IS NEUROMORPHIC COMPUTING READY FOR PRIME TIME?

Brad Aimone

Center for Computing Research

Sandia National Laboratories

SAND2025-10422C

AI IS DISRUPTING COMPUTING IN MANY WAYS

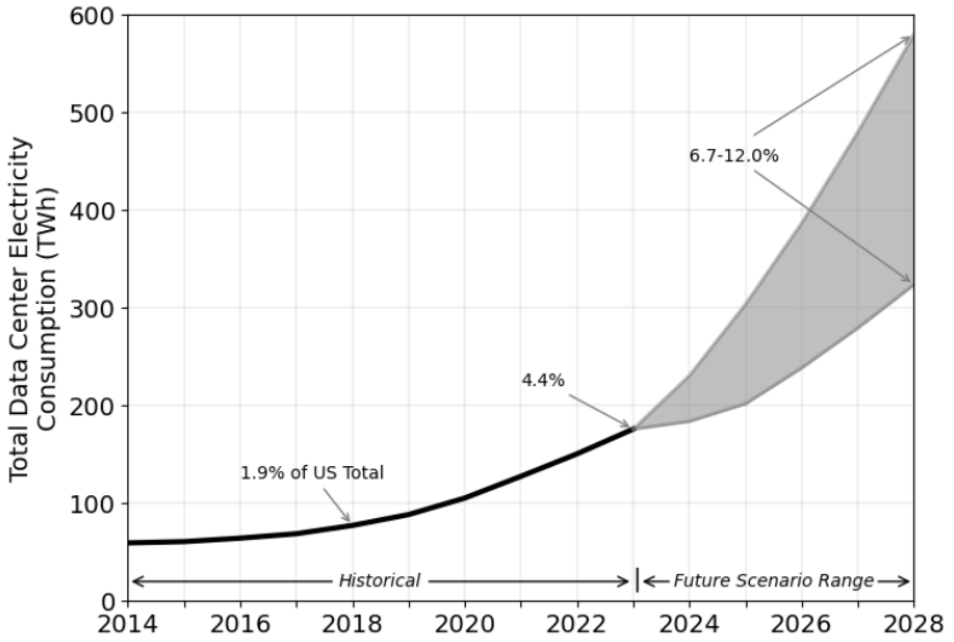
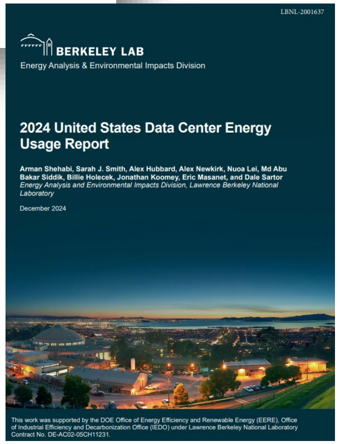


Figure ES-1. Total U.S. data center electricity use from 2014 through 2028.



LBNL-2001637

BERKELEY LAB
Energy Analysis & Environmental Impacts Division

2024 United States Data Center Energy Usage Report

Arman Shehabi, Sarah J. Smith, Alex Hubbard, Alex Newkirk, Nuoa Lei, Md Abu Bakar Siddik, Billie Holecek, Jonathan Koomey, Eric Masanet, and Dale Sartor
Energy Analysis and Environmental Impacts Division, Lawrence Berkeley National Laboratory

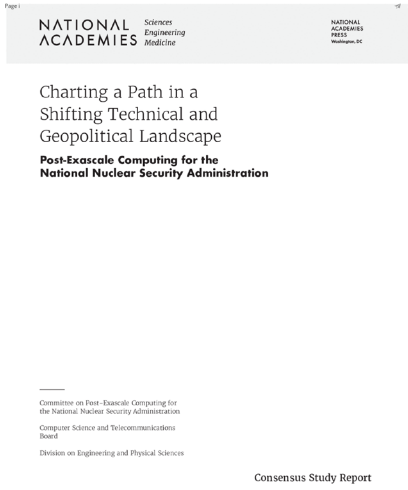
December 2024

This work was supported by the DOE Office of Energy Efficiency and Renewable Energy (EERE), Office of Industrial Efficiency and Decarbonization Office (IEDO) under Lawrence Berkeley National Laboratory Contract No. DE-AC02-05CH11231.

OVERARCHING FINDING: The combination of increasing demands for computing with the technology and market challenges in HPC requires an intentional and thorough reevaluation of ASC's approach to algorithms, software development, system design, computing platform acquisition, and workforce development. *Business-as-usual will not be adequate.*

The approach used to reach petascale and now exascale capabilities is unlikely to be sufficient for the next two decades. Instead, NNSA will need to reevaluate how its mission problems, not limited to physics simulations, are best solved through advanced computing, and rethink what type of models, algorithms, and data analysis techniques are suited to each problem; what computing capabilities will be needed; and how it can best acquire those capabilities.

Owing to a confluence of technology, marketplace, and workforce challenges, NNSA's ASC program is at a critical crossroads. The program has for decades delivered impressive and state-of-the-art predictive simulation capabilities using in-house expertise in applied mathematics, computer science, and the physical sciences, along with research and development (R&D) investments in the computer vendor community. However, the current deployment model is not likely to be sufficient for future NNSA missions.



Page i

NATIONAL ACADEMIES Sciences Engineering Medicine

NATIONAL ACADEMIES PRESS Washington, DC

Charting a Path in a Shifting Technical and Geopolitical Landscape

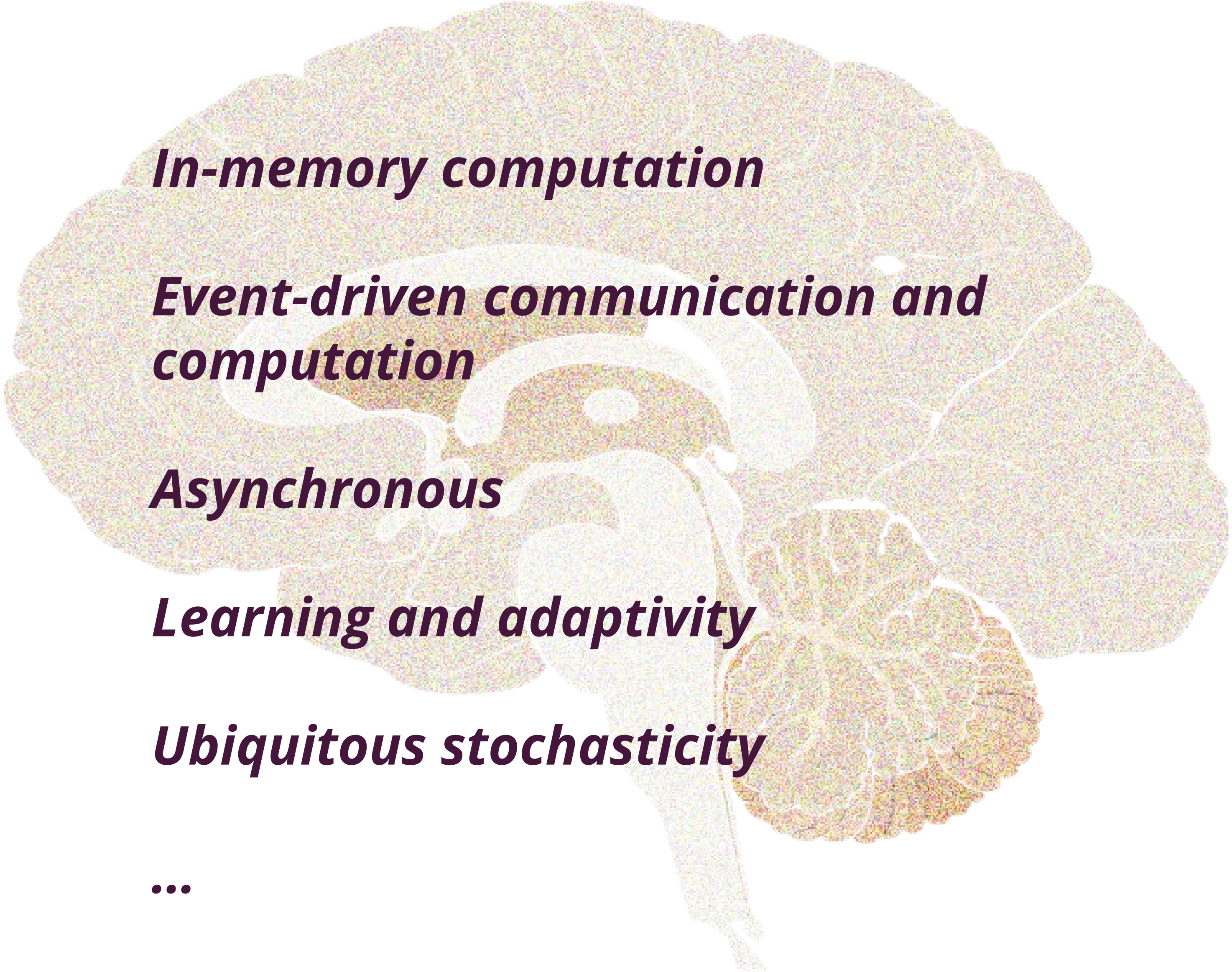
Post-Exascale Computing for the National Nuclear Security Administration

Committee on Post-Exascale Computing for the National Nuclear Security Administration

Computer Science and Telecommunications Board

Division on Engineering and Physical Sciences

Consensus Study Report



In-memory computation

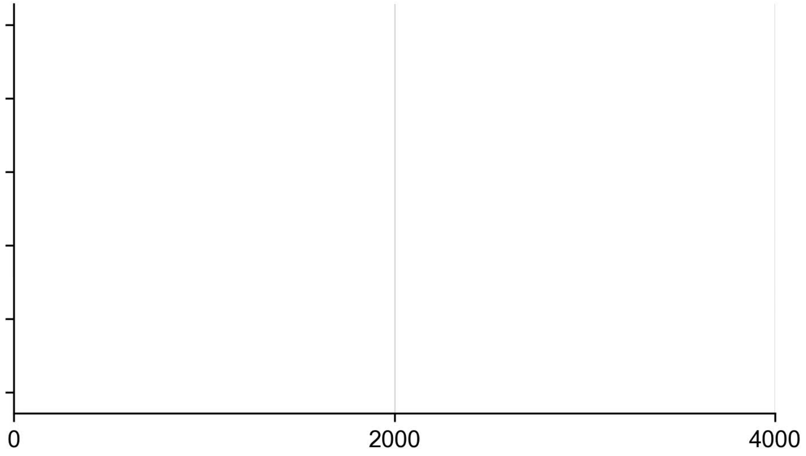
Event-driven communication and computation

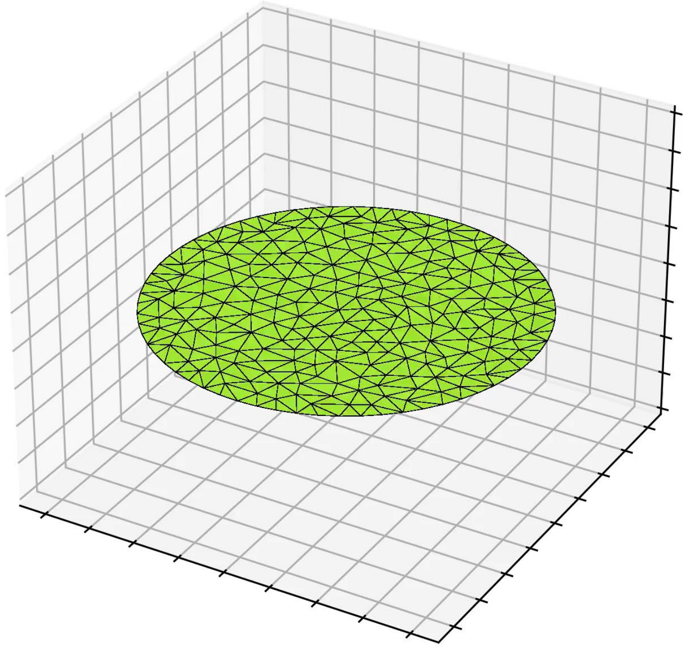
Asynchronous

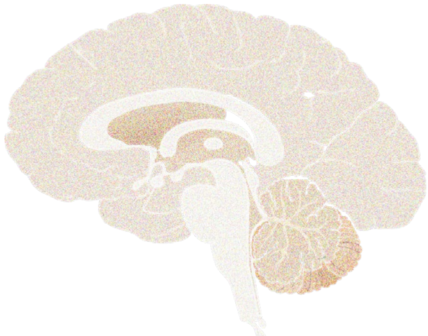
Learning and adaptivity

Ubiquitous stochasticity

…







Why you would want an architecture with brain-like activity to solve problems such as finite element simulations?

PART 1:

WHAT IS NEUROMORPHIC COMPUTING?

THERE IS NOT A SINGLE ROADMAP FOR NEUROMORPHIC COMPUTING

- Neuromorphic research exists across the technology stack
  - New post-CMOS materials (memristors, ECRAM, MTJs, quantum materials, …)
  - Non-digital devices (analog, stochastic, optical, …)
  - Bio-inspired circuits (reconfigurable, dendrites, learning, …)
  - Neuromorphic architectures (spiking, event-driven sensors, ANN accelerators, …)
  - Software paradigms (compilers, intermediate representations, …)
  - Neuromorphic algorithms

When is it ready for prime time?
What is needed to get there?

WHEN WILL NEUROMORPHIC COMPUTING BE A REALITY?

GPUs

Well described materials (i.e., silicon)

Today's fabrication technology

Demonstrated Applications

Stable programming model

Dataflow

Well-understood but not conventional materials

Some scaling risk but fabrication possible

Prototyped Applications

Developing programming model

Quantum Computing

Exotic materials

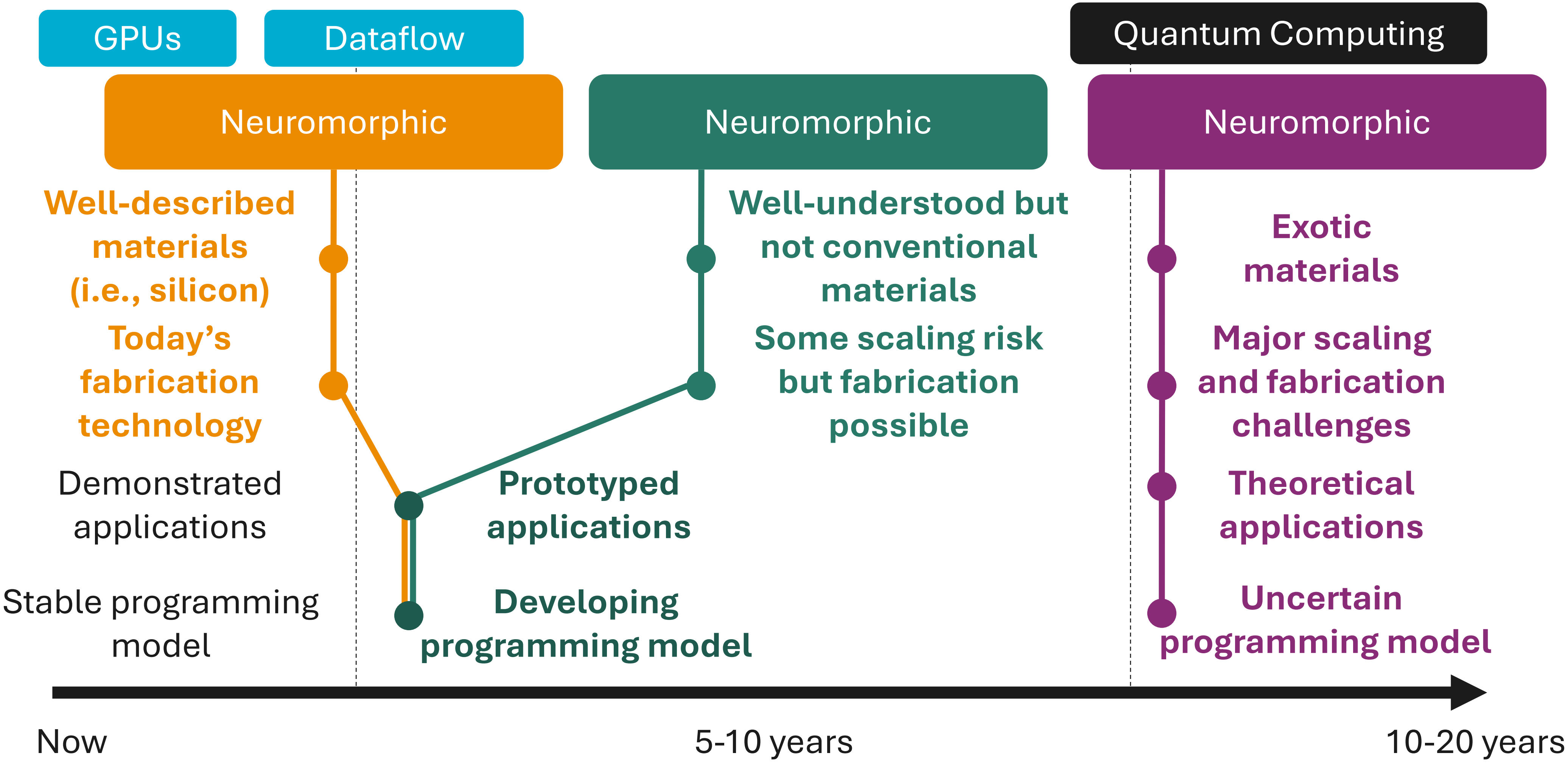
Major scaling and fabrication challenges

