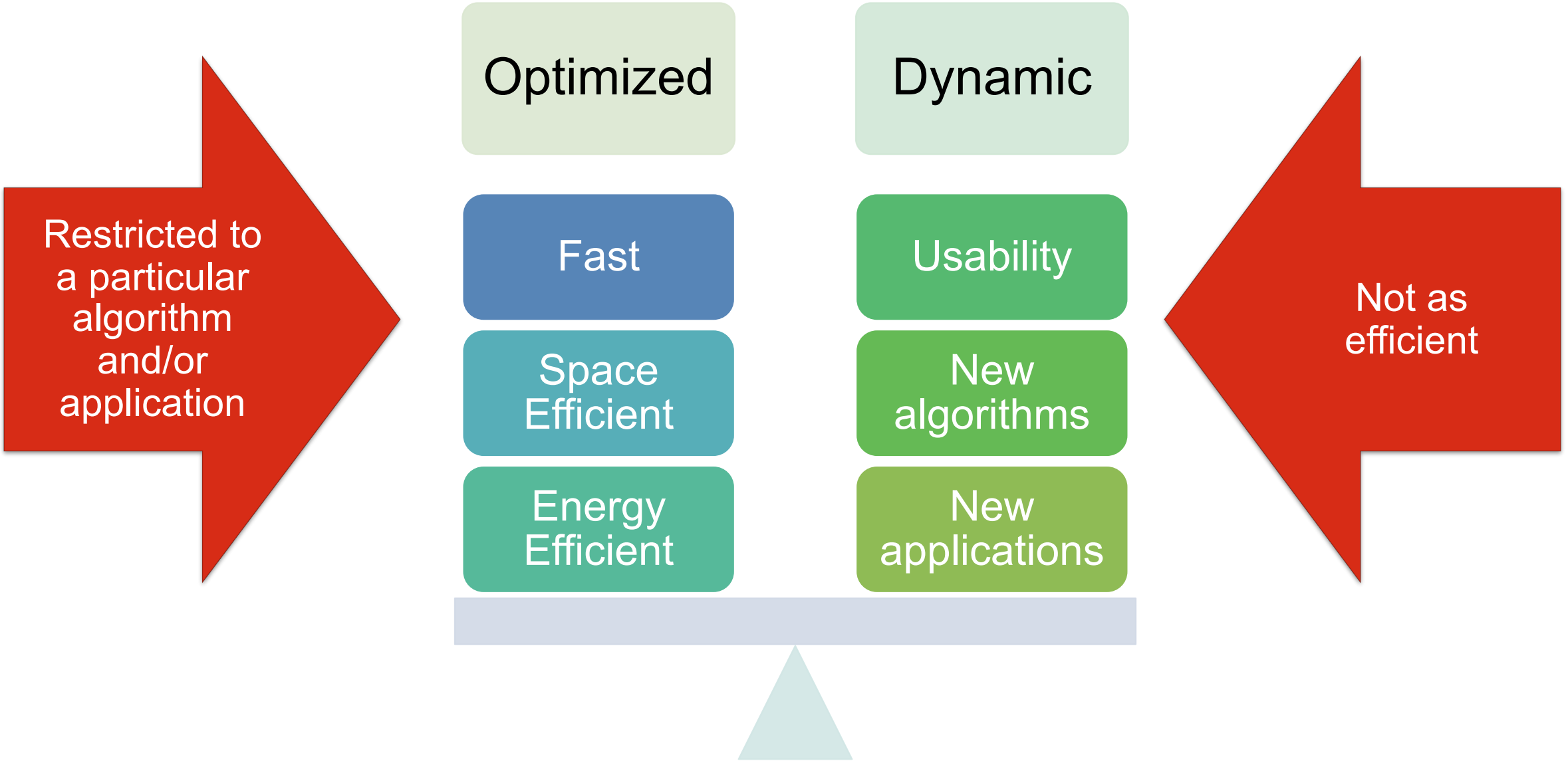
Architectures from an Algorithms/Application Perspective



Architectures from an Algorithms/Application Perspective



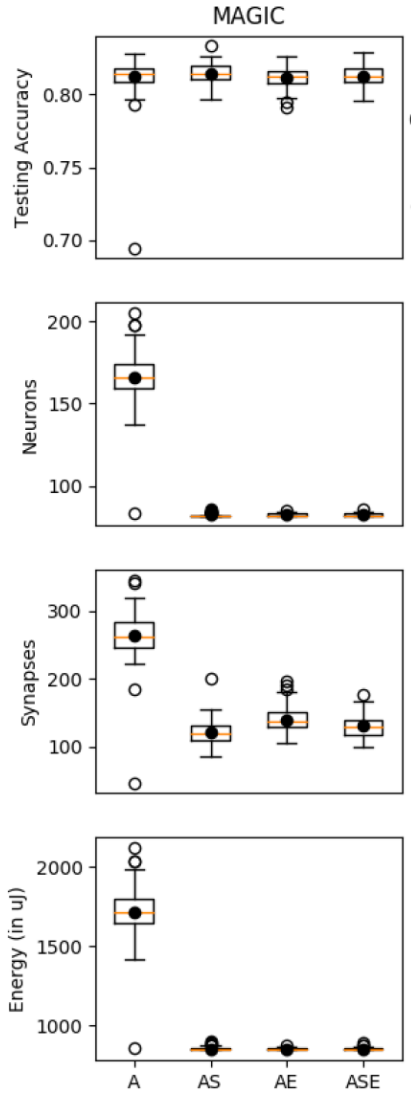
Architectures from an Algorithms/Application Perspective



Low SWaP Opportunities for ASIC Memristive Chips

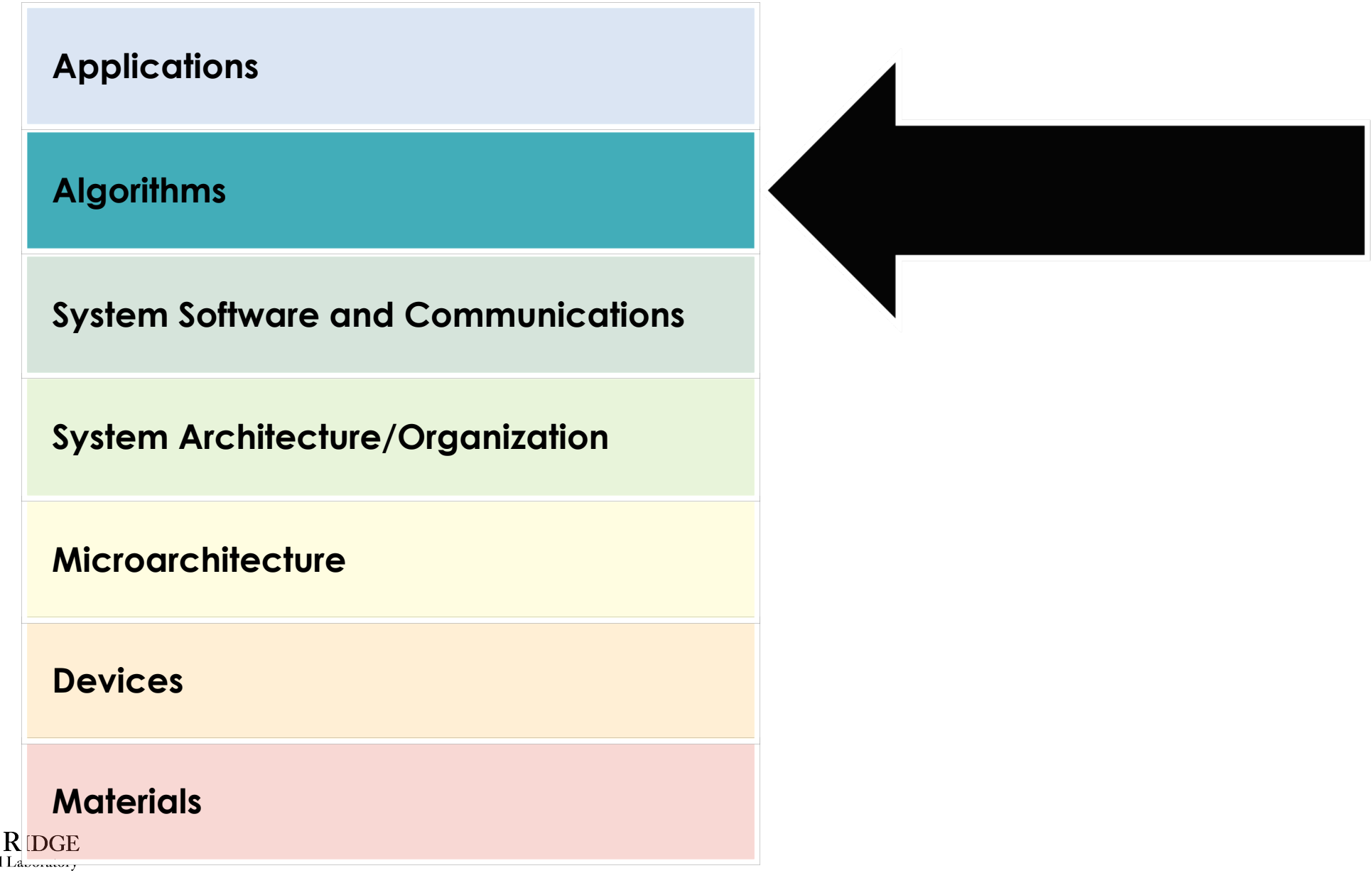
- Combining application specific memristive chips with algorithms that optimize for smaller size and lower energy usage can result in dramatically lower energy usage with very little impact on application performance
- May be very well-suited to deployment at the edge

Optimizing	Testing	Neurons	Synapses	Energy
Accuracy	82.8%	168	257	1766.7 uJ
Accuracy+Size	83.4%	82	128	846.2 uJ
Accuracy+Energy	82.6%	82	120	847.2 uJ
Accuracy+Size+ Energy	82.9%	82	108	845.5 uJ



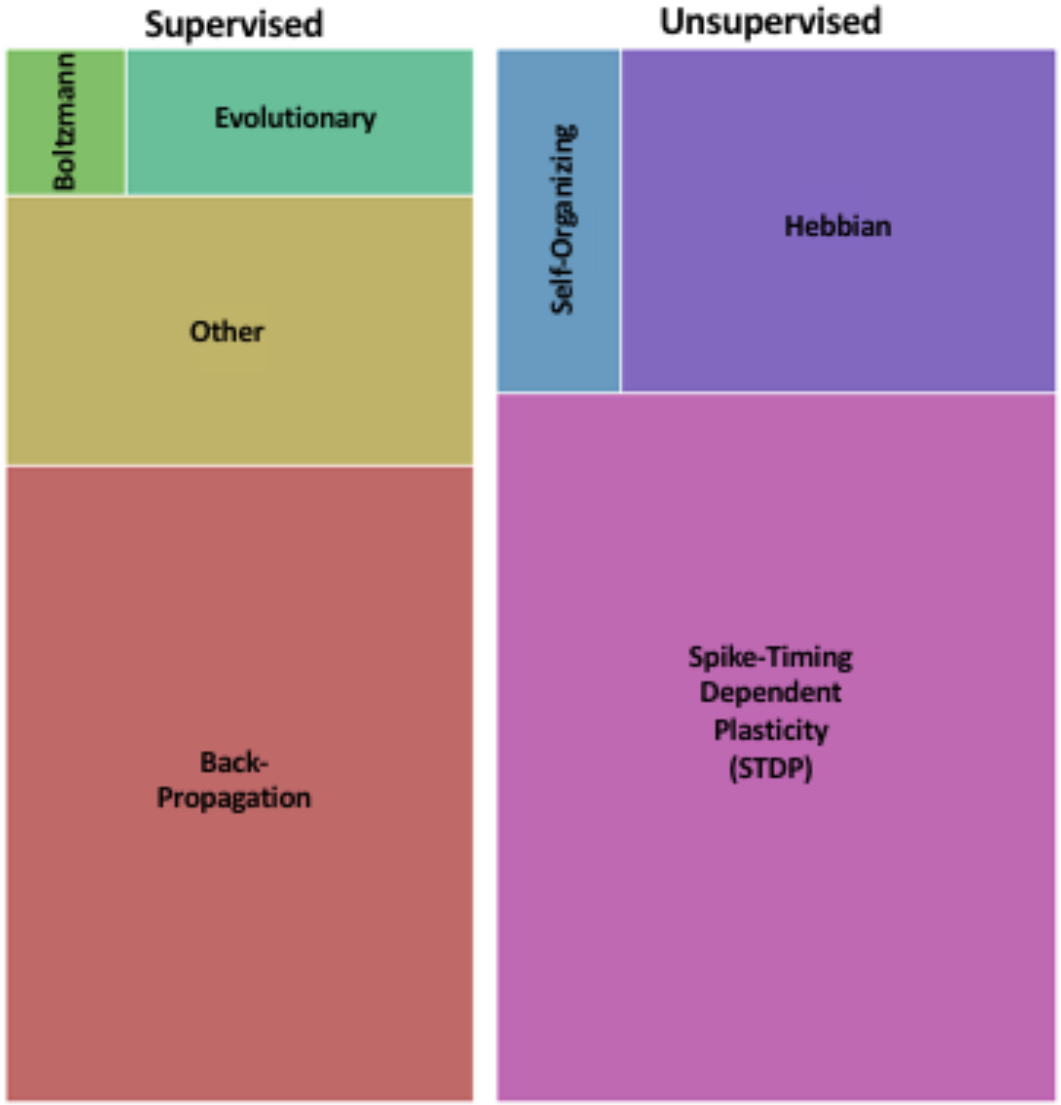
Submitted: Catherine D. Schuman, et al. “Automated Design of Neuromorphic Networks for Scientific Applications at the Edge.”

Neuromorphic Computing "Stack"



Algorithms for Memristive Neuromorphic Systems

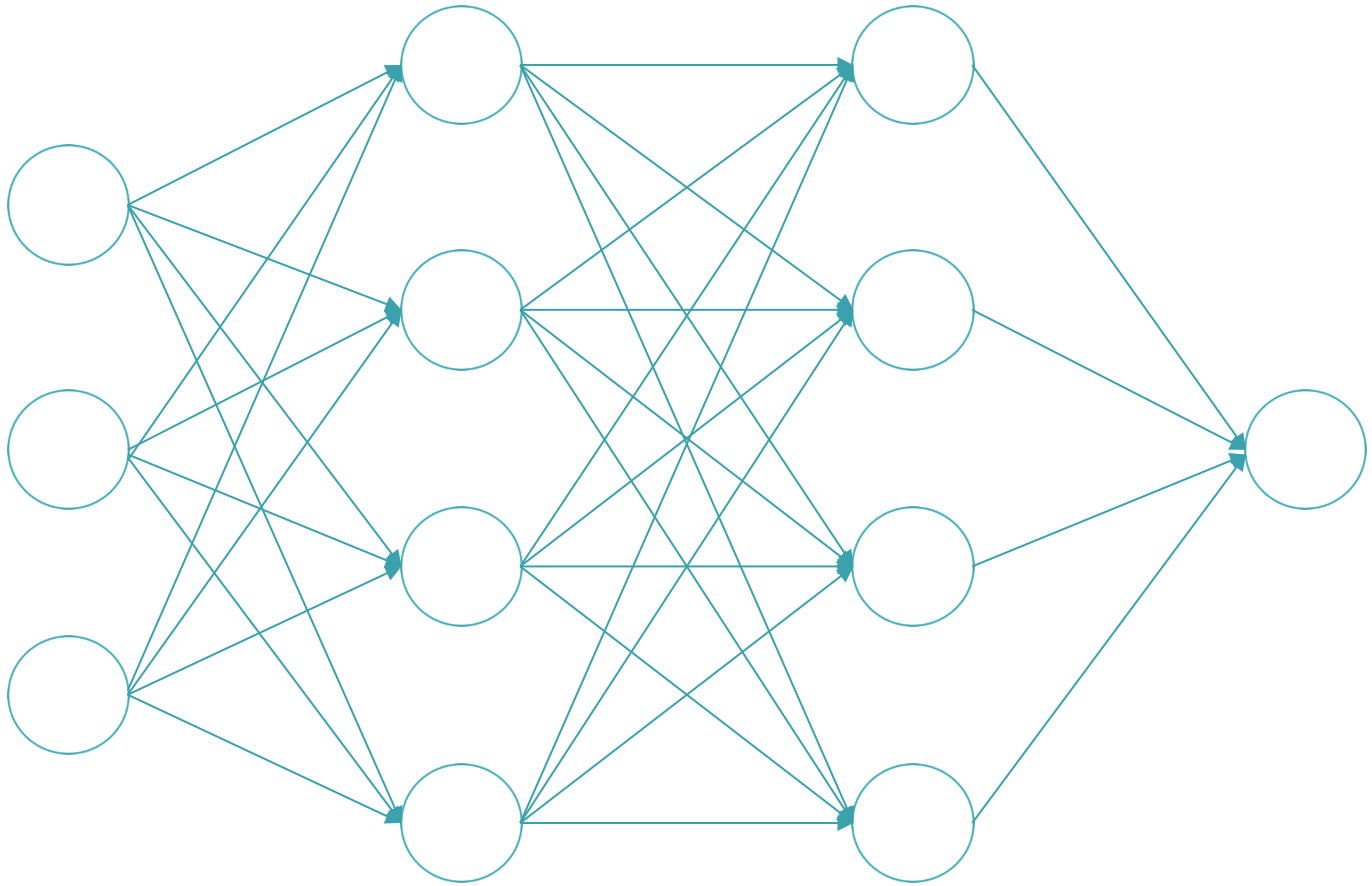
- Key considerations for algorithm development on neuromorphic hardware:
  - Realizable network structures
  - Reduced precision in the synaptic weights
  - On-chip training, chip-in-the-loop, or off-chip training performance
  - Dealing with noise, process variations, cycle-to-cycle variation
  - Hardware optimized for training or inference



Schuman, Catherine D., et al. "A survey of neuromorphic computing and neural networks in hardware." *arXiv preprint arXiv:1705.06963* (2017).

Algorithms: Back-Propagation-Like Approaches

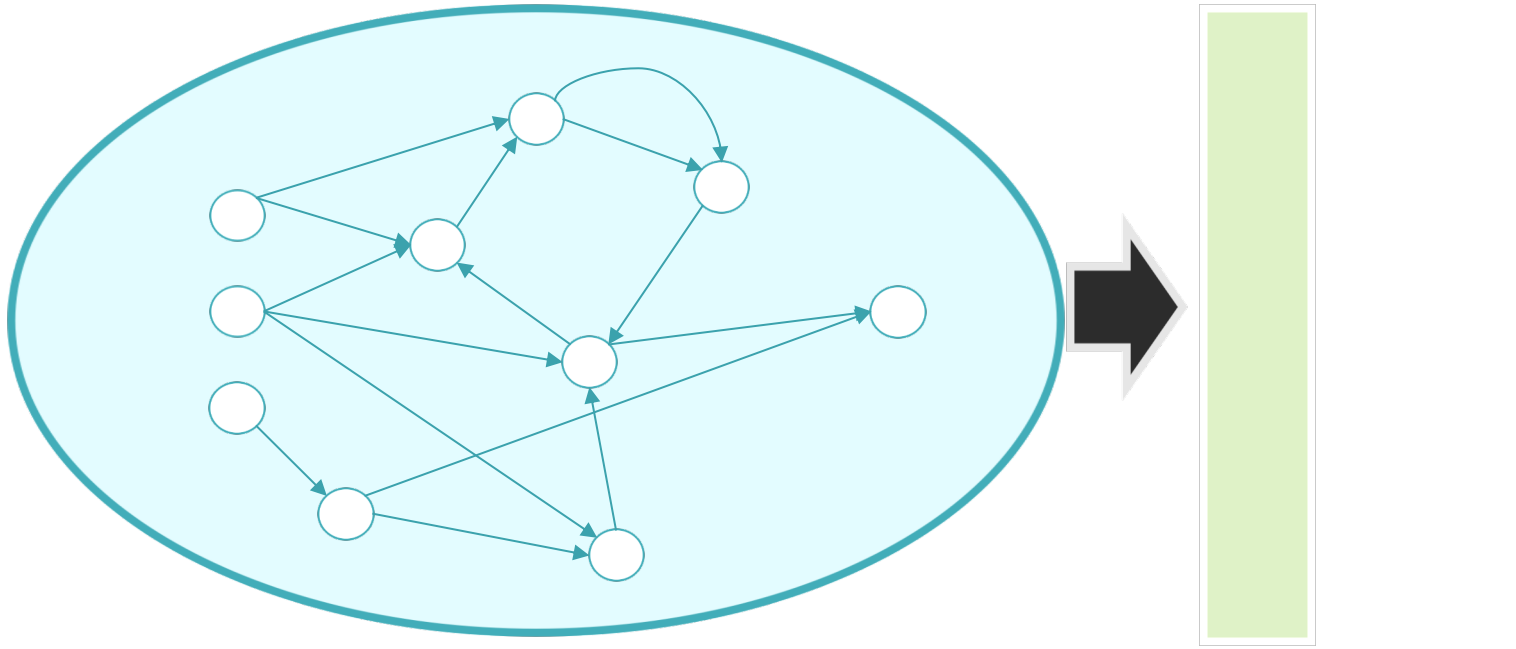
- Dense connectivity
- Algorithm adaptations for low-precision
- Multiple memristors per synapse may be required to increase precision
- On-chip training or chip-in-the-loop may be required to overcome device variability and alternate current paths



Hasan, Raqibul, Tarek M. Taha, and Chris Yakopcic. "On-chip training of memristor based deep neural networks." *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017.