Theoretical Applications

Uncertain programming model

Now — 5-10 years — 10-20 years

NEUROMORPHIC COMPUTING HAS PROMISE AT DIFFERENT TIME SCALES



TODAY'S DIGITAL NEUROMORPHIC SYSTEMS ARE APPROACHING BRAIN-SCALE

- ❑ Systems like Intel's Loihi 2 and SpiNNCloud's SpiNNaker 2 can surpass 1 billion neurons
- ❑ Individual chips are ~1 million neurons and ~1 Watt
- ❑ Fully CMOS (little fabrication risk)
- ❑ Digital or Digital + Analog hybrid

- ❑ Future devices and novel materials can amplify potential impact

Review

Neuromorphic computing at scale

https://doi.org/10.1038/s41586-024-08253-8
Received: 12 June 2023
Accepted: 18 October 2024
Published online: 22 January 2025
Check for updates

Dhireesha Kudithipudi[1], Catherine Schuman[2], Craig M. Vineyard[3], Tej Pandit[1], Cory Merkel[4], Rajkumar Kubendran[5], James B. Aimone[3], Garrick Orchard[6], Christian Mayr[7], Ryad Benosman[6], Joe Hays[8], Cliff Young[9], Chiara Bartolozzi[10], Amitava Majumdar[11], Suma George Cardwell[3], Melika Payvand[12], Sonia Buckley[13], Shruti Kulkarni[14], Hector A. Gonzalez[15], Gert Cauwenberghs[11], Chetan Singh Thakur[16], Anand Subramoney[17] & Steve Furber[18]

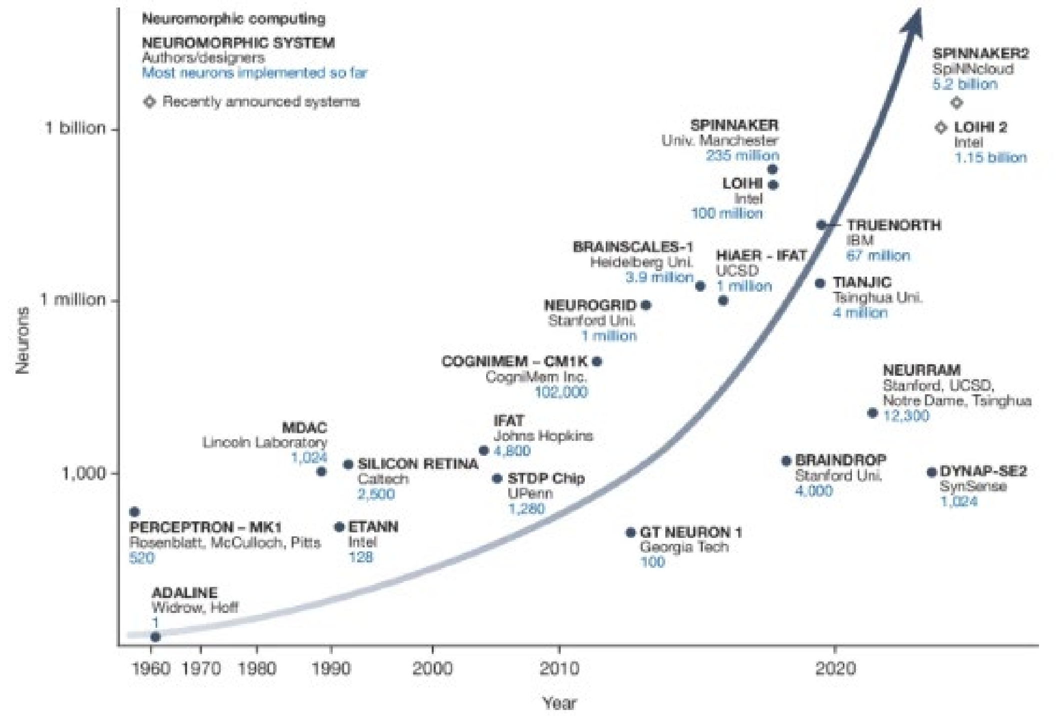
Neuromorphic computing is a brain-inspired approach to hardware and algorithm design that efficiently realizes artificial neural networks. Neuromorphic designers apply the principles of biointelligence discovered by neuroscientists to design efficient computational systems, often for applications with size, weight and power constraints. With th
course for the devel
approaches for crea
features. We discus
challenges that nee
ecosystem necessar
scaling neuromorph
fields, providing gu
computing who aim

As neural networks continue to affect a growing range of applications and further advances are sought, human brains remain a vibrant source of inspiration for modelling rich computational prowess. However, the pursuit of brain-inspired machine intelligence will require a change in the way we design and build computational platforms. One of the most promising research efforts in this direction is neuromorphic computing—a brain-inspired approach to hardware and algorithm design that efficiently realizes artificial neural networks[1]. Neuromorphic computing designers apply the principles of biointelligence discovered by neuroscientists to design efficient computational systems, often for applications with size, weight and power constraints.

Extrapolation from the recent rate of progress in prototype neuromorphic systems suggests an enormous potential for future artificial intelligence (AI) applications: the market for neuromorphic computing chips is expected to reach US$556.6 million by 2026 (ref. 2). Some neuromorphic chips are rapidly entering the early-stage commercial market and have demonstrated capabilities to solve computational tasks at varying scales, with extremely low power budget and latency[3,4]. One reason for such an explosion is that these systems are versatile. For example, advances in traditional computing are often focused on a specific class of architecture—exascale for supercomputers or small scale for embedded systems—and the same class is typically not explored to influence both. However, neuromorphic computing has the potential to be disruptive in both classes by using homogeneous computing technology throughout. The question of whether the field is ready to enable substantial computational breakthroughs, such as the 'AlexNet moment' described in Box 1, and how to comprehend the

[1]University of Texas at San Antonio, San Antonio, TX, USA. [2]University of Tennessee, Knoxville
Technology, Rochester, NY, USA. [5]University of Pittsburgh, Pittsburgh, PA, USA. [6]Intel Labs, San
Laboratory, Washington, DC, USA. [9]Google DeepMind, Mountain View, CA, USA. [10]Italian Institu
of Neuroinformatics, University of Zürich and ETH Zürich, Zürich, Switzerland. [13]National Institu
TN, USA. [15]SpiNNcloud Systems GmbH, Dresden, Germany. [16]Indian Institute of Science, Benga
Manchester, UK. e-mail: dhireesha.kudithipudi@utsa.edu

Fig. 1: Progression of neuromorphic computing systems.



Kudithipudi et al., Nature 2025

NOT QUITE A MOLE OF NEURONS

… BUT MORE NEURONS THAN A MOLE

WHAT IS NEUROMORPHIC COMPUTING TODAY?

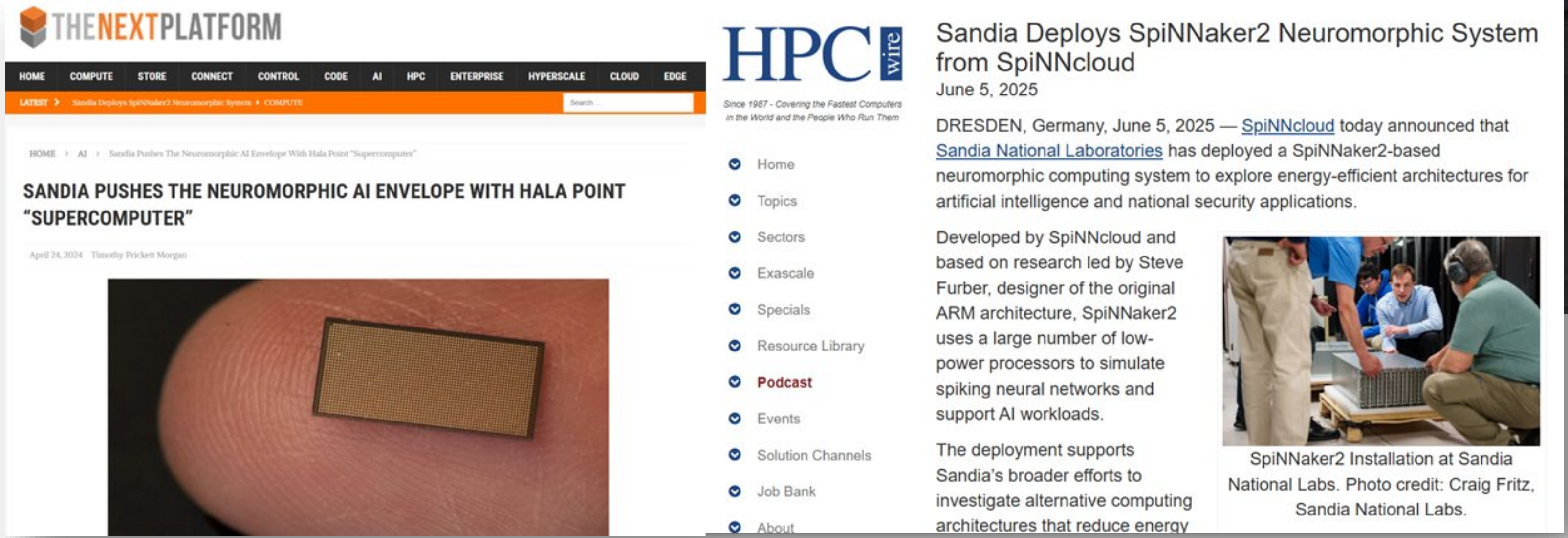
- In digital silicon (CMOS) technology
  - Over 1 billion neuron system at Sandia
    - Roughly the number in a parrot or small primate brain
  - Neurons are "simulated" in an efficient way
  - Generally a leaky integrate-and-fire model

- Analog systems
  - Wide range of technologies, but far smaller
  - Emulate the brain's biophysics in different materials
  - Wide range of neural dynamics emulated

Left to right: William Chapman, Brad Theilman, Craig Vineyard, Mark Plagge

WHAT IS NEUROMORPHIC COMPUTING TODAY?

- In digital silicon (CMOS) technology
  - Over 1 billion neuron system at Sandia
    - Roughly the number in a parrot or small primate brain
  - Neurons are "simulated" in an efficient way
  - Generally a leaky integrate-and-fire model


THENEXTPLATFORM

HOME COMPUTE STORE CONNECT CONTROL CODE AI HPC ENTERPRISE HYPERSCALE CLOUD EDGE

SANDIA PUSHES THE NEUROMORPHIC AI ENVELOPE WITH HALA POINT "SUPERCOMPUTER"

April 24, 2024 Timothy Prickett Morgan

HPCwire

Since 1987 - Covering the Fastest Computers in the World and the People Who Run Them

Home
Topics
Sectors
Exascale
Specials
Resource Library
Podcast
Events
Solution Channels
Job Bank
About

Sandia Deploys SpiNNaker2 Neuromorphic System from SpiNNcloud

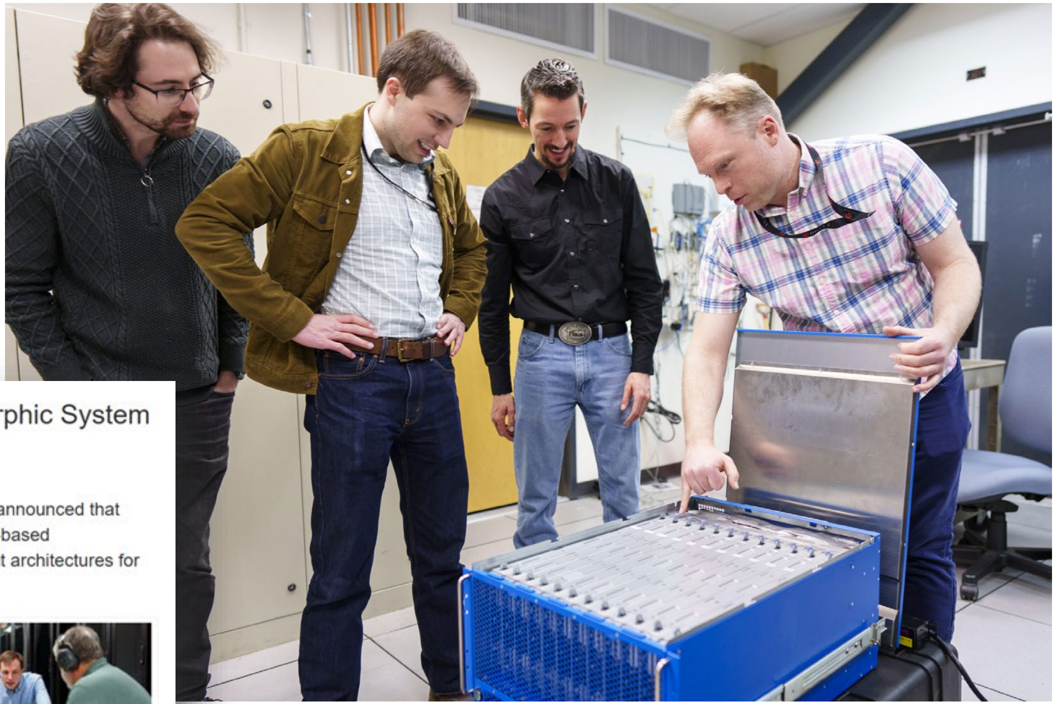
June 5, 2025

DRESDEN, Germany, June 5, 2025 — SpiNNcloud today announced that Sandia National Laboratories has deployed a SpiNNaker2-based neuromorphic computing system to explore energy-efficient architectures for artificial intelligence and national security applications.

Developed by SpiNNcloud and based on research led by Steve Furber, designer of the original ARM architecture, SpiNNaker2 uses a large number of low-power processors to simulate spiking neural networks and support AI workloads.

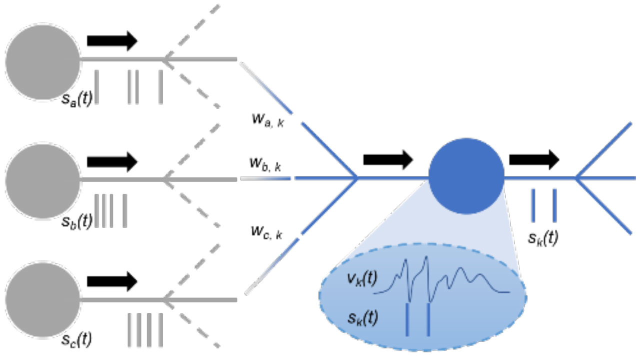
The deployment supports Sandia's broader efforts to investigate alternative computing architectures that reduce energy

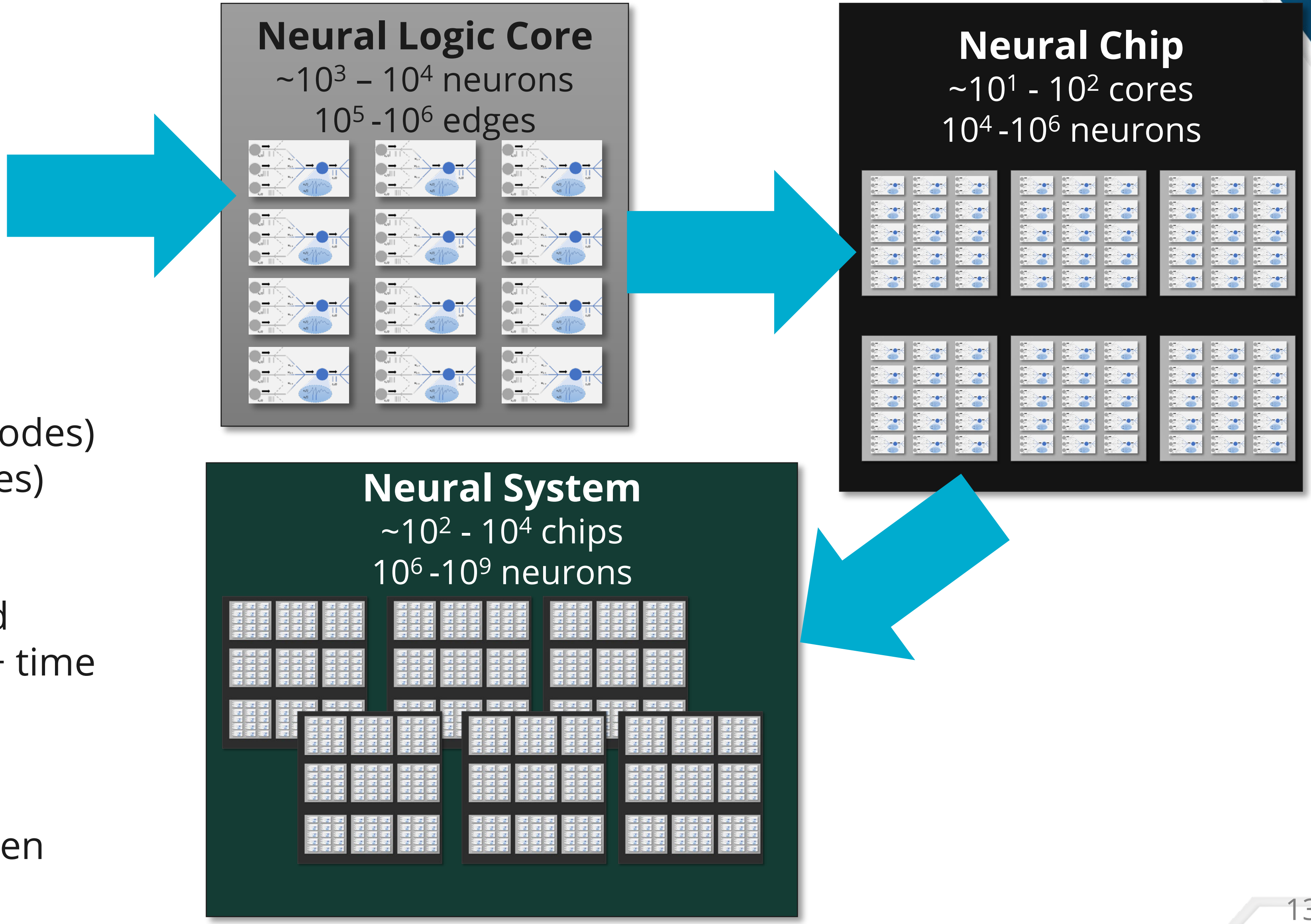
SpiNNaker2 Installation at Sandia National Labs. Photo credit: Craig Fritz, Sandia National Labs.