Algorithms: Reservoir Computing

- Dense (readout layer) and sparser (liquid) connectivity
- Has been shown to be resilient to noise due to process variation
- Well-suited to memristors because it benefits from their nonlinear and memory characteristics



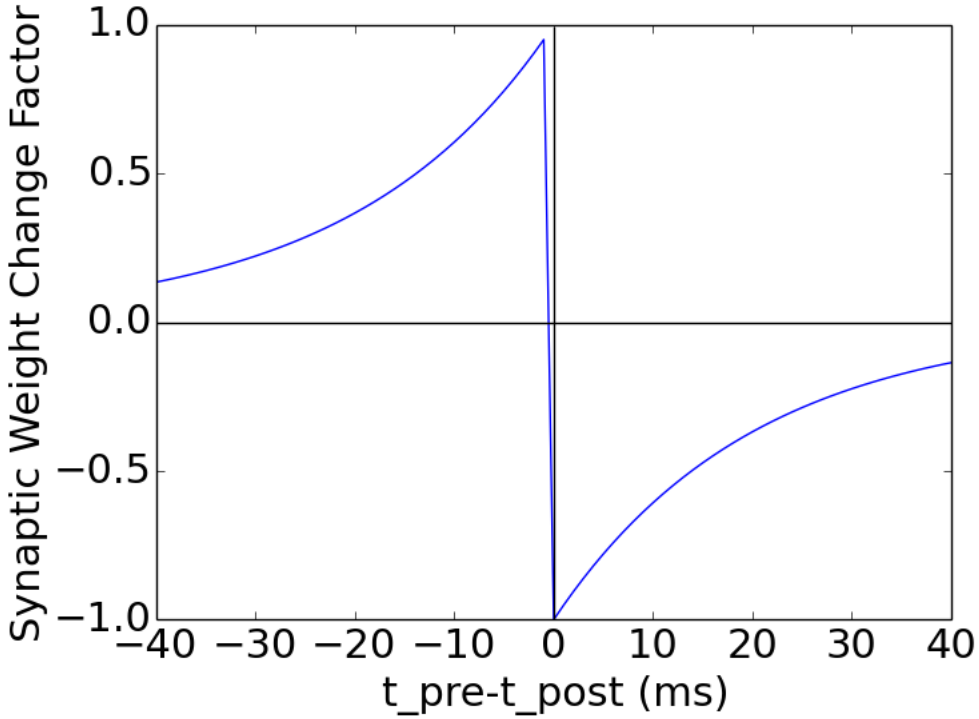
Untrained Reservoir

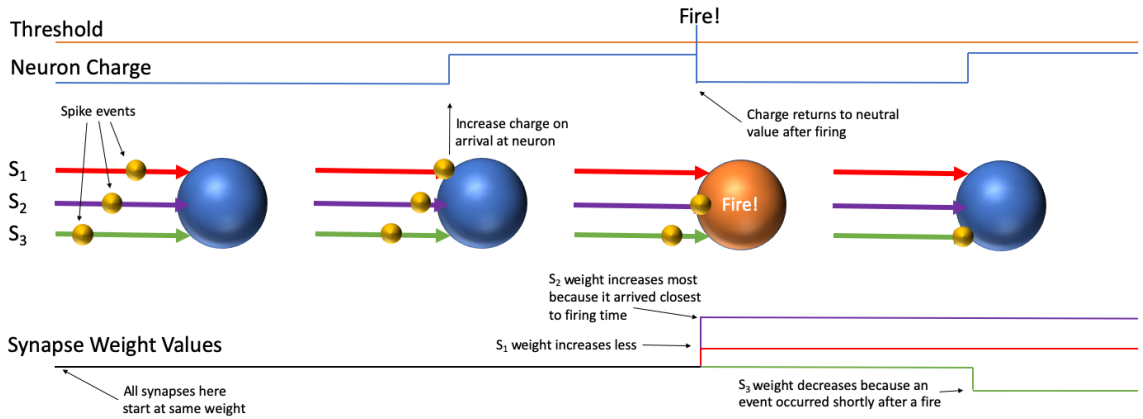
Trained Readout Layer

Soures, Nicholas, Lydia Hays, and Dhireesha Kudithipudi. "Robustness of a memristor based liquid state machine." *2017 international joint conference on neural networks (ijcnn)*. IEEE, 2017.
Kulkarni, Manjari S., and Christof Teuscher. "Memristor-based reservoir computing." *2012 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*. IEEE, 2012.

Algorithms: Synaptic Plasticity

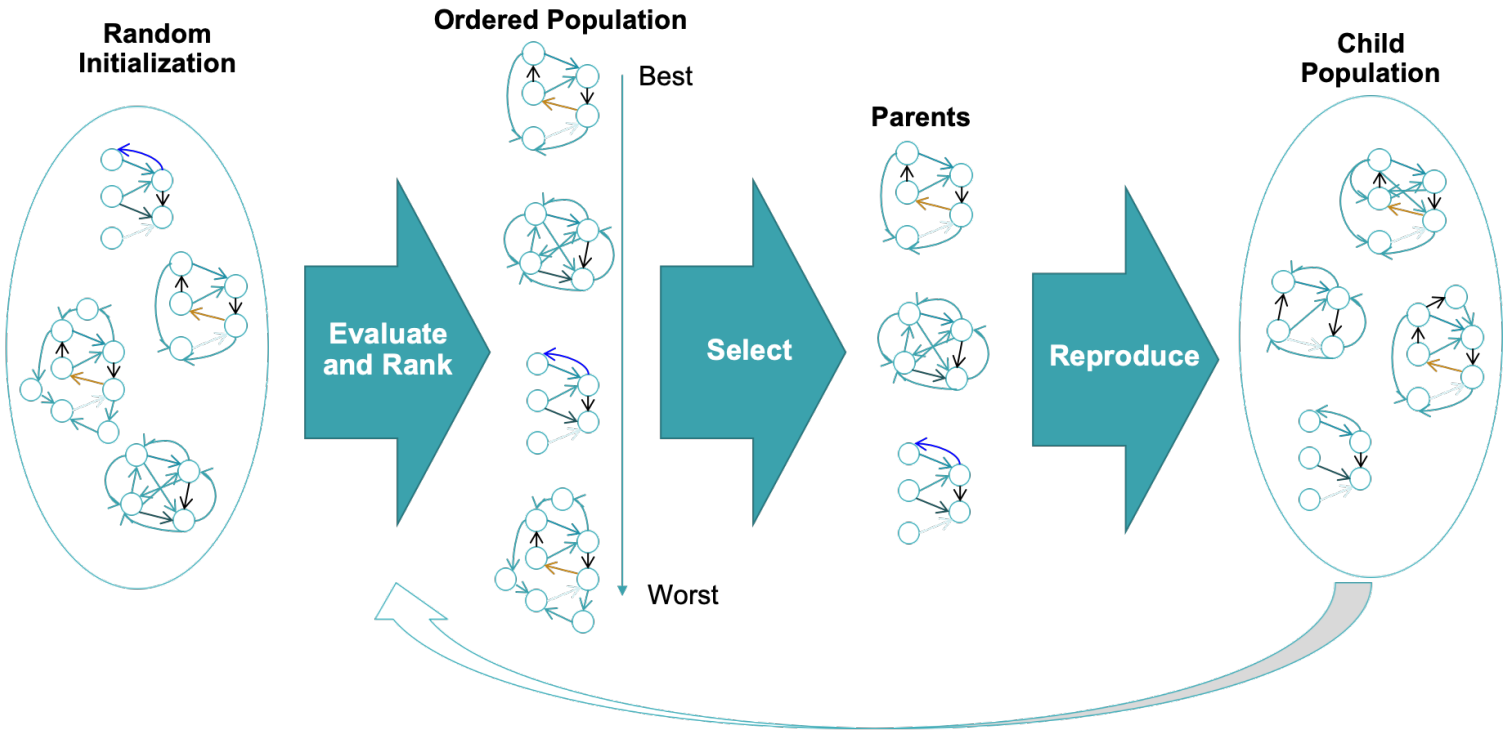
- Requires on-chip implementation of plasticity
  - Precise plasticity mechanisms are not well understood
  - Implementations such as Intel's Loihi are now including programmable plasticity
- Network structures are not well understood
  - May require dynamic adaptation of structure





Algorithms: Evolutionary Optimization

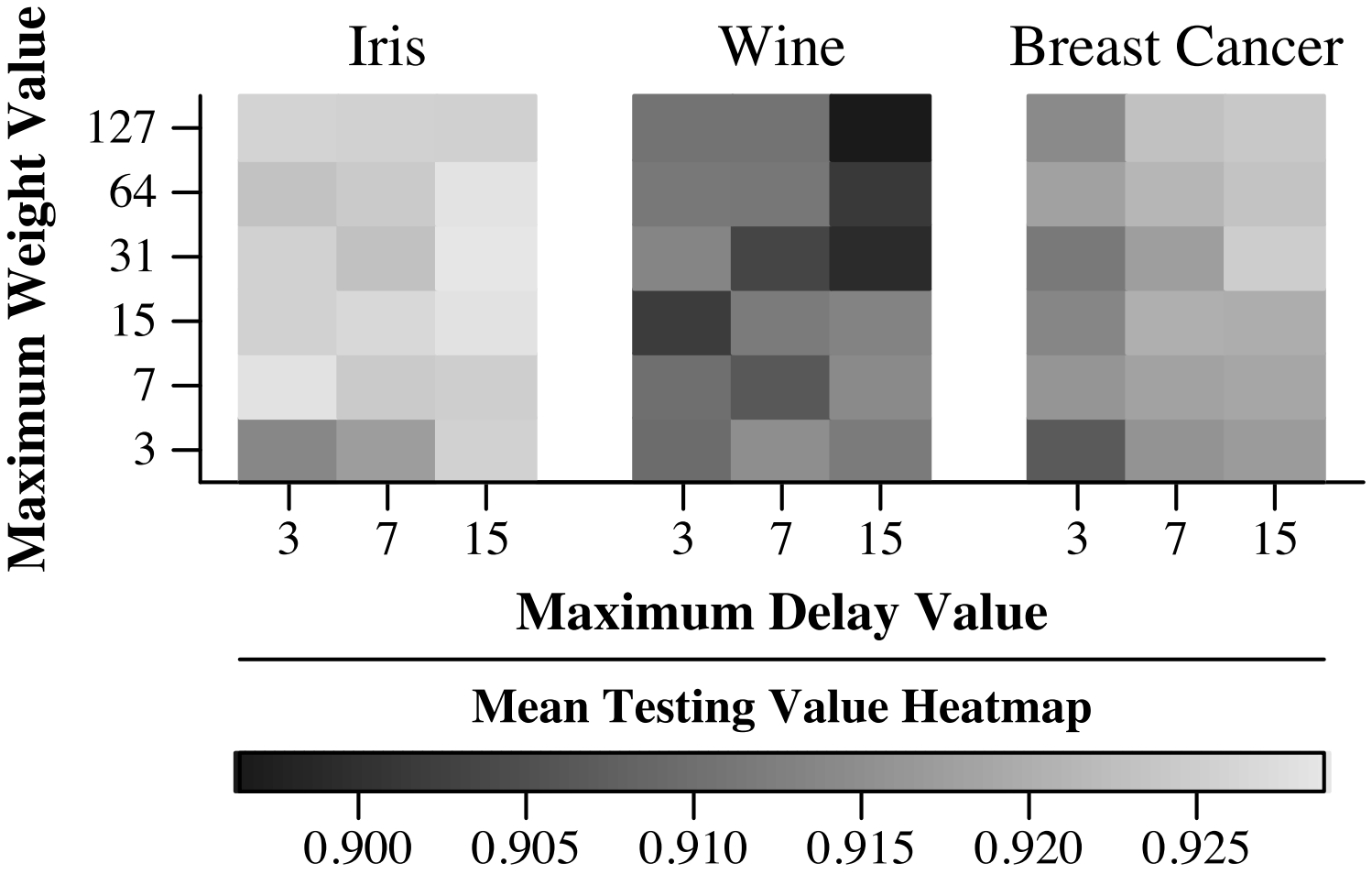
- Often produces networks with very sparse connectivity
- Can utilize chip-in-the-loop without algorithm adaptation
- Will attempt to optimize within the characteristics of the device
- Can train for low precision
- Can be slow to train



Schuman, Catherine D., et al. "An evolutionary optimization framework for neural networks and neuromorphic architectures." *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016.

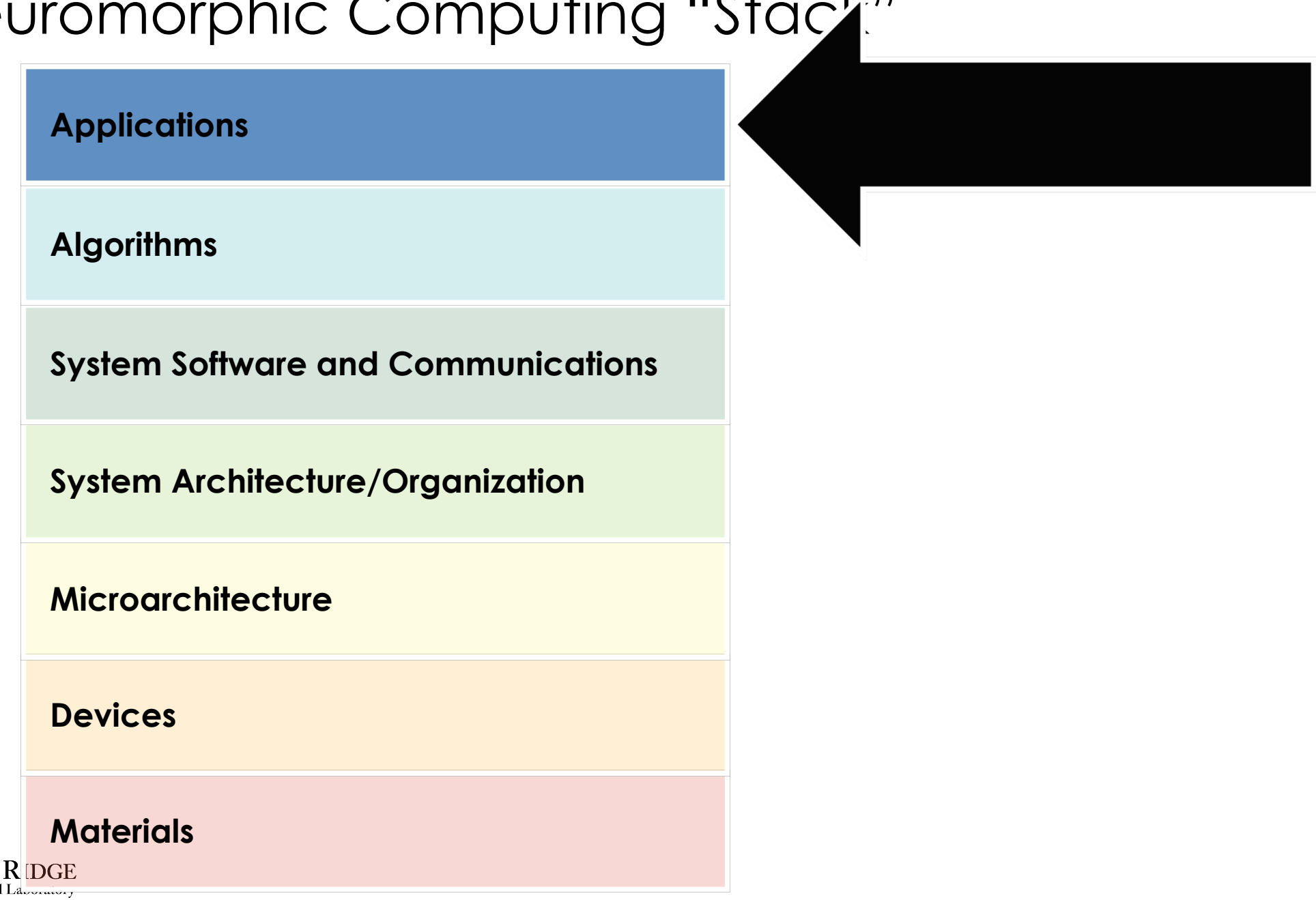
Optimizing within Hardware Constraints

- EONS (an evolutionary optimization approach) can discover comparable performing networks within hardware constraints, such as weight precision constraints
- It will also optimize within the capabilities of the device (such as the biomimetic memristor shown previously)



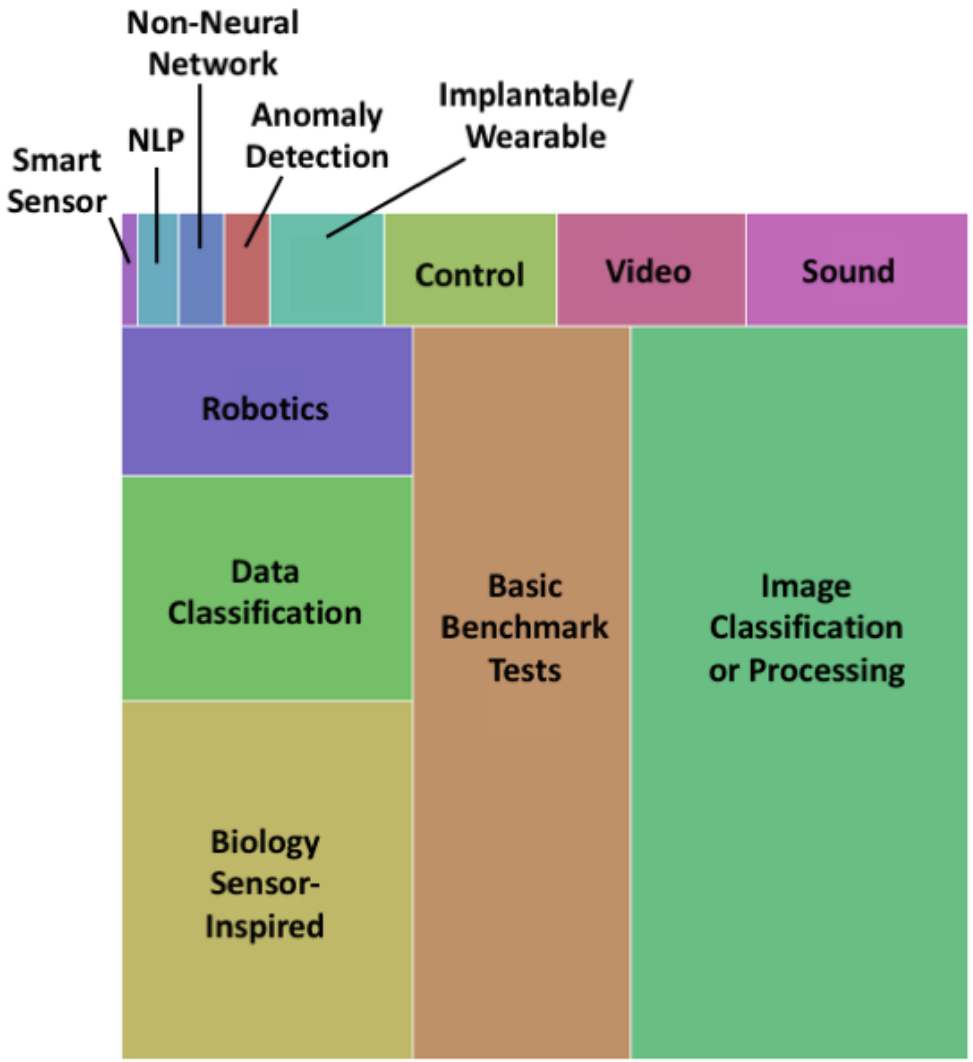
Catherine D. Schuman, J. Parker Mitchell, Robert M. Patton, Thomas E. Potokand James S. Plank. 2020. Evolutionary Optimization for Neuromorphic Systems. In Neuro-inspired Computational Elements Workshop (NICE '20),March 17–20, 2020, Heidelberg, Germany.ACM, New York, NY, USA, 9 pages.

Neuromorphic Computing "Stack"



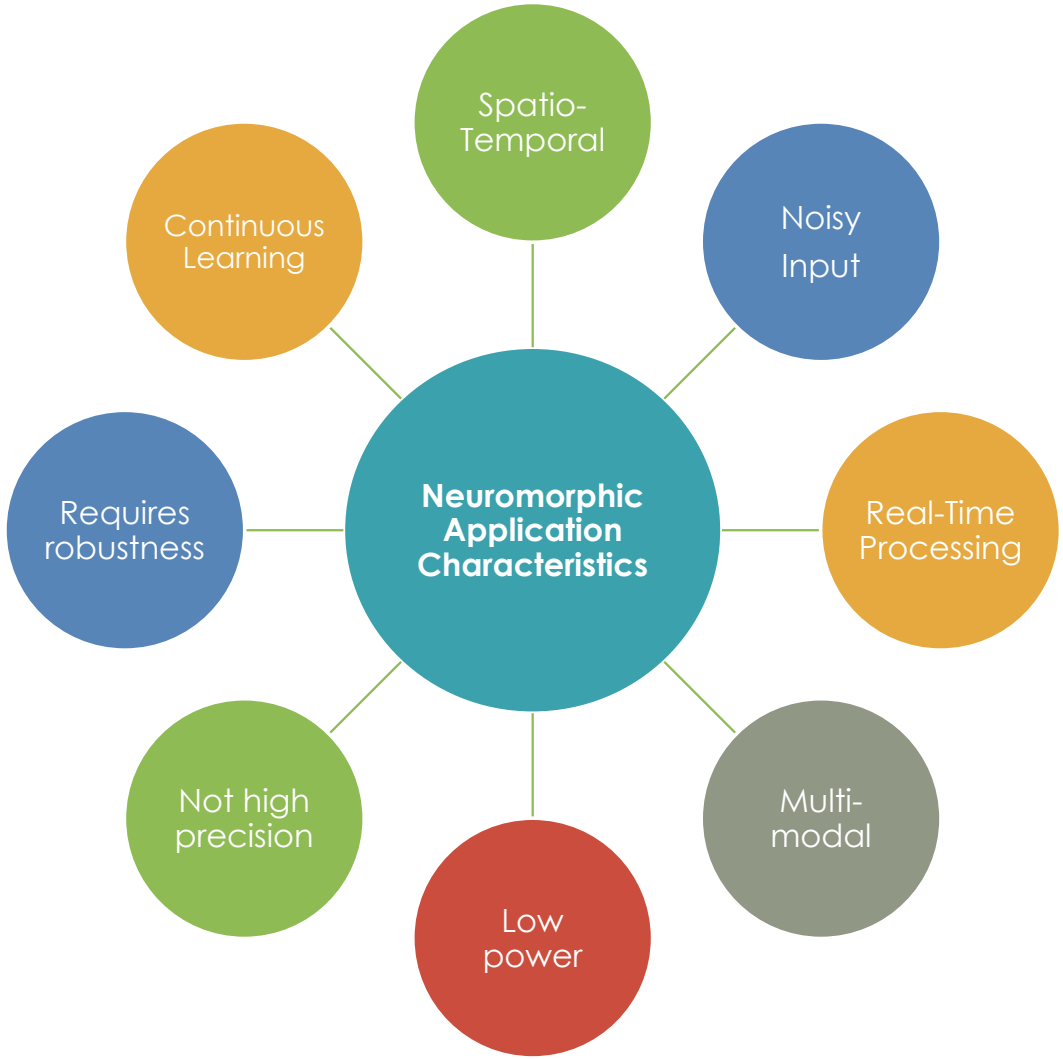
Applications of Memristive Neuromorphic Hardware