Left to right: William Chapman, Brad Theilman, Craig Vineyard, Mark Plagge

SPIKING NEUROMORPHIC TODAY:
SCALABLE AND PROGRAMMABLE IN-MEMORY COMPUTE





Computational Primitives:

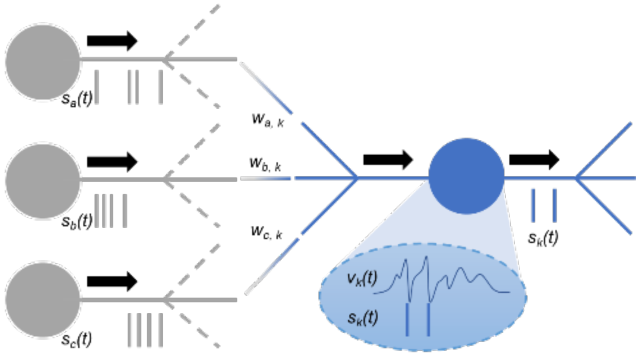
Spiking Neurons (vertices/nodes)

Synapses (connections/edges)

Programmable as arbitrary graphs

- Edges: Directed and weighted
- Nodes: Threshold gate logic + time
- *Artificial neural networks are a special case*
- Programmability, theoretical, analysis, and software are open research questions

SPIKING NEUROMORPHIC TODAY: STILL FAR FROM THE ACTUAL BRAIN



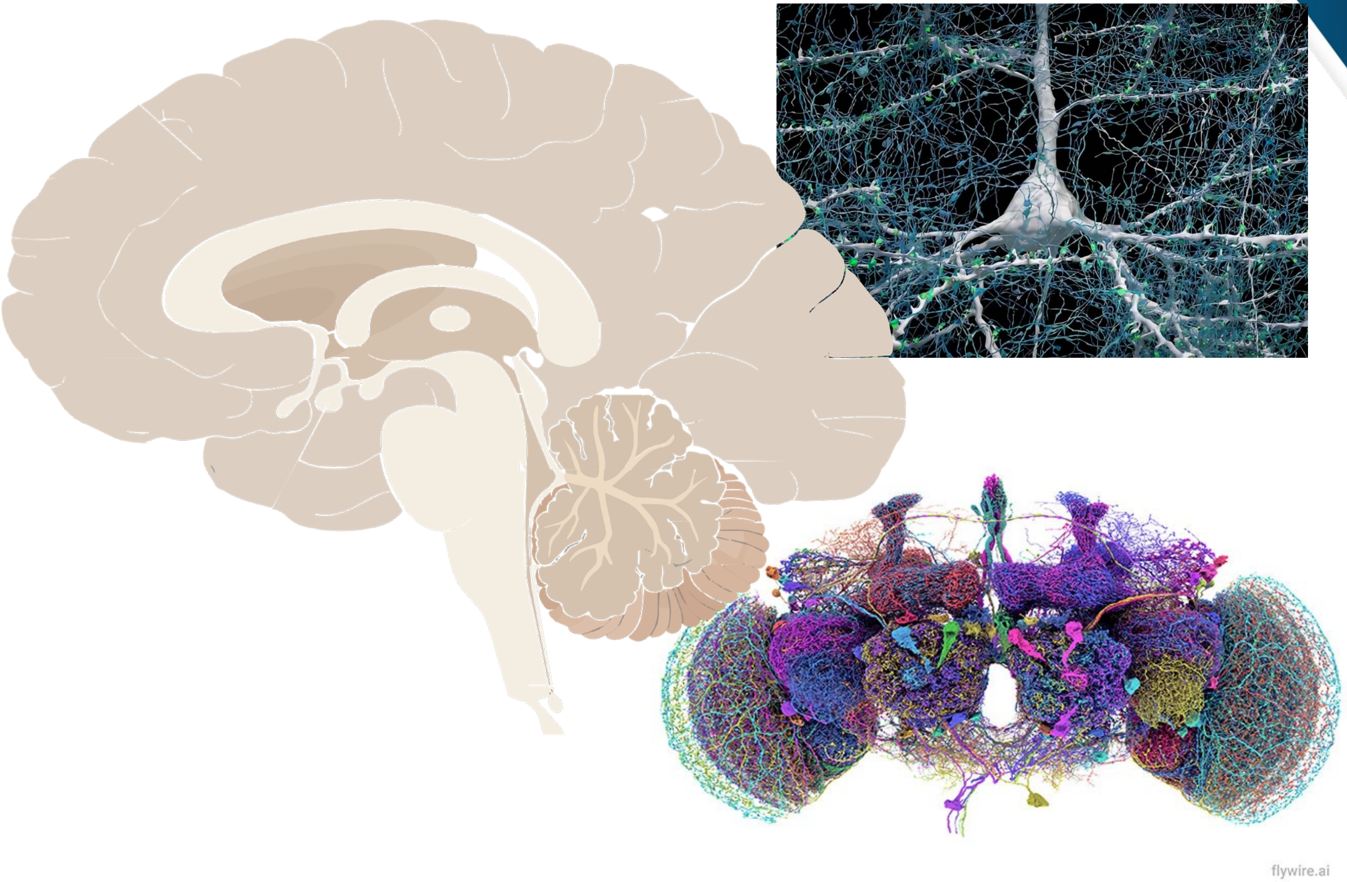
Computational Primitives:

Spiking Neurons (vertices/nodes)

Synapses (connections/edges)

Programmable as arbitrary graphs

- Edges: Directed and weighted
- Nodes: Threshold gate logic + time
- *Artificial neural networks are a special case*
- Programmability, theoretical, analysis, and software are open research questions



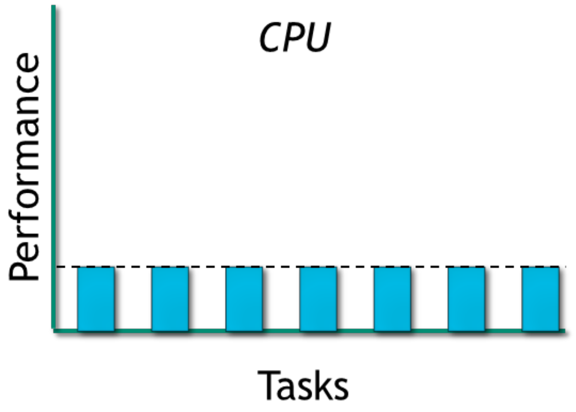
Today’s neuromorphic systems are far from the brain in terms of complexity and scale

PART 2:

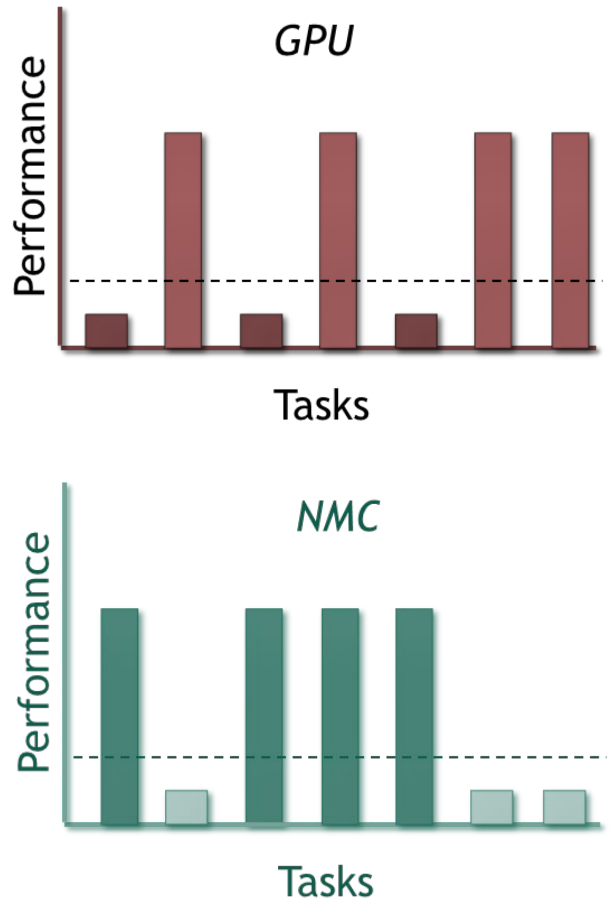
WHAT MAKES NEURAL COMPUTING DIFFERENT?

NEUROMORPHIC IS LIKELY SIMILAR TO GPUS IN DEGREE OF SPECIALIZATION

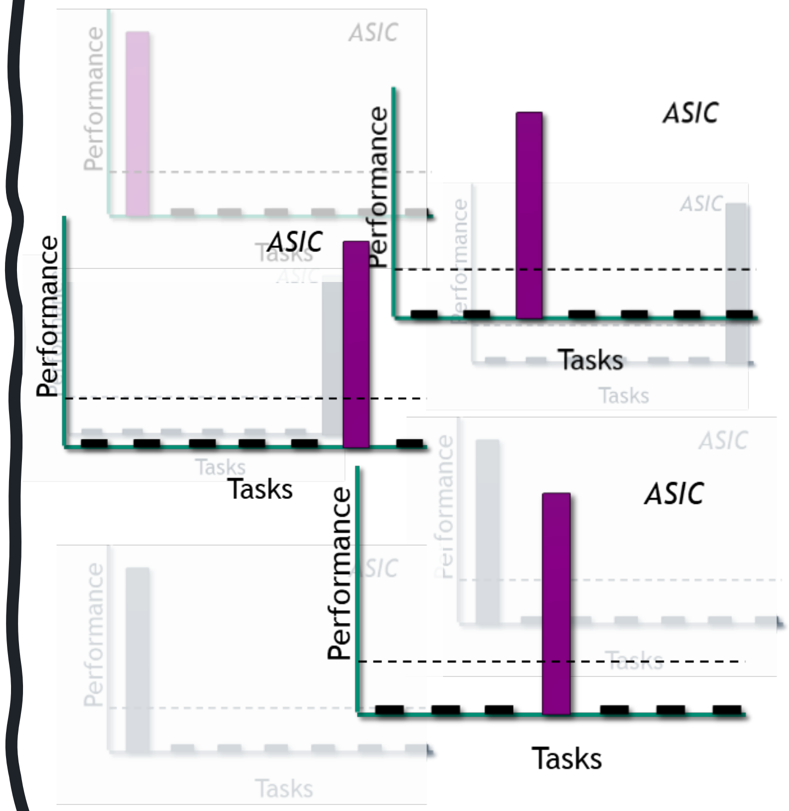
Truly General Purpose



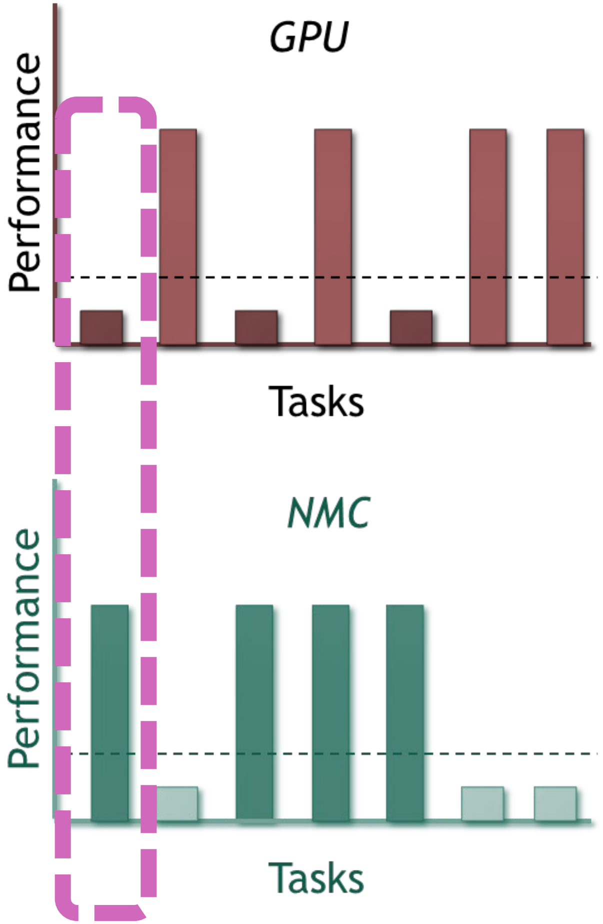
Specialized General Purpose



Application Specific



CLAIM: THE MAJOR CHALLENGE TO NEUROMORPHIC COMPUTING TODAY IS THE ALGORITHM IMPACT

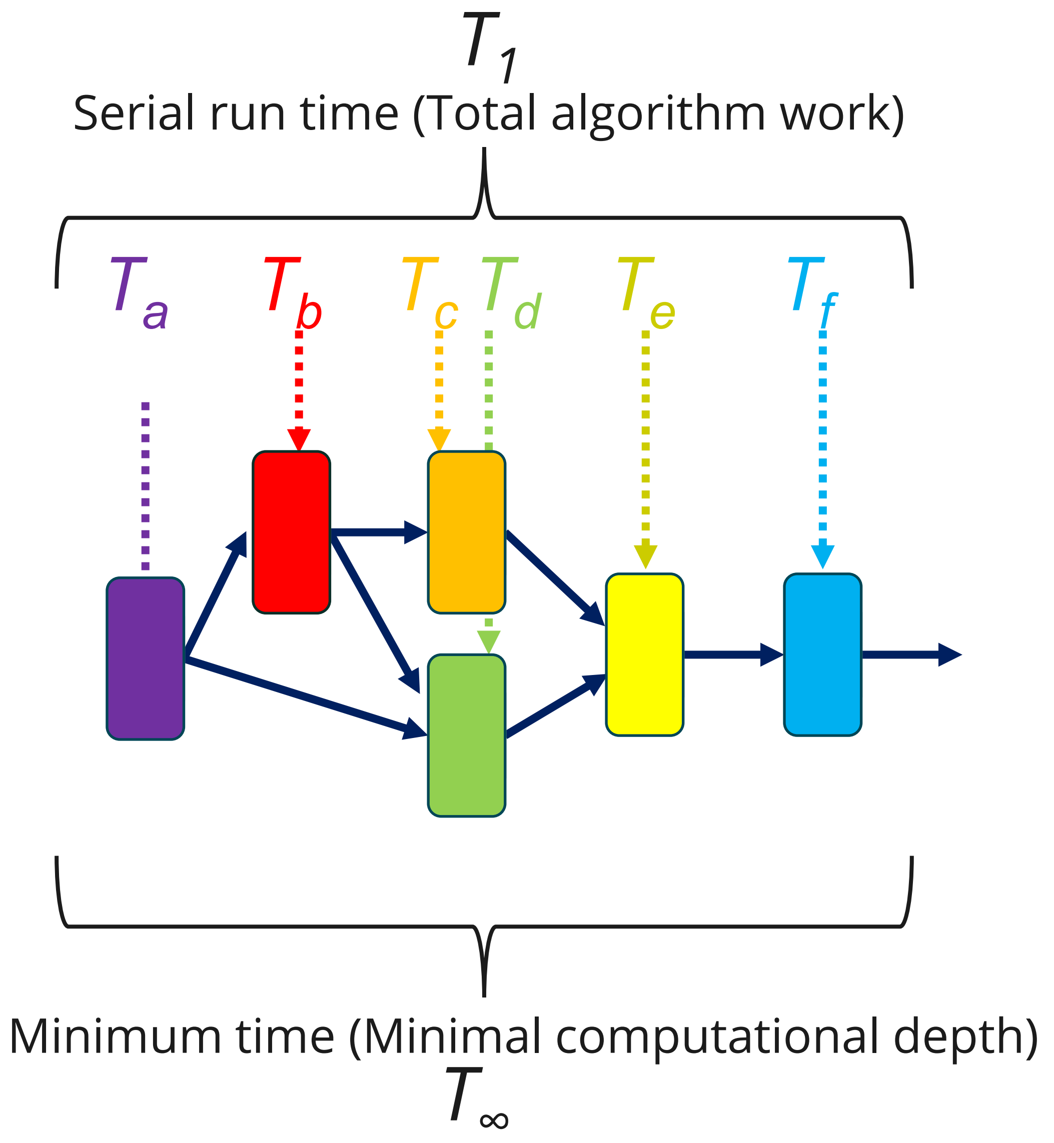


Identifying neuromorphic advantages today will

- ❑ Communicate the fundamental value proposition of neuromorphic hardware
- ❑ Determine how neuromorphic computing fits into the broader ecosystem (GPUs, accelerators, etc)
- ❑ Clarify what aspects of today's neuromorphic architectures need to be improved
- ❑ Justify cost of moving to new non-CMOS materials