- Key Considerations
  - Low power
  - Small/embeddable
  - Processing speed required
  - Robustness and resilience
  - Integration with sensors and other compute
  - On-chip training or learning

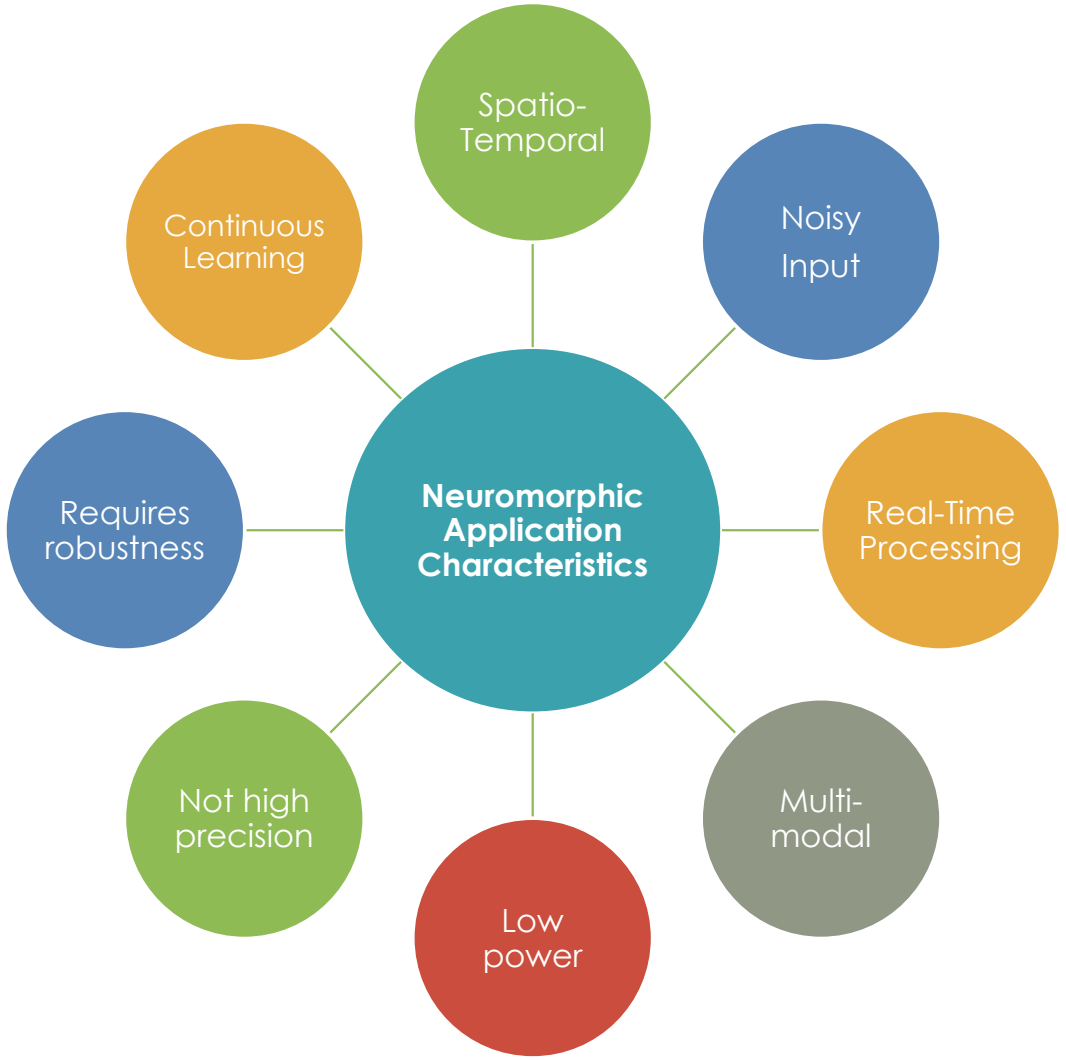


Schuman, Catherine D., et al. "A survey of neuromorphic computing and neural networks in hardware." *arXiv preprint arXiv:1705.06963* (2017).

Applications of Neuromorphic Computing

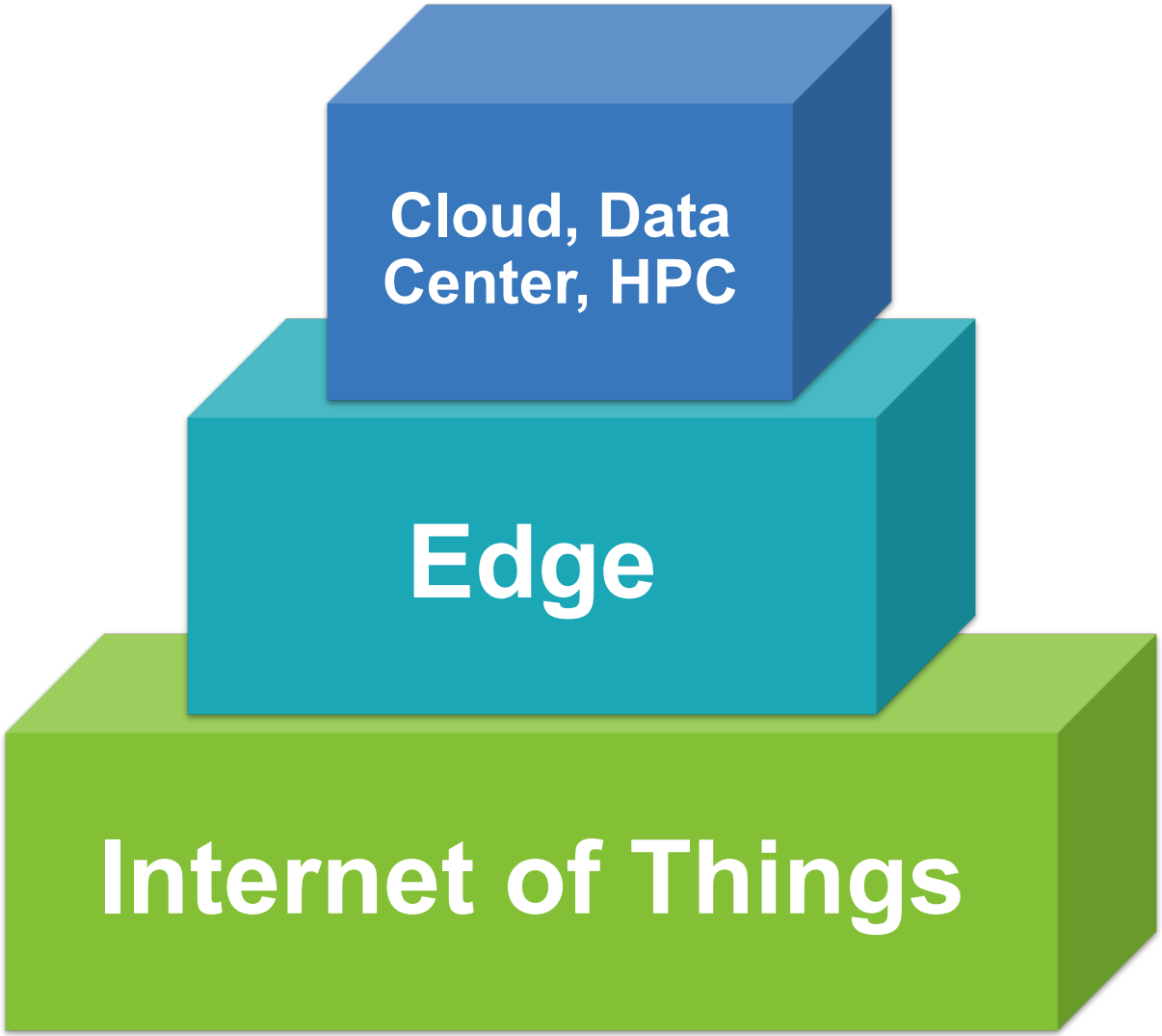


Applications of Neuromorphic Computing



- Scientific discovery
- Co-processor
- Large-scale data analytics
- Cyber security
- Autonomous vehicles
- Robotics
- Internet of things
- Smart sensors

Applications of Neuromorphic Computing

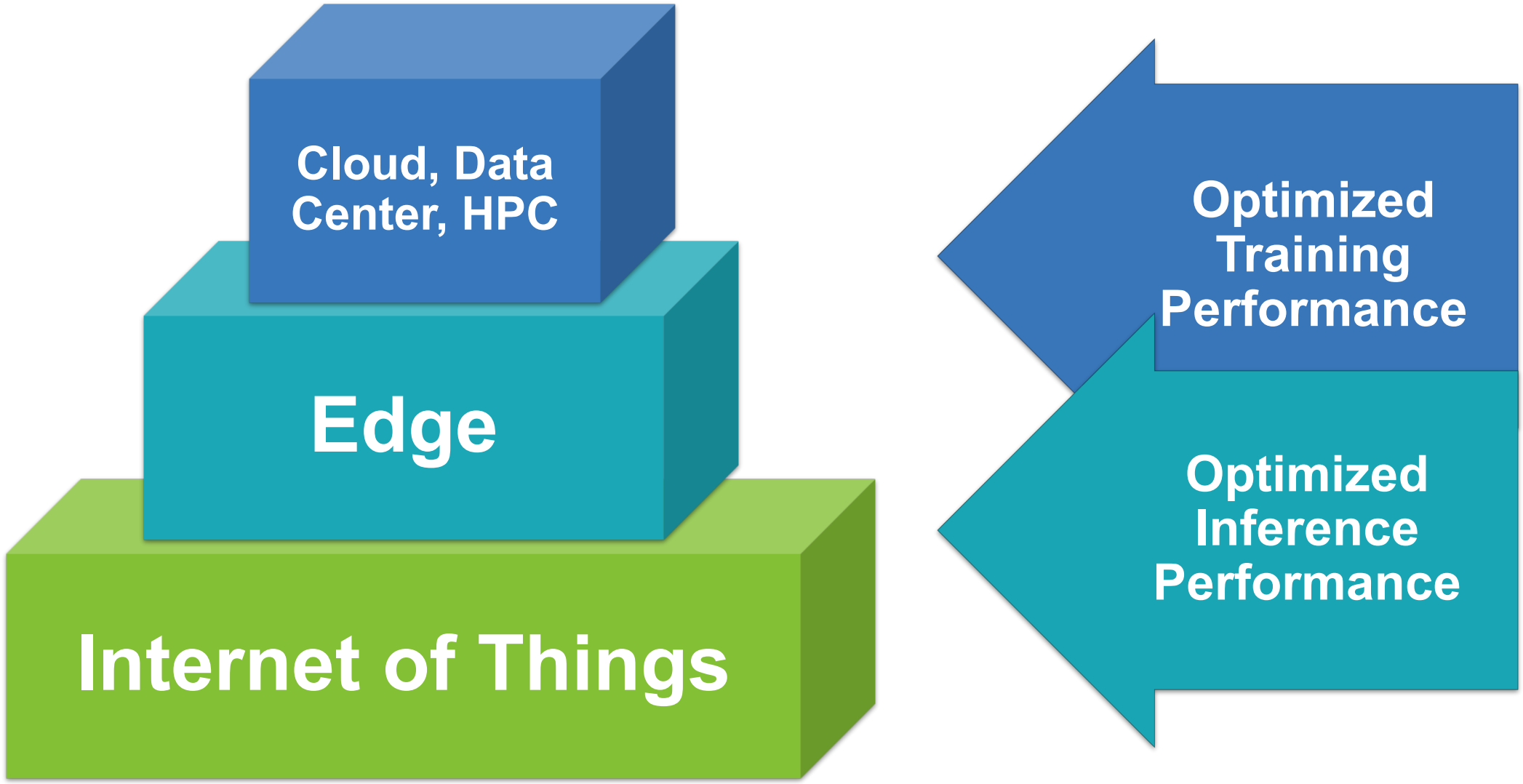


- Large-scale models
- Faster than real-time processing
- Fast re-training

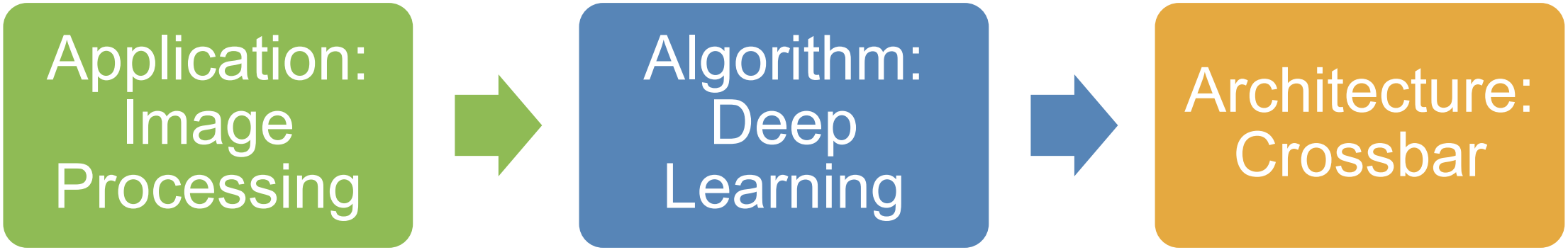
- Multi-modal
- Continuous ”local” learning

- Extremely low power
- Resilient/robust
- Embedded

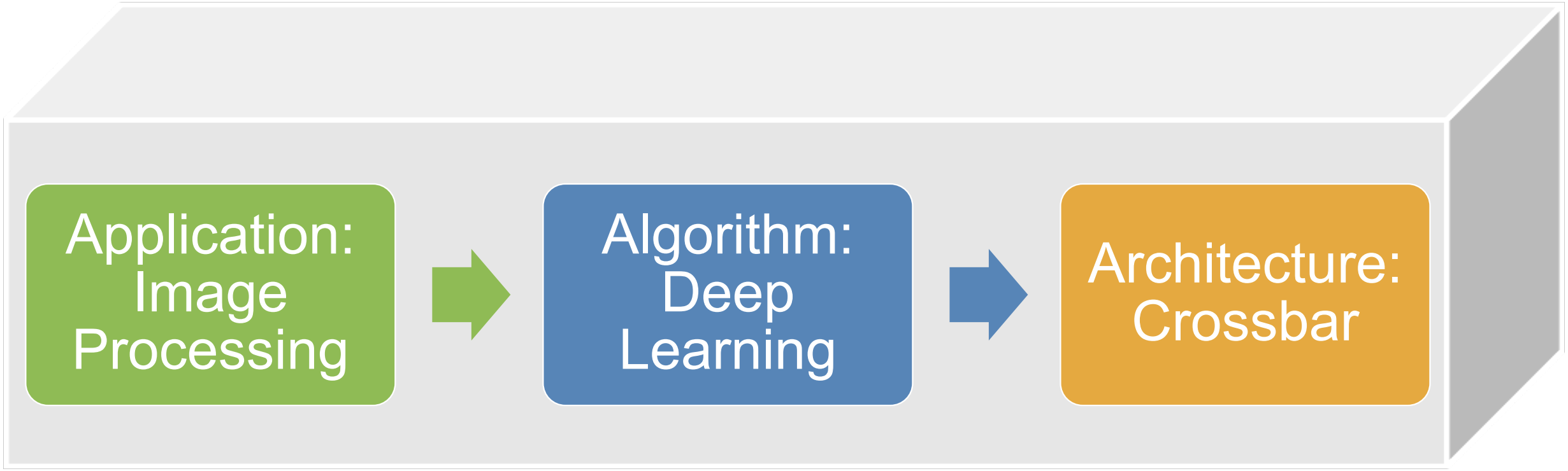
Applications of Neuromorphic Computing



Application-Driven Hardware

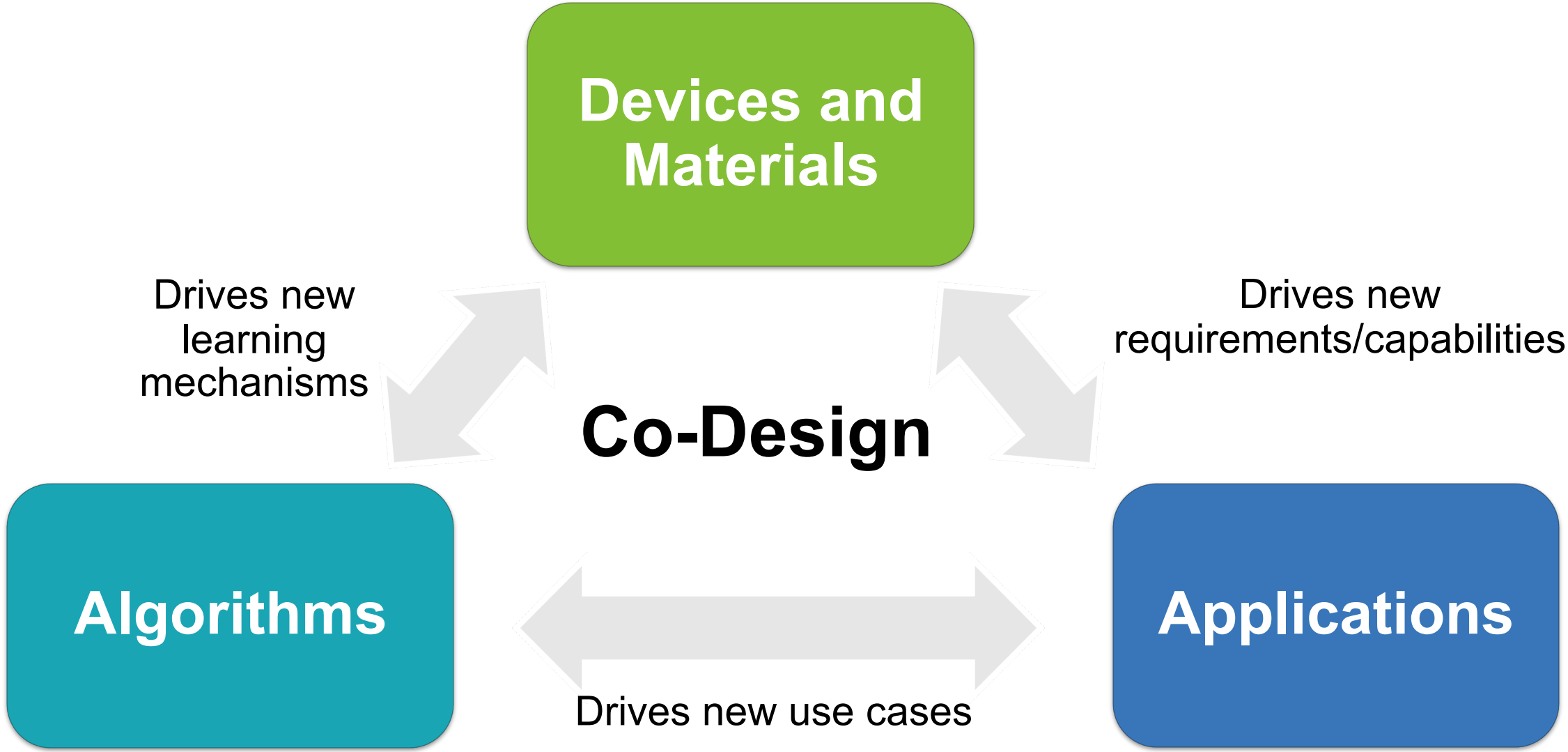


Potential Pitfall



Are we boxing ourselves into a particular use case and limiting innovation?

Opportunity for Co-Design



Key Challenge in Neuromorphic Computing

Neuromorphic Computing Devices and Materials Researchers

Know how to build novel neuromorphic implementations.

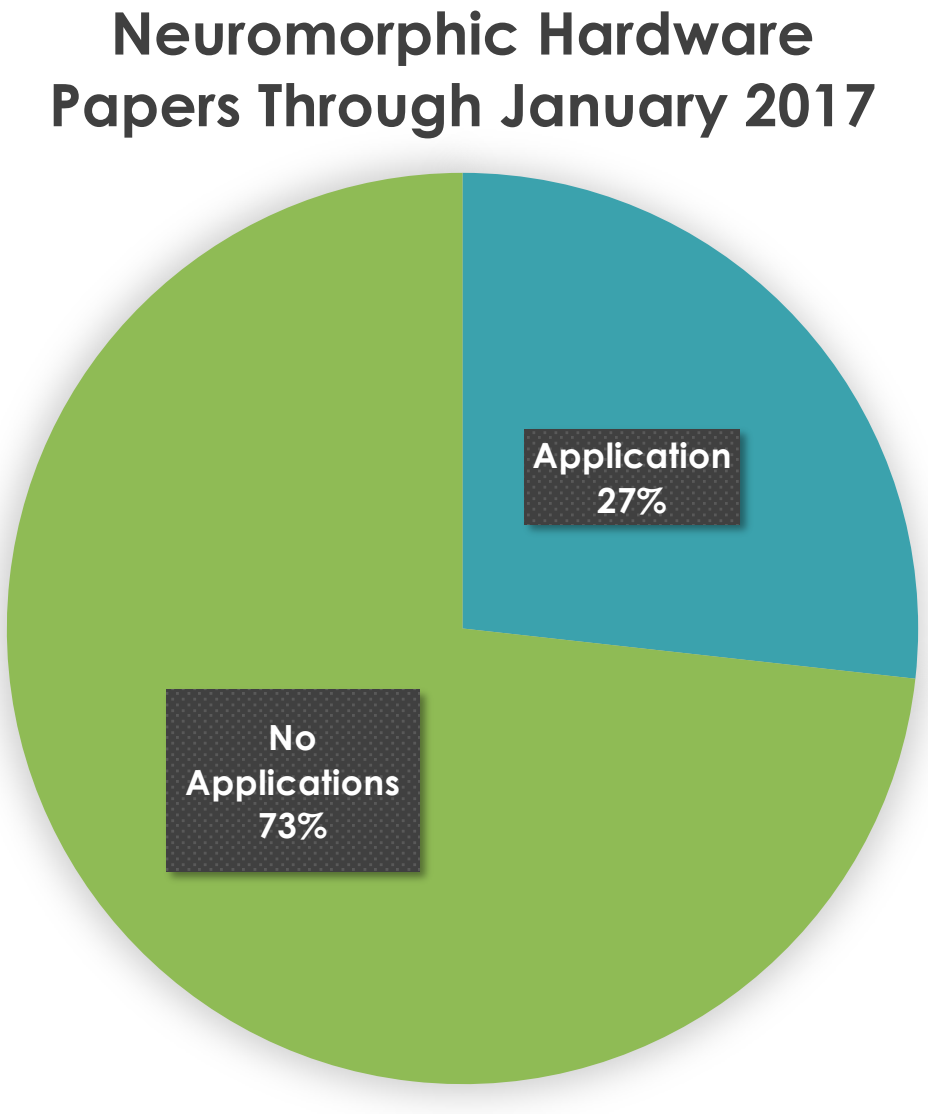
Don’t always know how to best use the implementations to do something interesting.