Hypothesis: Identifying classes of computation that are preferentially accelerated by neuromorphic can perhaps do for those paradigms what GPUs did for AI

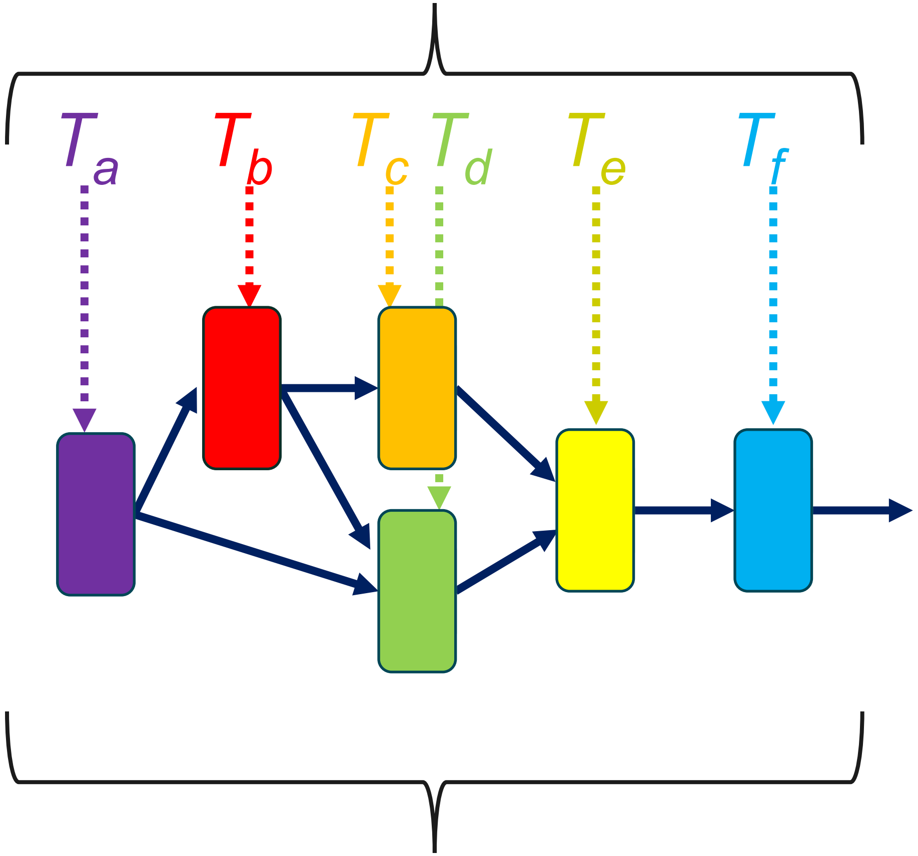
UNDERSTANDING NEUROMORPHIC COMPLEXITY



IN-MEMORY COMPUTING FUNDAMENTALLY CHANGES SCALING

T_1

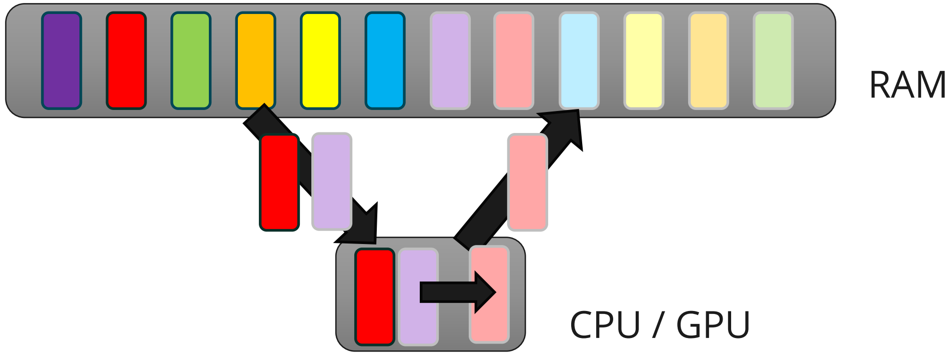
Serial run time (Total algorithm work)



Minimum time (Minimal computational depth)

T_∞

Conventional programs and data exist in memory

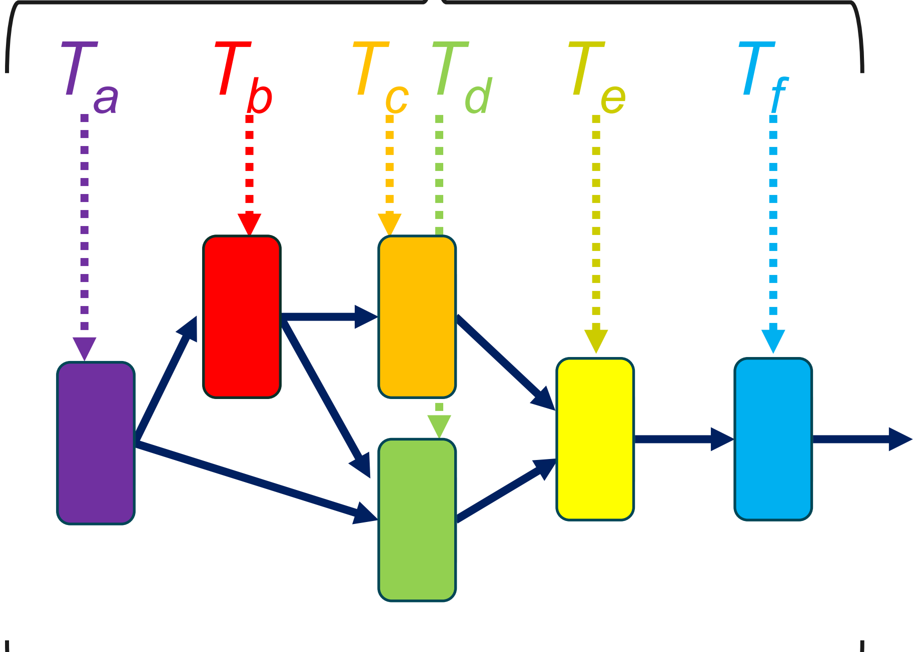


Neuromorphic programs must be fully realized in hardware

THE SIZE OF ALGORITHM DICTATES TIME AND SPACE OF NEUROMORPHIC IMPLEMENTATION

T_1

Serial run time (Total algorithm work)

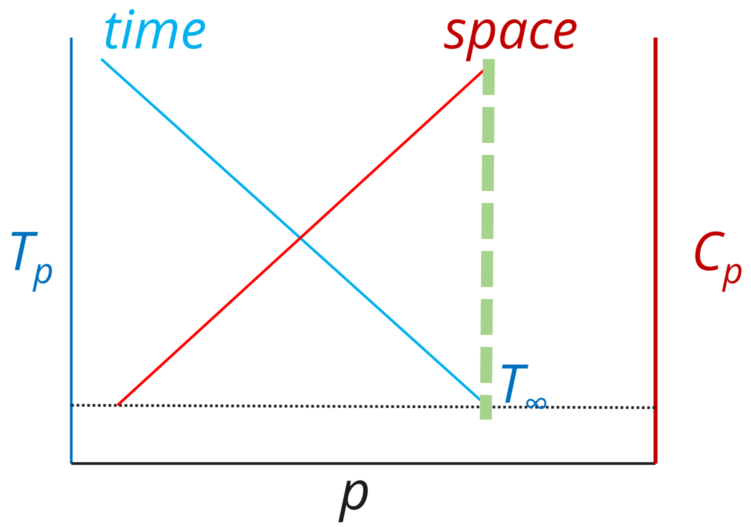


Minimum time (Minimal computational depth)

T_∞

Brent's Theorem: for p processors, run time T_p is bounded by graph depth T_∞

$$\max\left(T_\infty, \frac{T_1}{p}\right) \le T_p \le \frac{T_1}{p} + T_\infty$$



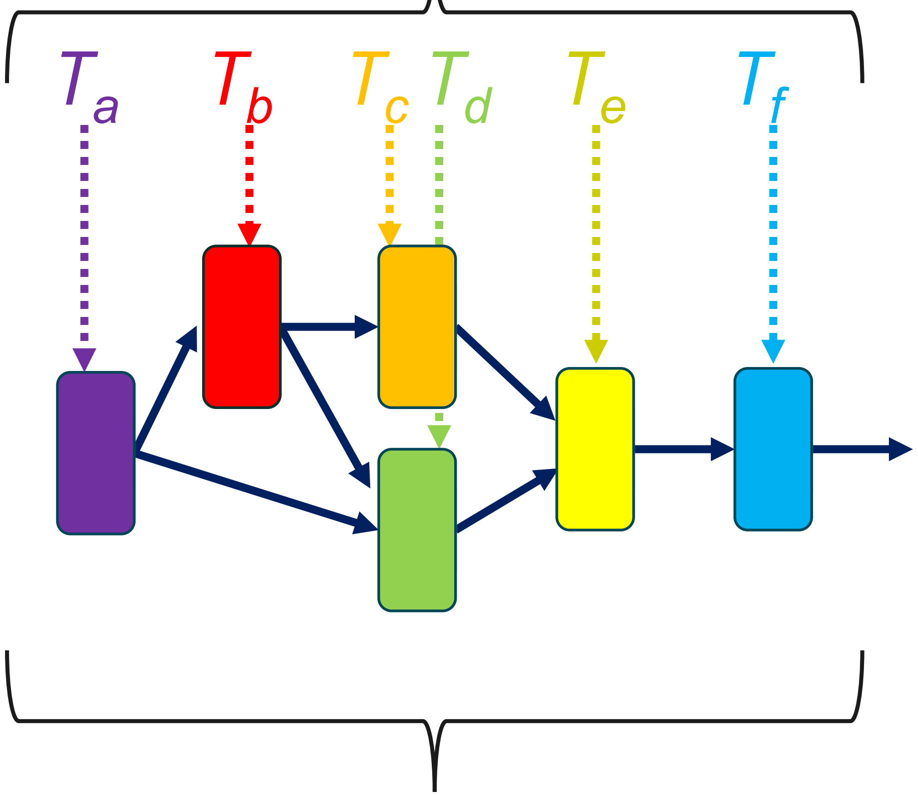
For conventional systems, time and space can be directly traded off

For NMC (and all in memory compute), *there is no tradeoff*

IMPLICATION: NEUROMORPHIC CAN BE FAST… BUT YOU NEED A LOT OF NEURONS

T_1

Serial run time (Total algorithm work)



Minimum time (Minimal computational depth)

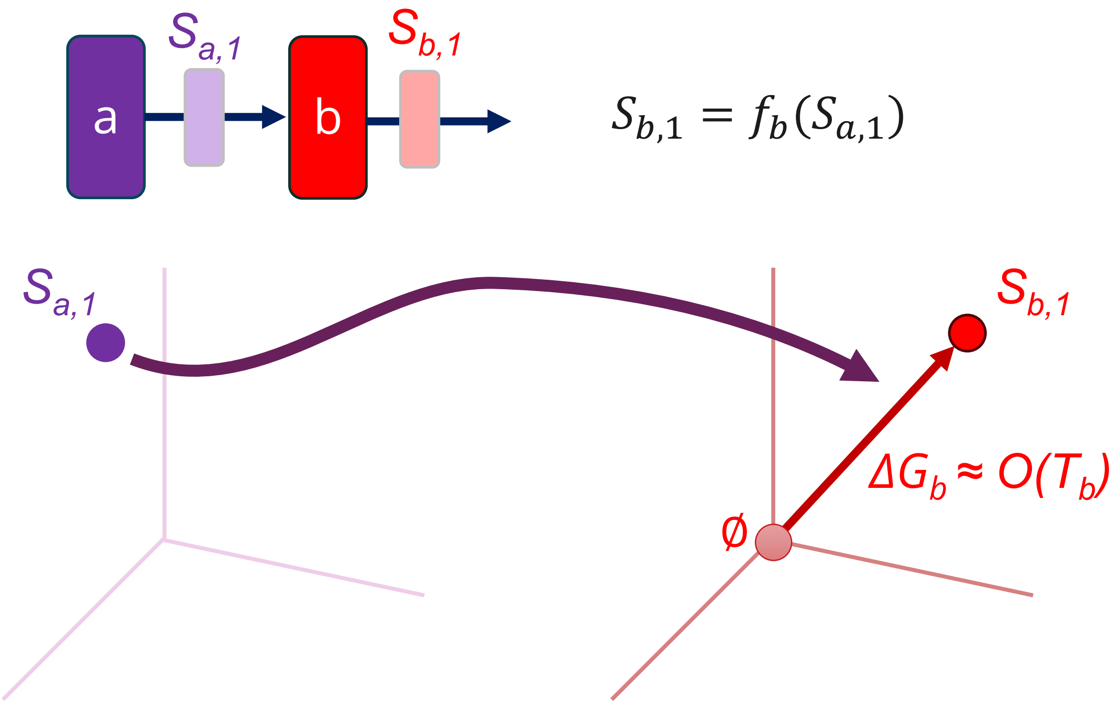
T_∞

TABLE II
TIME, SPACE, AND ENERGY SCALING OF NEUROMORPHIC AND CONVENTIONAL SYSTEMS

	Time (T)	Space (S)
Conventional		
Ideal	$O\left(\frac{T_1}{p}\right)$	$O(p)$
CPU (Realized)	$O(T_1)$	$O(1)$
GPU (Realized)	$O\left(\frac{T_1}{p \times p_{efficiency}}\right)$	$O(p)$
Neuromorphic		
Ideal	$O(T_{\text{inf}})$	$O(T_1)$
Realized	$O(N_{core} T_{\text{inf}})$	$O(T_1/N_{core})$

arXiv preprint arXiv:2507.17886

MEMORY ACCESS DOMINATES CONVENTIONAL ENERGY



$$S_{b,1} = f_b(S_{a,1})$$

Energy =
Energy(operations) + **Energy(communication)**

Conventional

- Energy costs (at least within processors) are largely communication to and from memory
- Because processors are shared, conventional algorithms must continually communicate algorithm states to and from memory.
- Memory updates are largely state-independent, so energy costs are relatively independent of initial and end states

Conventional energy scales with total work

NEUROMORPHIC ENERGY DOES NOT SCALE THE SAME WAY

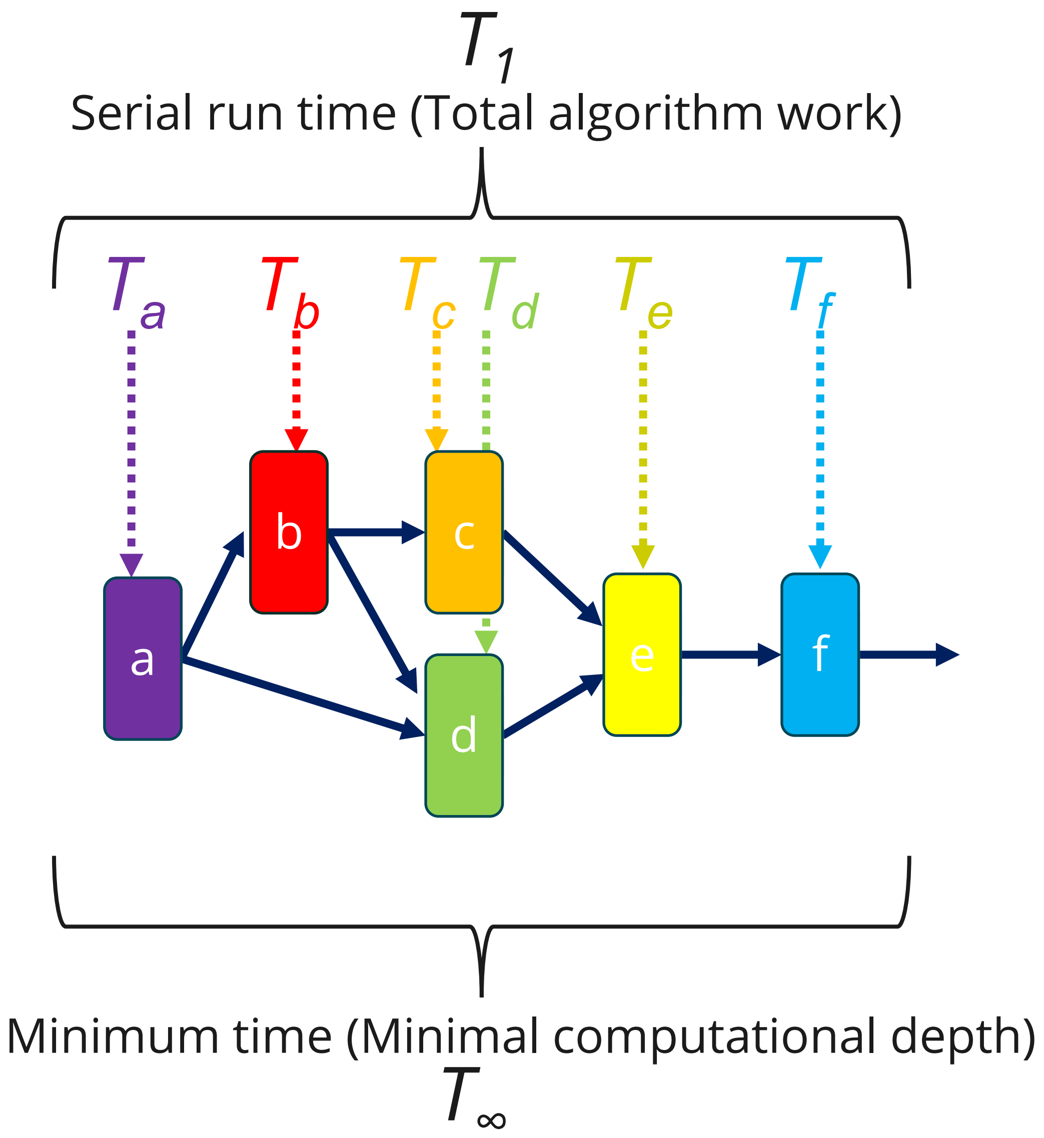


TABLE II
TIME, SPACE, AND ENERGY SCALING OF NEUROMORPHIC AND CONVENTIONAL SYSTEMS

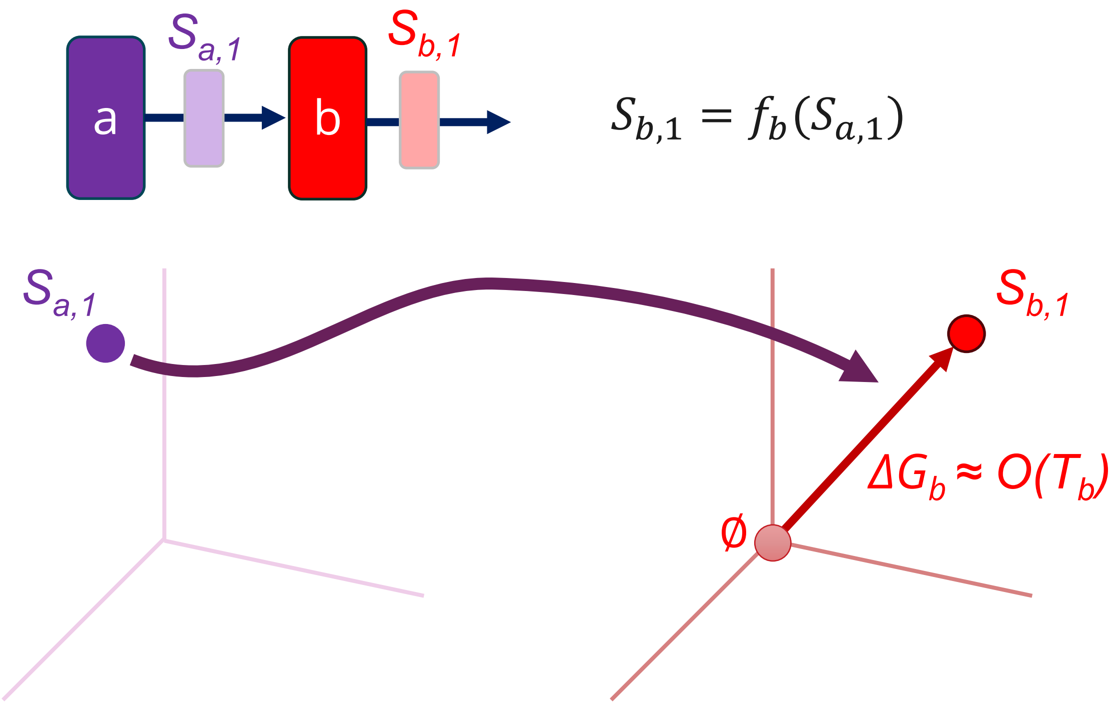
	Time (T)	Space (S)	Energy (E)
Conventional			
Ideal	$O\left(\frac{T_1}{p}\right)$	$O(p)$	$O(T_1)$
CPU (Realized)	$O(T_1)$	$O(1)$	$O(T_1)$
GPU (Realized)	$O\left(\frac{T_1}{p \times P_{efficiency}}\right)$	$O(p)$	$O(T_1)$
Neuromorphic			
Ideal	$O(T_{\inf})$	$O(T_1)$	
Realized	$O(N_{core}T_{\inf})$	$O(T_1/N_{core})$	

arXiv preprint arXiv:2507.17886

WHAT DOES EVENT-DRIVEN IMPLY?

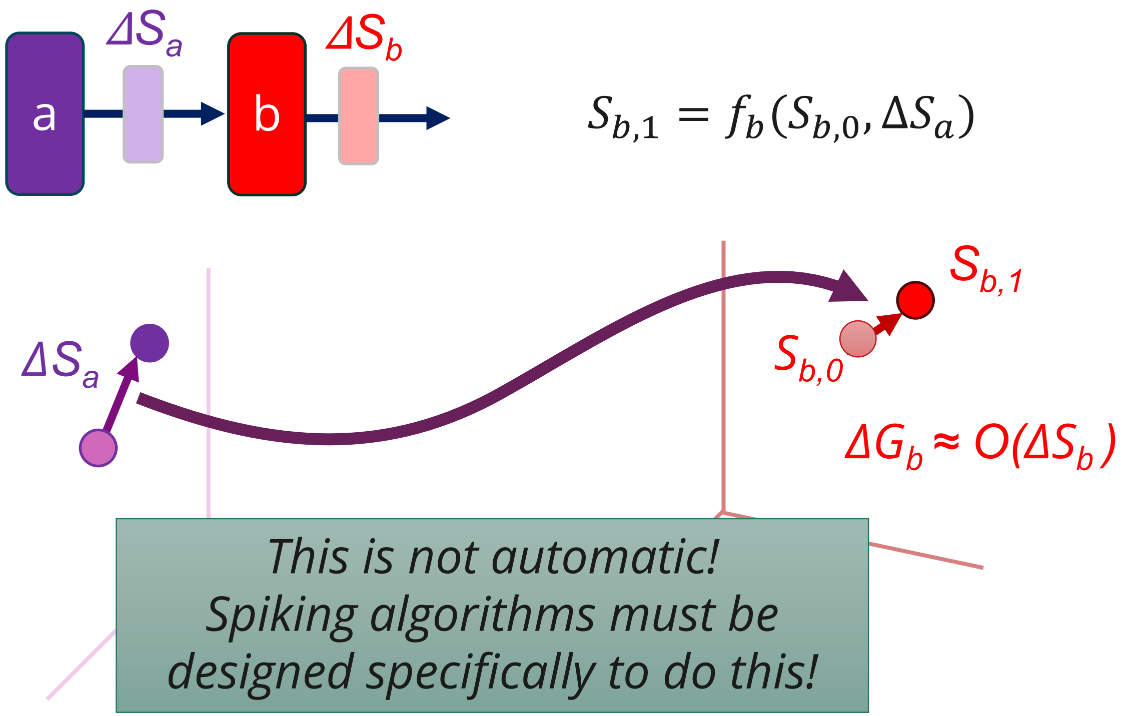
Because processors are shared, conventional algorithms must continually communicate algorithm states to and from memory.

This is why energy cost is relatively independent of what initial and end states are.



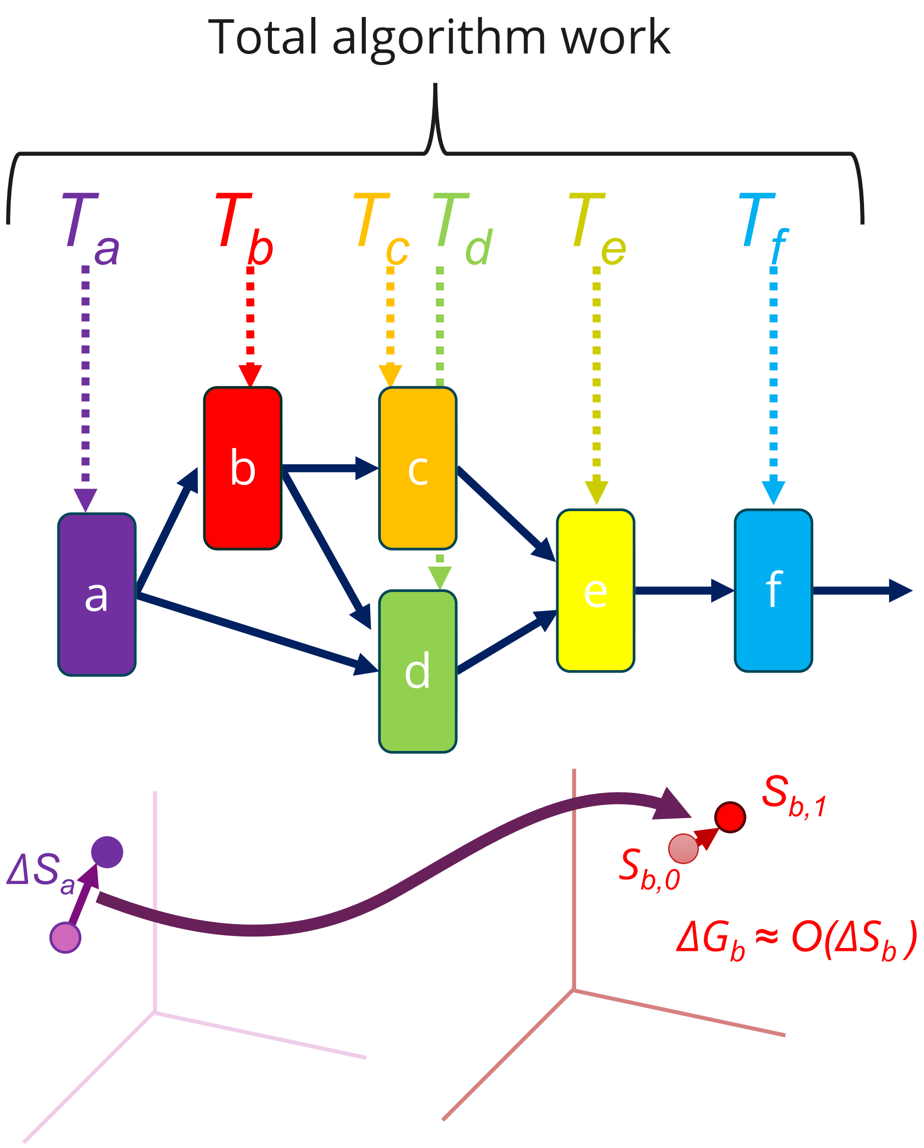
Neurons are event-driven, which means their state is only updated if necessary.

Neural algorithm communication should focus on how these states should be updated



This is not automatic!
Spiking algorithms must be designed specifically to do this!

ENERGY OF NEUROMORPHIC SCALES WITH *CHANGE OF STATE*



Energy =
Energy(operations) + **Energy(communication)**

Conventional

- Energy costs are largely communication to and from memory
- Memory updates are largely state-independent

Conventional energy scales with total work

Neuromorphic

- Energy costs are largely communication between compute elements
- Communication is event-driven. No energy expended if there is no change

Neuromorphic energy scales with the <u>change of state</u> across computational graph

- Upper bound is still total work
- Lower bound can be really low…

ENERGY OF NEUROMORPHIC SCALES WITH *CHANGE OF STATE*

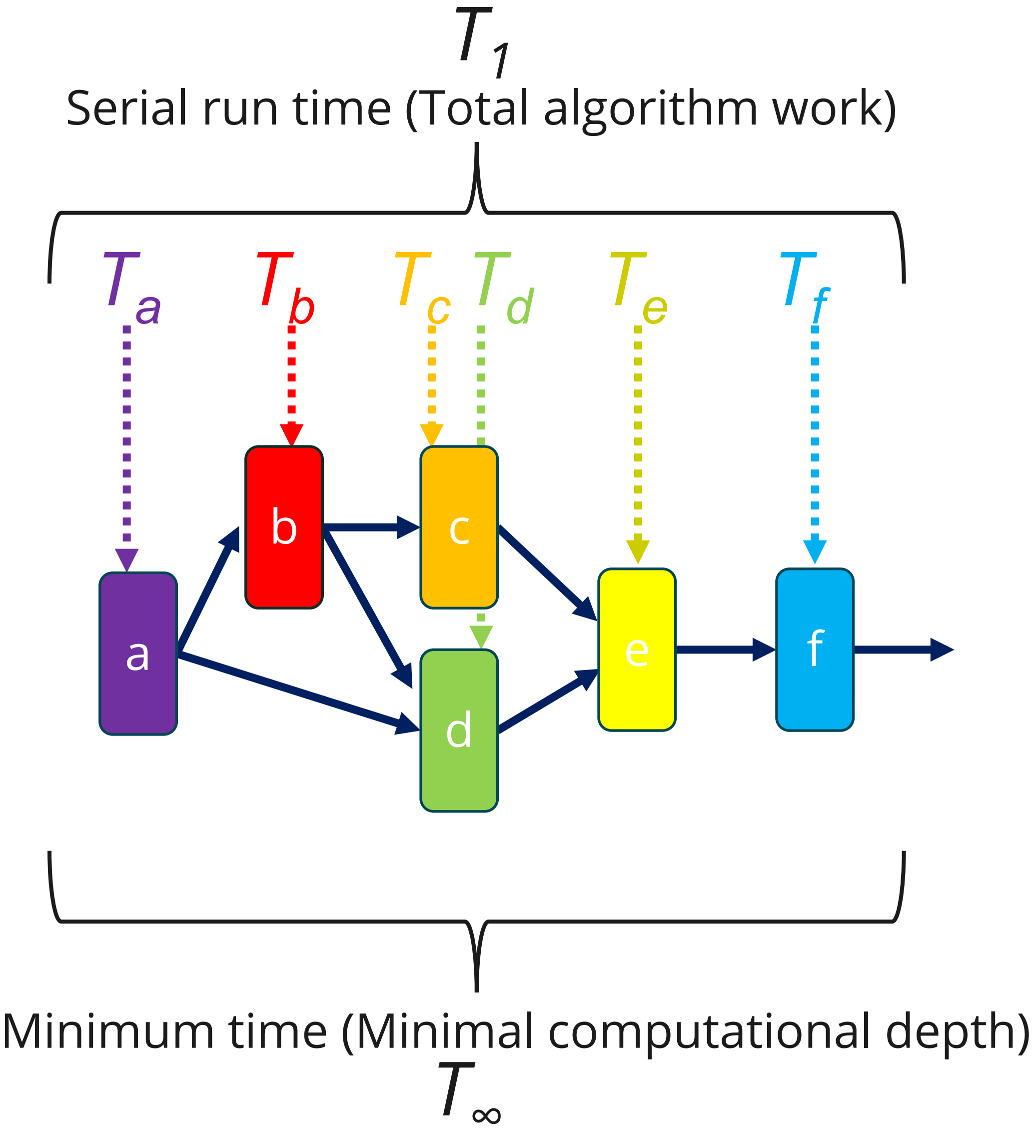
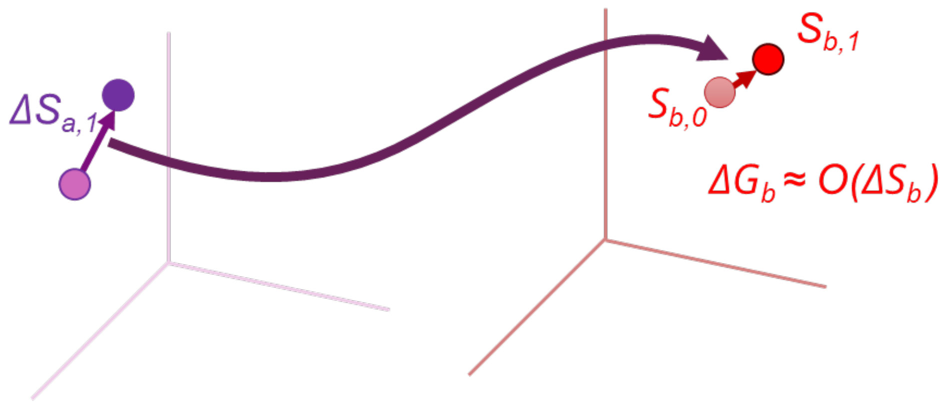


TABLE II
TIME, SPACE, AND ENERGY SCALING OF NEUROMORPHIC AND CONVENTIONAL SYSTEMS

	Time (T)	Space (S)	Energy (E)
Conventional			
Ideal	$O\left(\frac{T_1}{p}\right)$	$O(p)$	$O(T_1)$
CPU (Realized)	$O(T_1)$	$O(1)$	$O(T_1)$
GPU (Realized)	$O\left(\frac{T_1}{p \times p_{efficiency}}\right)$	$O(p)$	$O(T_1)$
Neuromorphic			
Ideal	$O(T_{\inf})$	$O(T_1)$	$O(\Delta G)$
Realized	$O(N_{core}T_{\inf})$	$O(T_1/N_{core})$	$O(\Delta G)$

arXiv preprint arXiv:2507.17886

WHAT MAKES FOR A LOW ΔG?