Key Challenge in Neuromorphic Computing

Neuromorphic Computing Devices and Materials Researchers

Know how to build novel neuromorphic implementations.

Don't always know how to best use the implementations to do something interesting.



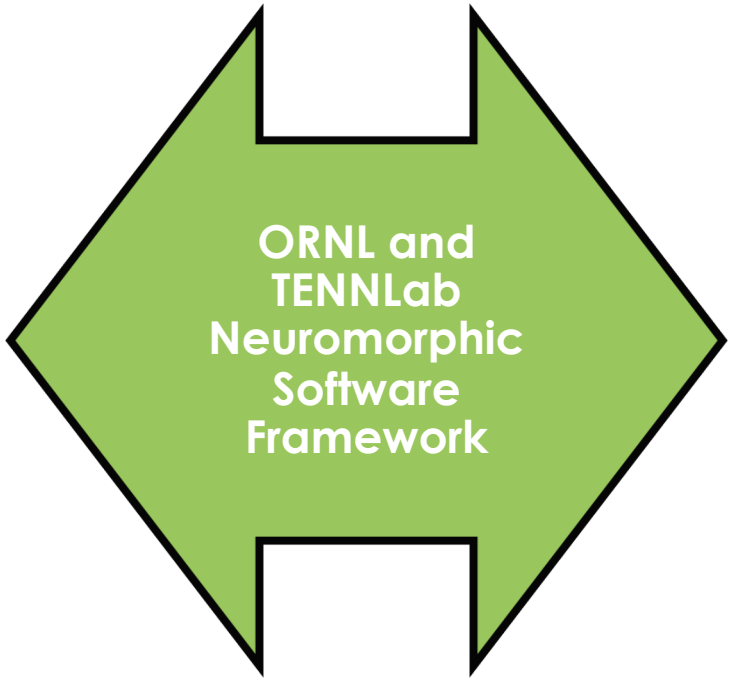
Source: Schuman, Catherine D., et al. "A survey of neuromorphic computing and neural networks in hardware." *arXiv preprint arXiv:1705.06963* (2017).

Key Challenge in Neuromorphic Computing

Neuromorphic Computing Devices and Materials Researchers

Know how to build novel neuromorphic implementations.

Don't always know how to best use the implementations to do something interesting.



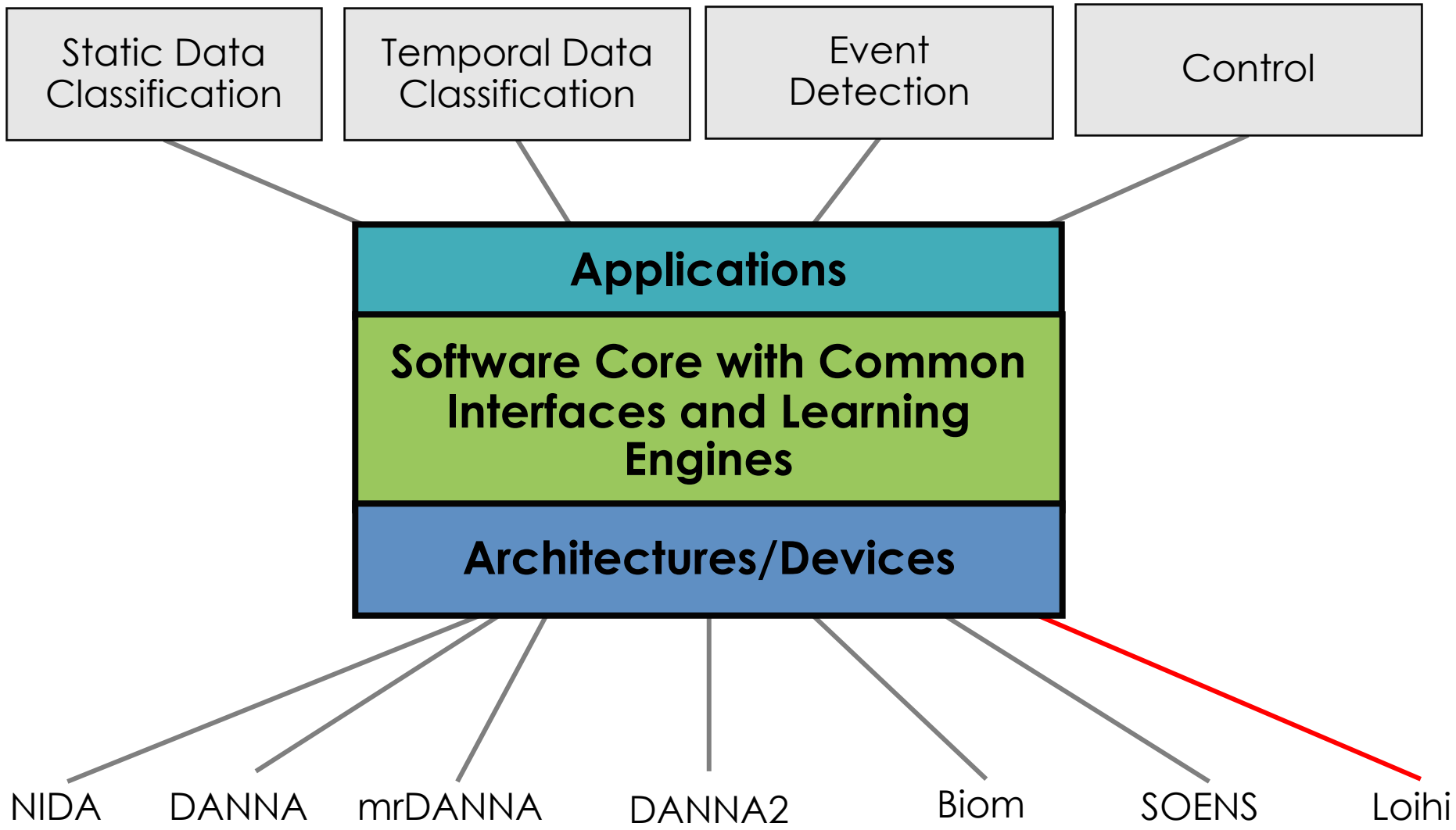
Applications Researchers

Have interesting problems to solve and know that their application can benefit from neuromorphic computing.

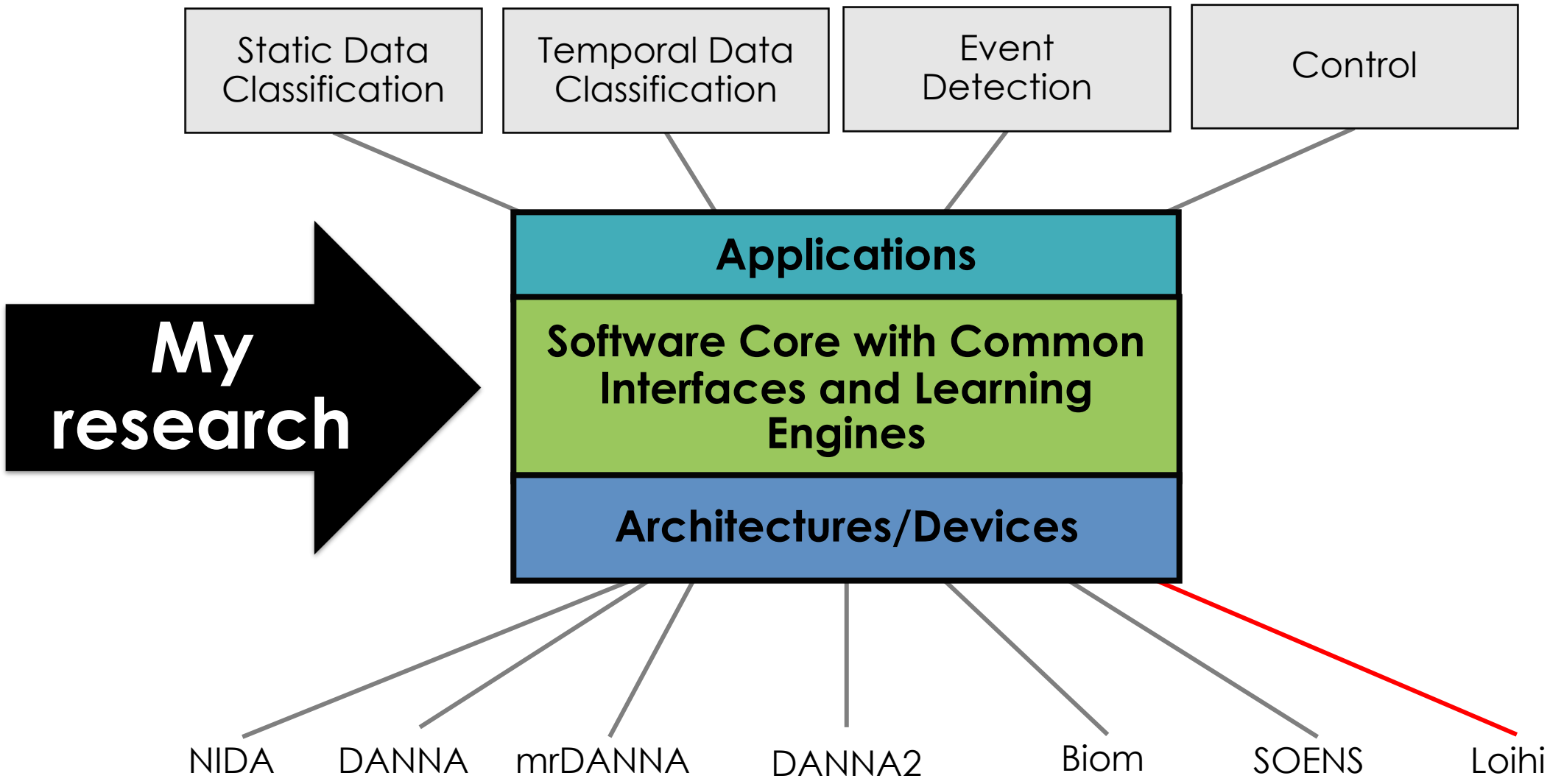
Don't always know how to build the appropriate "program" for a given neuromorphic computer or the appropriate platform to use.