Low ΔG

- Algorithm state does not change much
- Path-dependent trajectories
- Does not change algorithmic work

Promising candidates

- Sparse computations
- **Monte Carlo algorithms**
- **Iterative and recurrent algorithms**
- Optimization
- Online algorithms

High ΔG ($\Delta G \approx T_1$)

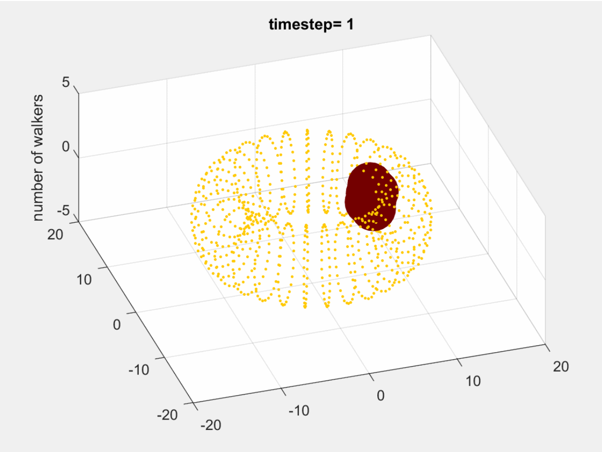
- Algorithm state changes are extensive
- Algorithms “touch” most/all memory
- Equal to algorithm work if everything changes with each operation

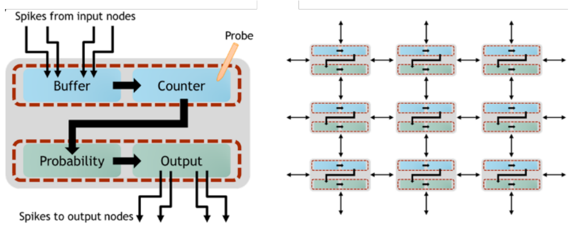
Example candidates

- Dense linear algebra
- Graphics rendering
- Modern AI algorithms (as formulated)
- Cryptography

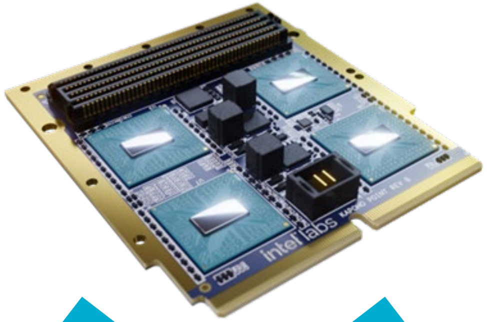
PART 3:

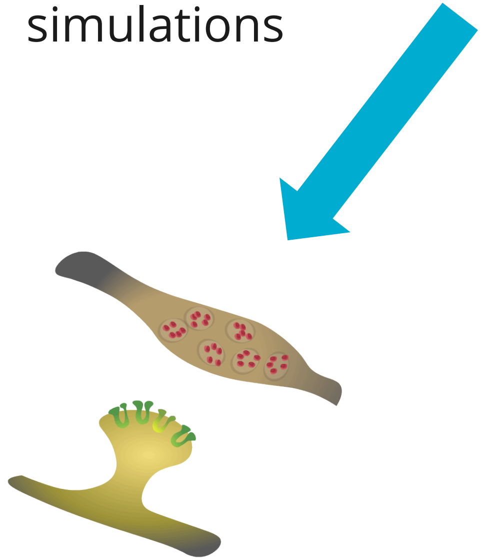
HOW DOES NEUROMORPHIC IMPACT REAL COMPUTING APPLICATIONS?



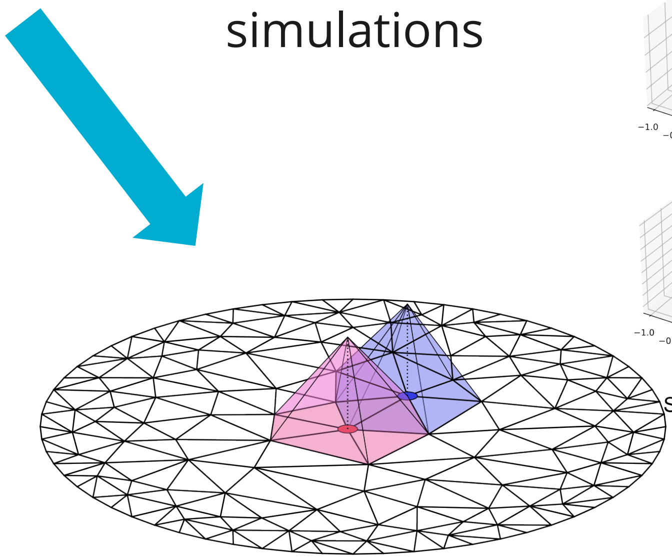


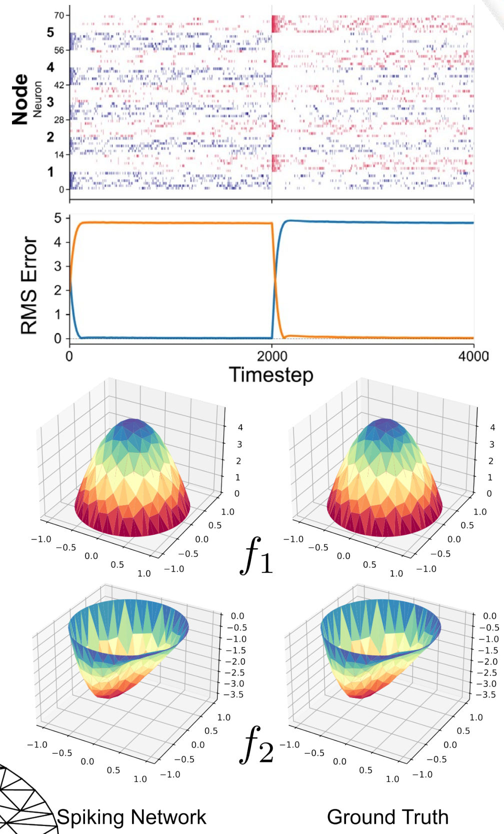
Neuromorphic hardware can *efficiently* solve stochastic Monte Carlo simulations





Neuromorphic hardware can *efficiently* solve finite element method simulations



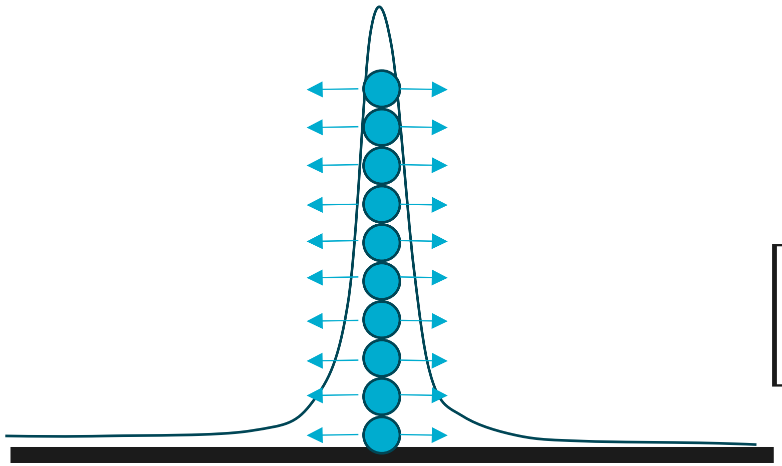


Theilman and Aimone, in press 2025

Smith et al., Nature Electronics, 2022

CAN WE REFORMULATE MONTE CARLO FOR NEUROMORPHIC?

Initial State

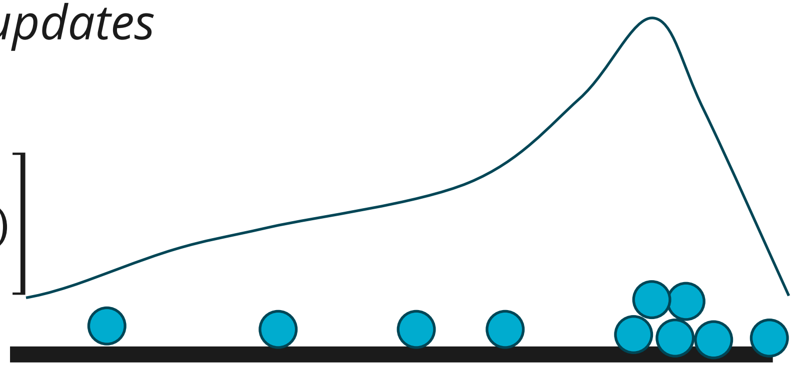


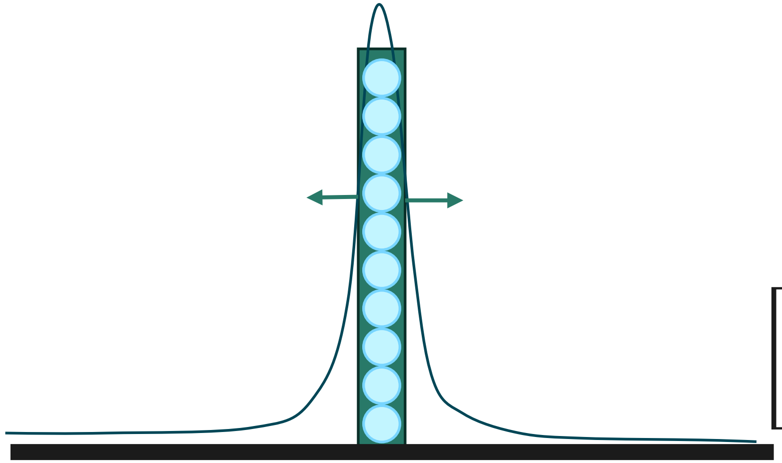
$$dx_i/dt = f(x_i, X, \cdots)$$

K particles   *K location-dependent updates*

$$\begin{bmatrix}\uparrow\\ x_i(t+1)\\ \downarrow\end{bmatrix} = f\left(\begin{bmatrix}\uparrow\\ x_i(t)\\ \downarrow\end{bmatrix}, X, \cdots\right) + \begin{bmatrix}\uparrow\\ x_i(t)\\ \downarrow\end{bmatrix}$$

Final State

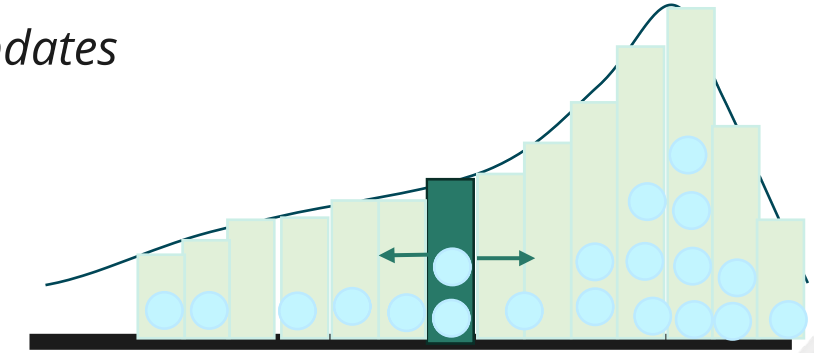




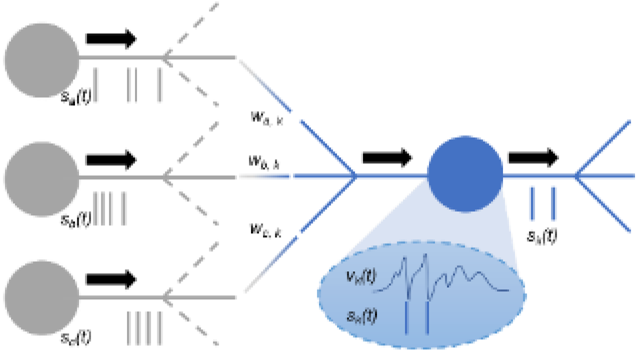
$$dm_i/dt = g_i(X, \cdots)$$

M locations   *M location-specific updates*

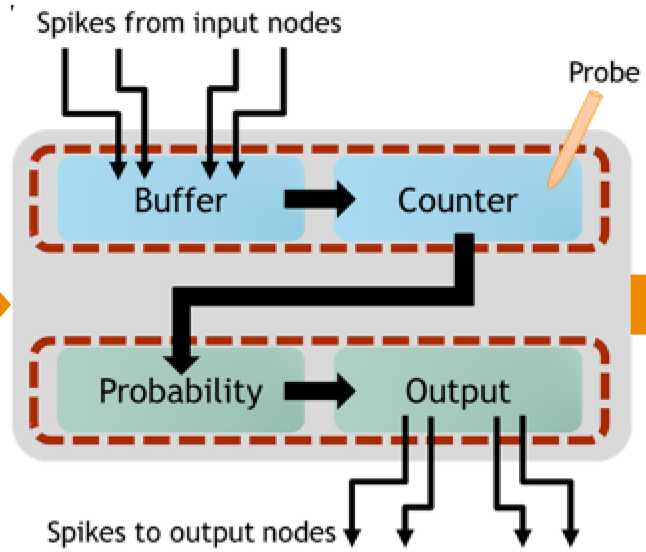
$$\begin{bmatrix}\uparrow\\ m_i(t+1)\\ \downarrow\end{bmatrix} = g_i(X, \cdots) + \begin{bmatrix}\uparrow\\ m_i(t)\\ \downarrow\end{bmatrix}$$

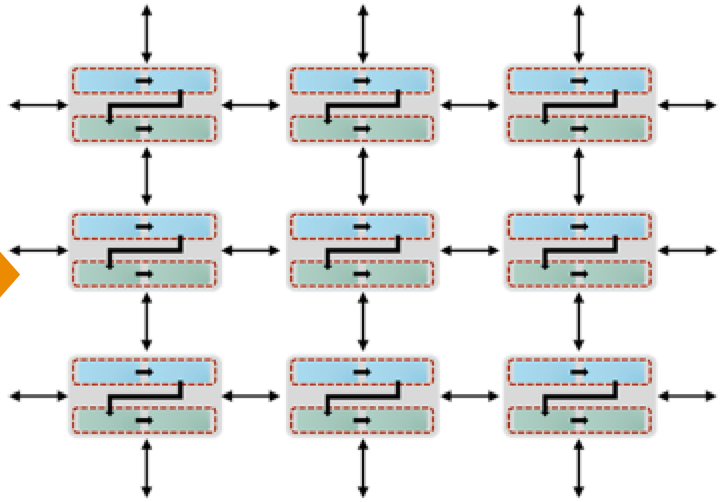


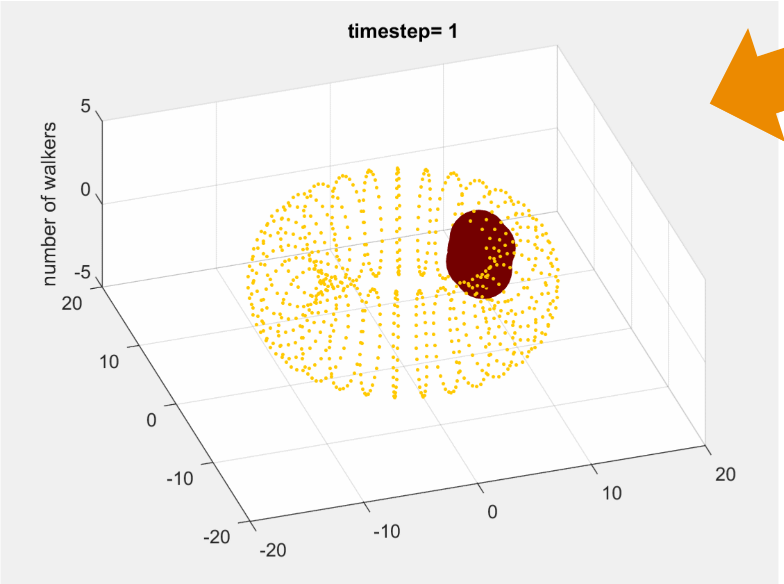
USE NEURONS TO REPRESENT *STATE SPACE* OF MONTE CARLO AND USE SPIKES TO REPRESENT PARTICLES