Goal: Bridge the gap between materials/device/architectures researchers and applications researchers to enable cutting edge research for both

TENNLab Software Framework



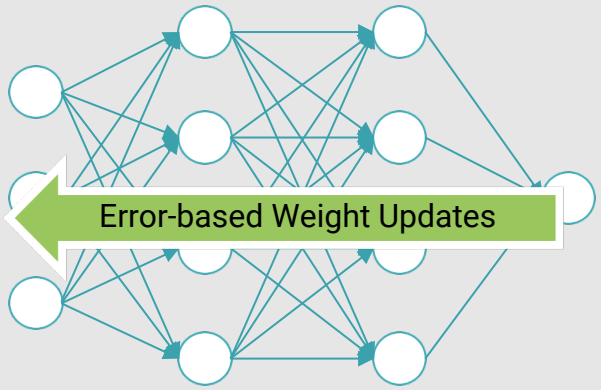
TENNLab Software Framework



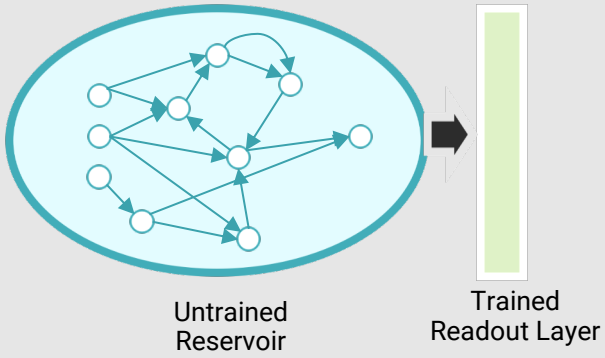
Current Approaches: Machine Learning to Build the Program

Use machine learning to define the *neuromorphic network*: the program that is loaded onto a neuromorphic computer

Back-Propagation-Like Approaches

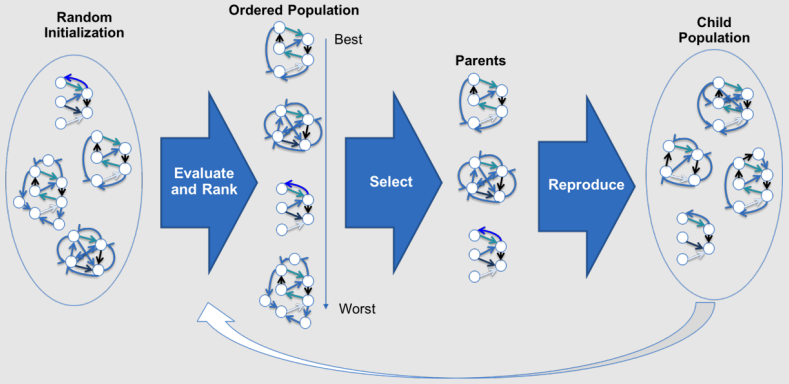


Liquid State Machines



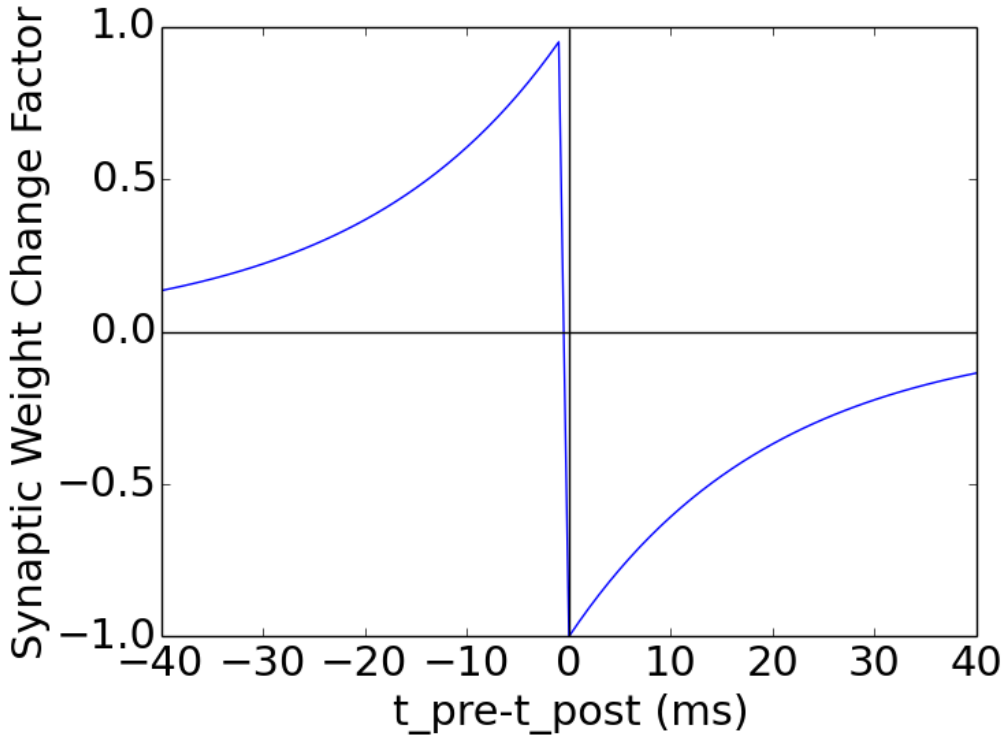
J. J. M. Reynolds, J. S. Plank, C. D. Schuman. "Intelligent Reservoir Generation for Liquid State Machines using Evolutionary Optimization." *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019.

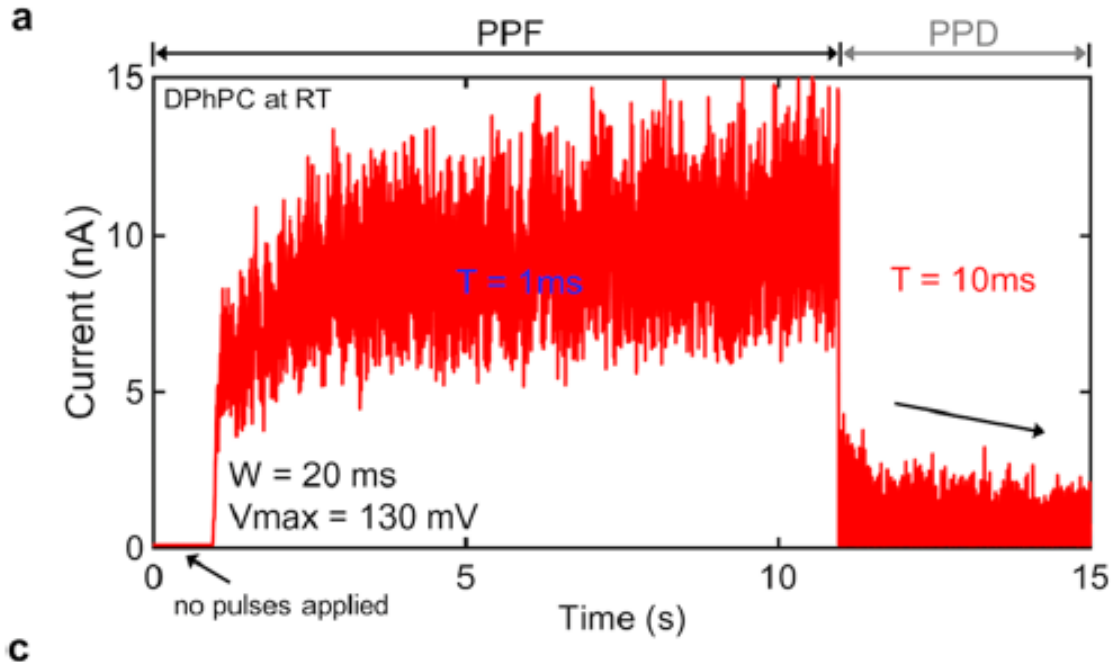
Evolutionary Optimization



C.D. Schuman, et al. "An evolutionary optimization framework for neural networks and neuromorphic architectures." *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016.

Current Approaches: Neuroscience-Inspired Plasticity



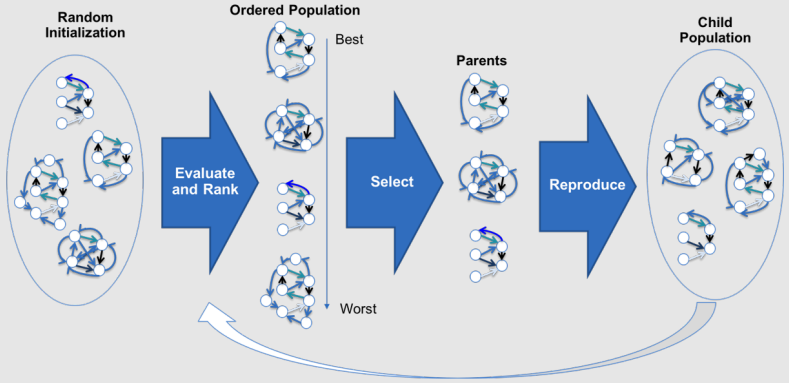


We don't really understand how these work or how to use them effectively in neuromorphic

Schuman, Catherine D. "The effect of biologically-inspired mechanisms in spiking neural networks for neuromorphic implementation." *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017.

Learning to Learn: Designing Novel Neuromorphic Algorithms with Machine Learning

Machine Learning Approaches



High Performance Computing

Image Source: https://www.olcf.ornl.gov/

Scientific Data Applications

Image Source: https://neutrons.ornl.gov/sns

New, On-Chip *Unsupervised Learning* Approaches for Neuromorphic Systems

Summary and Take Homes

- There is an opportunity to allow for materials research to inform the development of new algorithms
  - Using whatever “plasticity” is native to the device could influence algorithmic development
- We don’t yet know what the right algorithms are
  - There is a need to develop hardware that enables the development and evaluation of new algorithms
  - GPUs enabled much of the deep learning revolution. Neuromorphic hardware has the potential to enable a spiking neural network revolution
- Hardware design should take into account the final application needs
  - Different application and use cases have radically different needs

International Conference on Neuromorphic Systems

- Join us at ICONS!
- July 28-30, 2020 in Chicago, Illinois
- Submission Deadline:
  March 31, 2020
- Highlights from ICONS 2019:
  - Attendees from academia, industry, and government
  - International participation, from Europe, Asia, and Australia/New Zealand
  - Intel tutorial on Loihi

Website: icons.ornl.gov










neuromorphic.eecs.utk.edu

Work supported by:

Department of Energy

Oak Ridge National Laboratory

Air Force Research Laboratory

National Science Foundation

Thank you!

Questions?

Contact:
Email: schumancd@ornl.gov

Website: catherineschuman.com

Twitter: @cdschuman