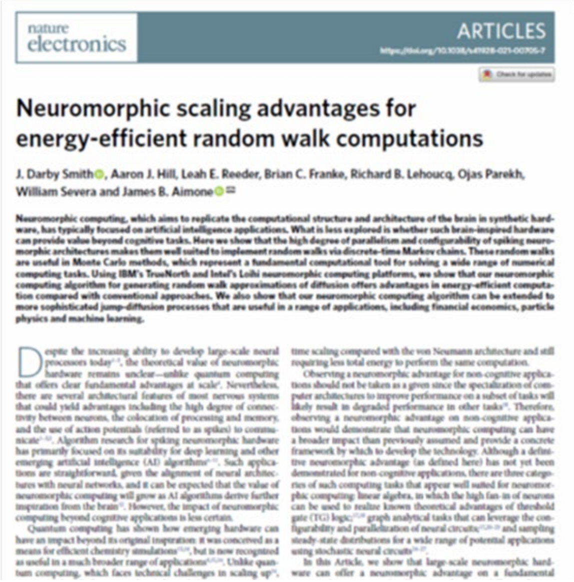


Leaky integrate-and-fire neuron









nature electronics

ARTICLES

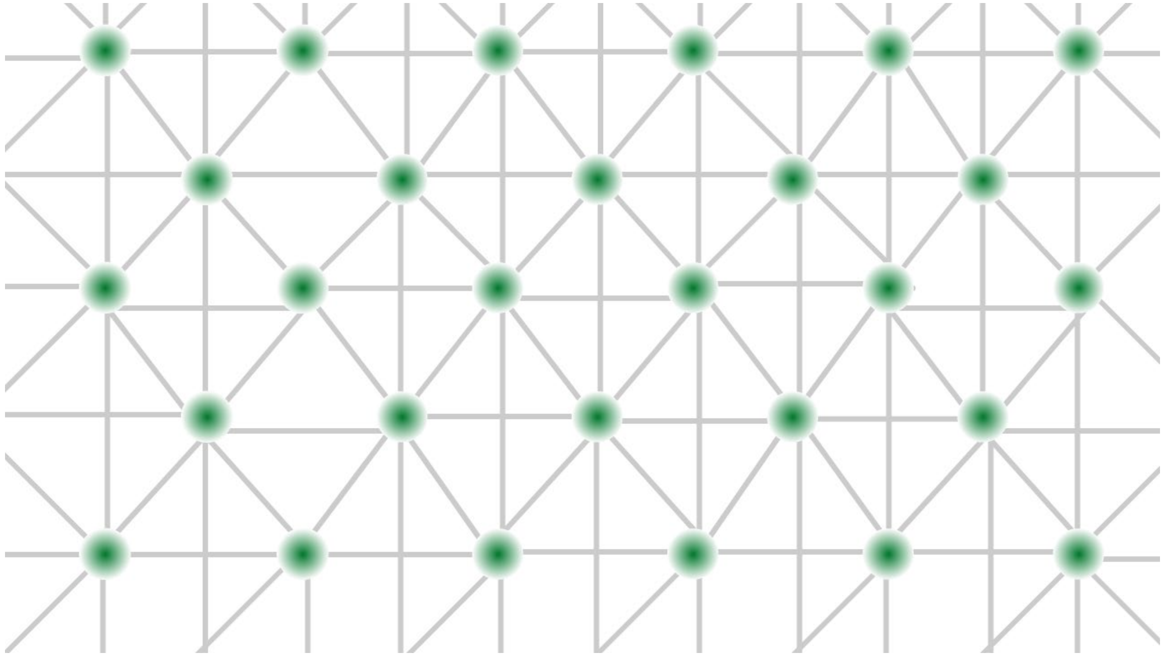
Neuromorphic scaling advantages for energy-efficient random walk computations

J. Darby Smith, Aaron J. Hill, Leah E. Reeder, Brian C. Franke, Richard B. Lehoucq, Ojas Parekh, William Severa and James B. Aimone

Neuromorphic computing, which aims to replicate the computational structure and architecture of the brain in synthetic hardware, has typically focused on artificial intelligence applications. What is less explored is whether such brain-inspired hardware can provide value beyond cognitive tasks. Here we show that the high degree of parallelism and configurability of spiking neuromorphic architectures makes them well suited to implement random walks via discrete-time Markov chains. These random walks are useful in Monte Carlo methods, which represent a fundamental computational tool for solving a wide range of numerical computing tasks. Using IBM's TrueNorth and Intel's Loihi neuromorphic computing platforms, we show that our neuromorphic computing algorithm for generating random walk approximations of diffusion offers advantages in energy-efficient computation compared with conventional approaches. We also show that our neuromorphic computing algorithm can be extended to more sophisticated jump-diffusion processes that are useful in a range of applications, including financial economics, particle physics and machine learning.

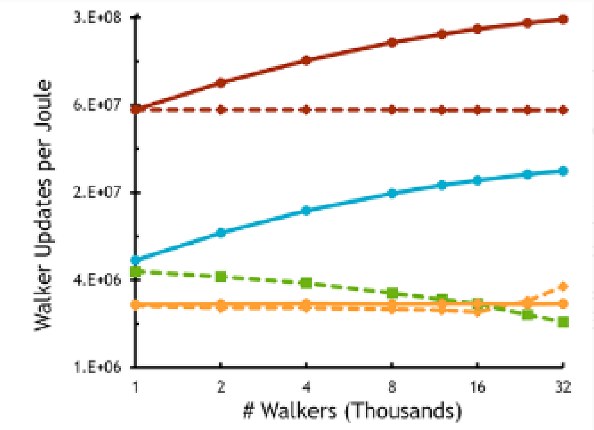
Smith et al., Nature Electronics 2022

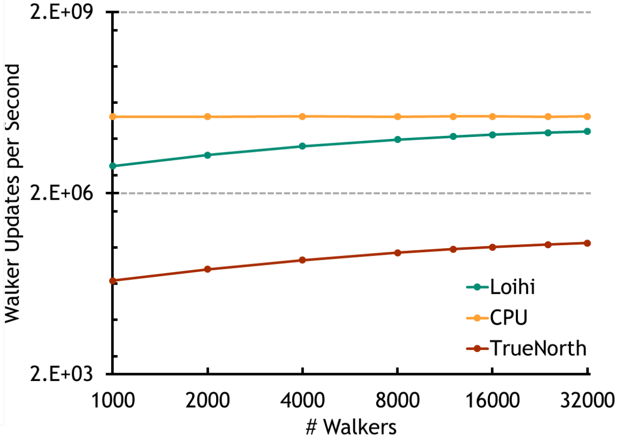
NEUROMORPHIC COMPUTING ADVANTAGE APPEARS TO BE WHEN AN ALGORITHM CAN SPLIT THE TASK ACROSS COMPUTATIONAL GRAPH WITH SPARSE COMMUNICATION

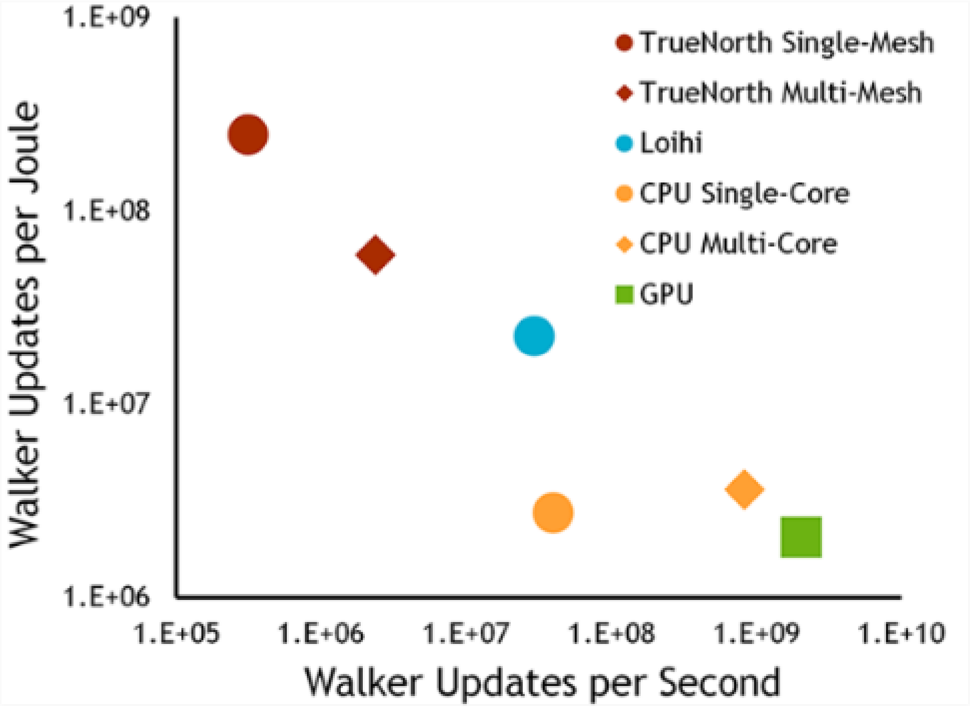


WE CAN IDENTIFY A NEUROMORPHIC ADVANTAGE FOR SIMULATING RANDOM WALKS

We define a *neuromorphic advantage* as an algorithm that shows a demonstrable **advantage** in terms of one resource (e.g., energy) while exhibiting comparable **scaling** in other resources (e.g., time).







Smith et al., Nature Electronics 2022

WHAT PDES CAN NEURAL RANDOM WALKS ADDRESS?

Class of Partial Integro-Differential Equations:

$$\frac{\partial}{\partial t}u(t,\boldsymbol{x}) = \frac{1}{2}\sum_{i,j}(\boldsymbol{a}\boldsymbol{a}^\top)_{i,j}(t,\boldsymbol{x})\frac{\partial^2}{\partial x_i \partial x_j}u(t,\boldsymbol{x}) + \sum_i b_i(t,\boldsymbol{x})\frac{\partial}{\partial x_i}u(t,\boldsymbol{x})$$
$$+\lambda(t,\boldsymbol{x})\int\Big(u\big(t,\boldsymbol{x}+\boldsymbol{h}(t,\boldsymbol{x},q)\big) - u(t,\boldsymbol{x})\Big)\phi_Q(q;t,\boldsymbol{x})\mathrm{d}q$$
$$+c(t,\boldsymbol{x})u(t,\boldsymbol{x}) + f(t,\boldsymbol{x}), \qquad x \in \mathbb{R}^d, t \in [0,\infty).$$

Stochastic Process:

NMC Hardware Simulates This Stochastic Process

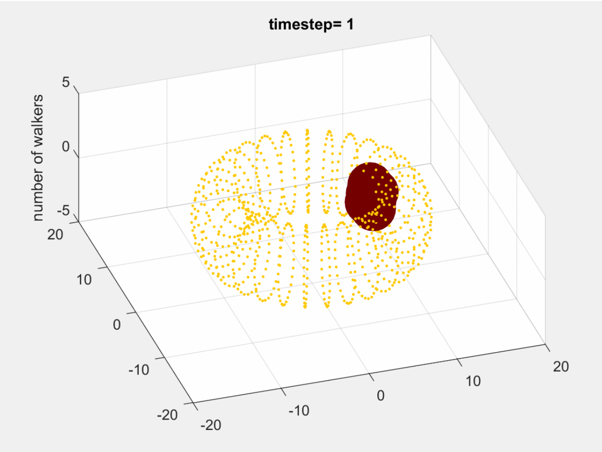
$$\mathrm{d}\boldsymbol{X}(t) = \boldsymbol{b}\big(t,\boldsymbol{X}(t)\big)\mathrm{d}t + \boldsymbol{a}\big(t,\boldsymbol{X}(t)\big)\mathrm{d}\boldsymbol{W}(t) + \boldsymbol{h}(t,\boldsymbol{X}(t),q)\mathrm{d}P\big(t;Q,\boldsymbol{X}(t)\big).$$

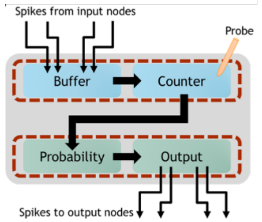
Monte Carlo Approximates This Expectation

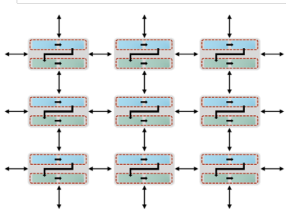
Solution to initial value problem (u(0,x)=g(x)):

$$u(t,\boldsymbol{x}) = \mathbb{E}\left[g\big(\boldsymbol{X}(t)\big)\exp\left(\int_0^t c\big(s,\boldsymbol{X}(s)\big)\mathrm{d}s\right) + \int_0^t f\big(s,\boldsymbol{X}(s)\big)\exp\left(\int_0^s c\big(\ell,\boldsymbol{X}(\ell)\big)\mathrm{d}\ell\right)\mathrm{d}s \,\middle|\, \boldsymbol{X}(0)=\boldsymbol{x}\right].$$

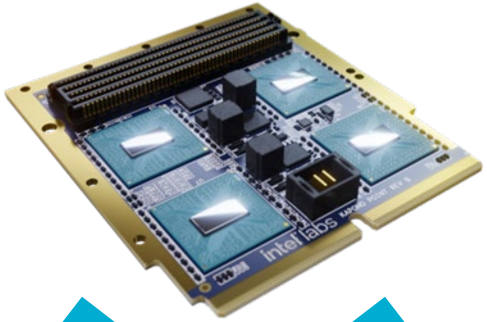
Smith et al., Nature Electronics 2022



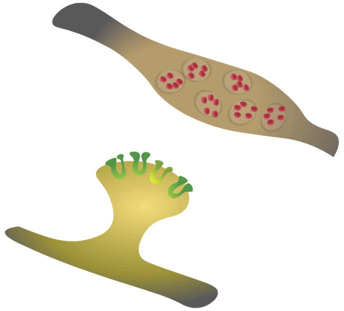


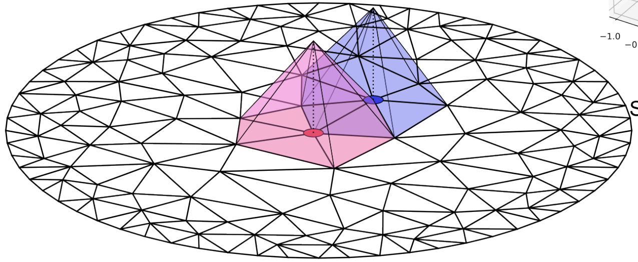


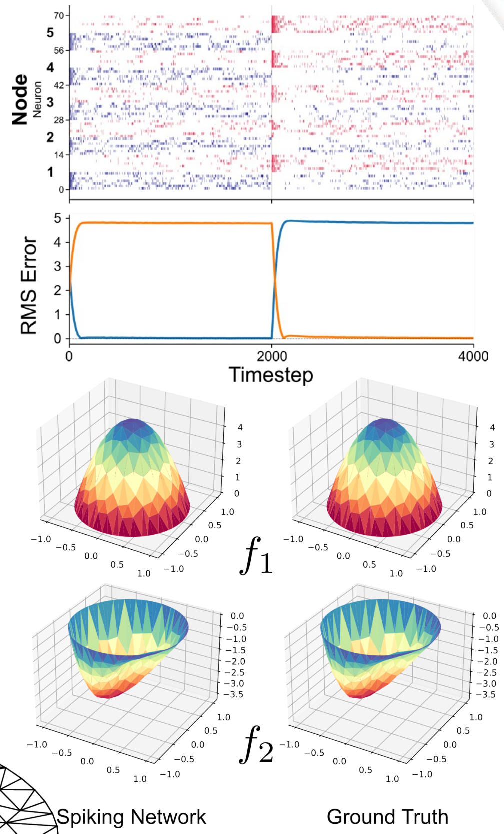
Neuromorphic hardware can *efficiently* solve stochastic Monte Carlo simulations



Neuromorphic hardware can *efficiently* solve finite element method simulations







Theilman and Aimone, in press 2025

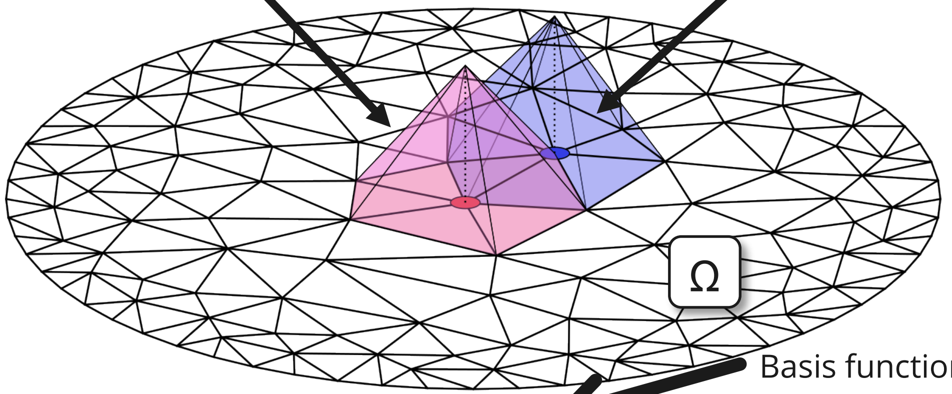
Smith et al., Nature Electronics, 2022

CAN WE TACKLE FEM WITH PROBABILISTIC NEURAL HARDWARE?

$$\boxed{\nabla^2 u = f}$$

Finite Element Methods – Gold Standard but expensive

$\varphi_i(x,y)$ $\varphi_j(x,y)$



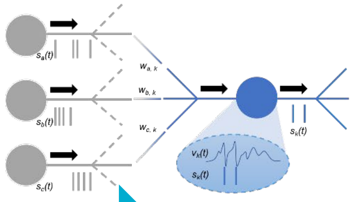
Basis functions

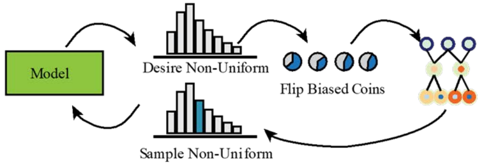
$$u(x,y) \approx \sum_i u_i \varphi_i(x,y)$$

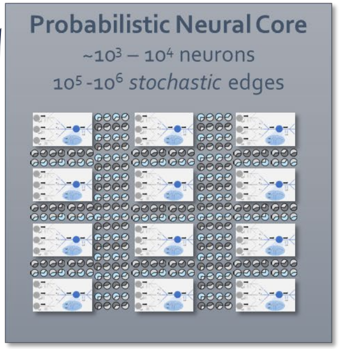
Coefficients

Sparse, linear system $\boxed{A\vec{u} = \vec{b}}$

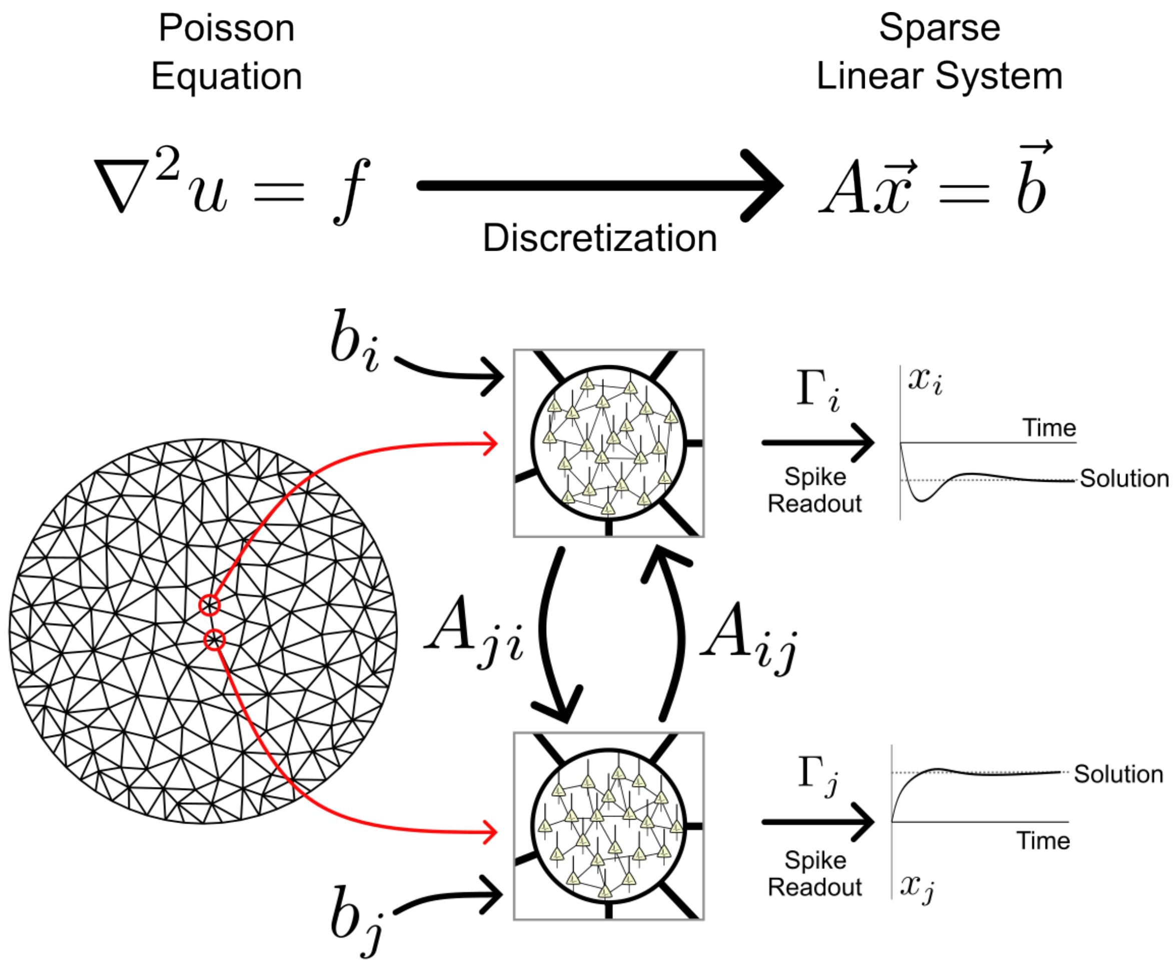
?????







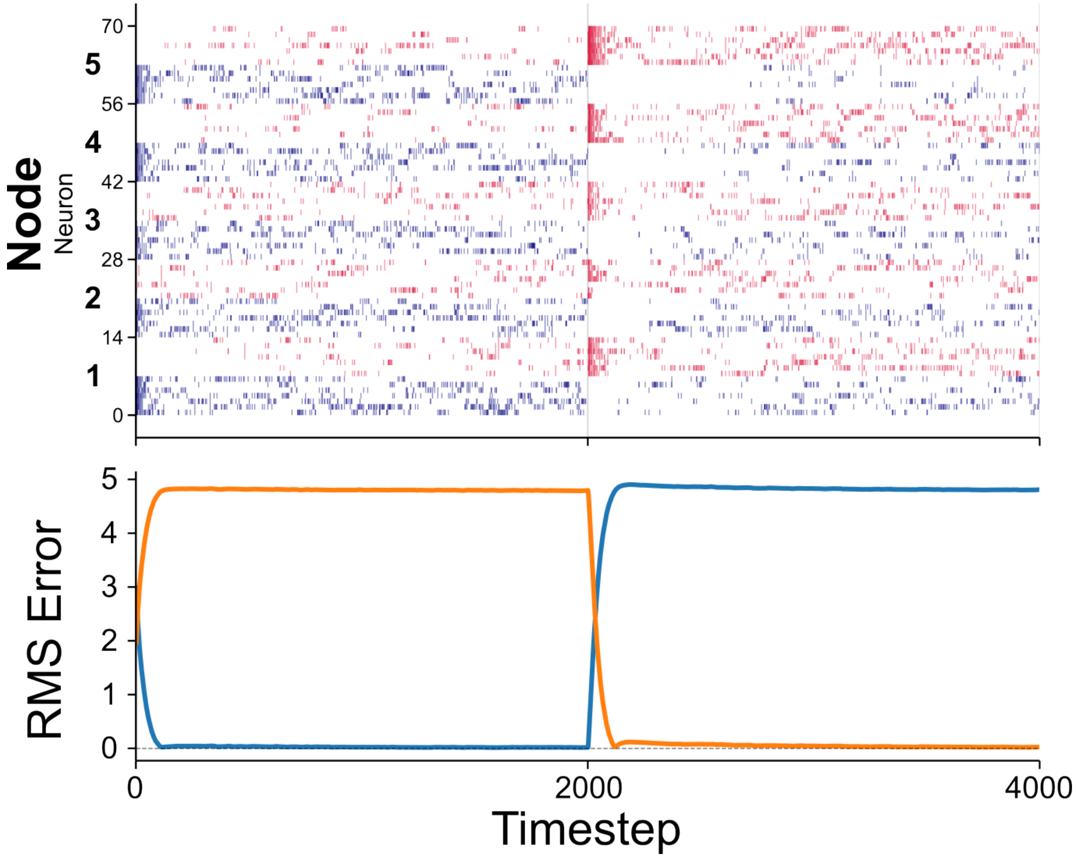
NEUROMORPHIC **FINITE ELEMENT METHODS?**

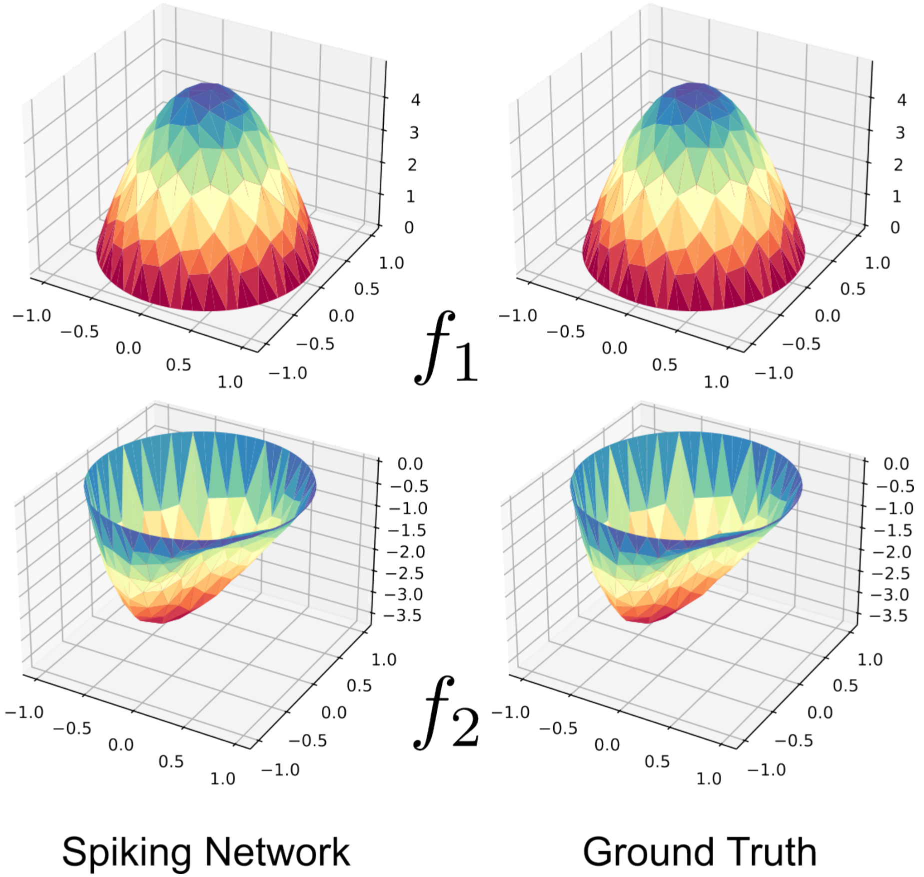


Neuromorphic virtues:

- Locally dense, globally sparse connectivity
- Scalable: neighbors ~ O(1)
- Sparse spiking activity

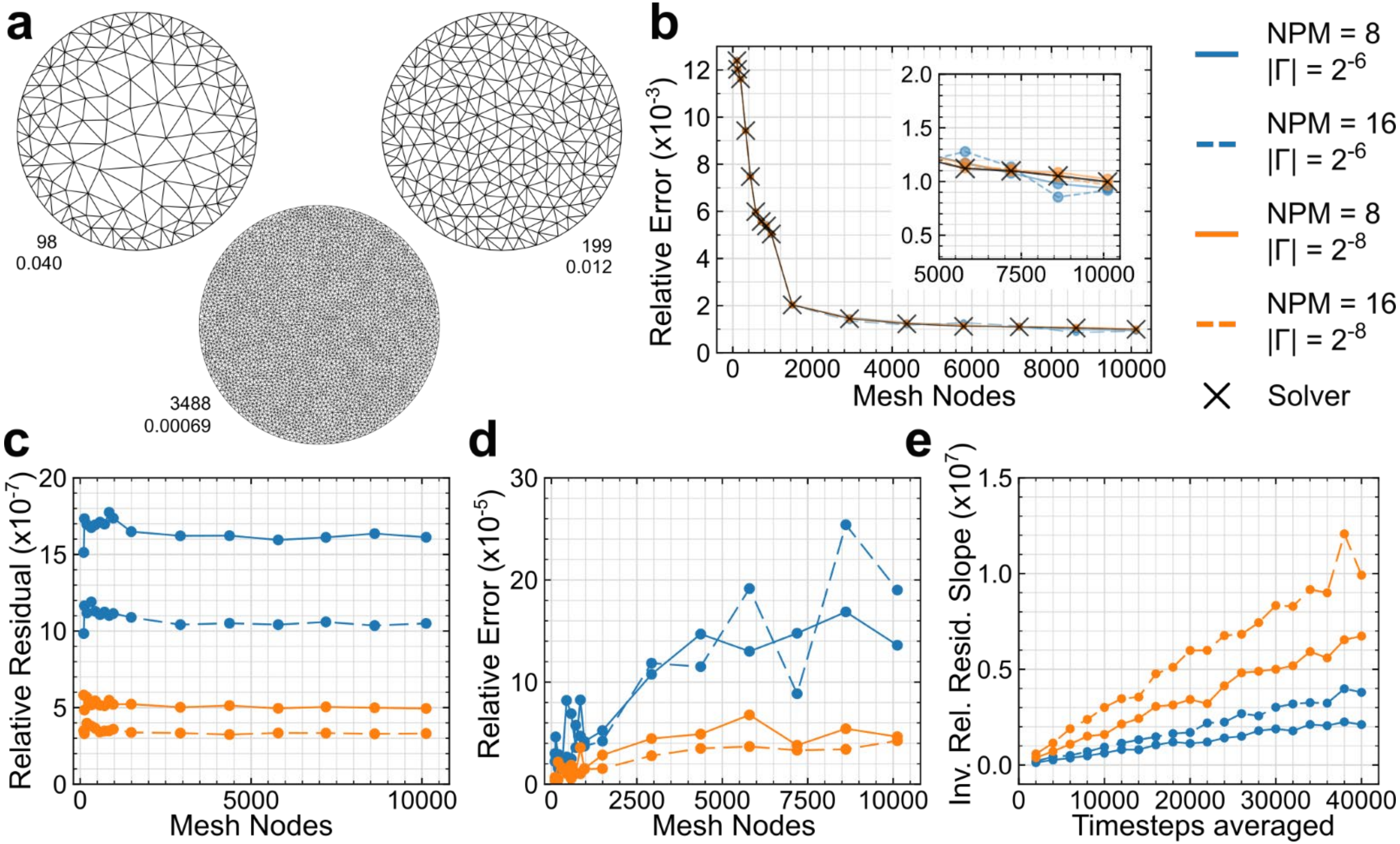
Theilman and Aimone, in press 2025





Theilman and Aimone, in press 2025

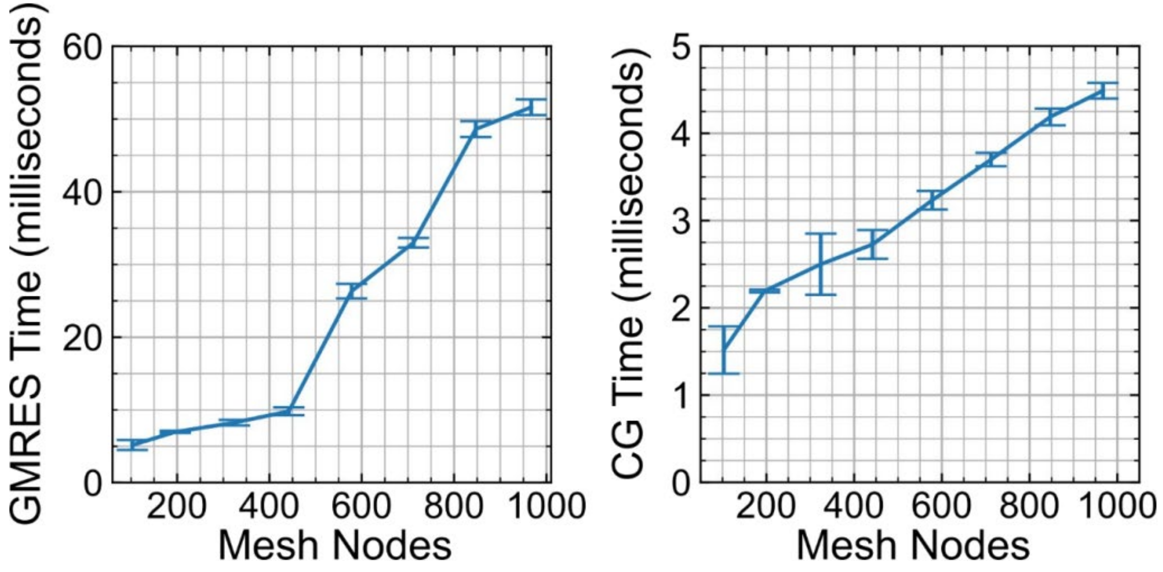
NEUROFEM PROVIDES SOLVER-QUALITY SOLUTIONS

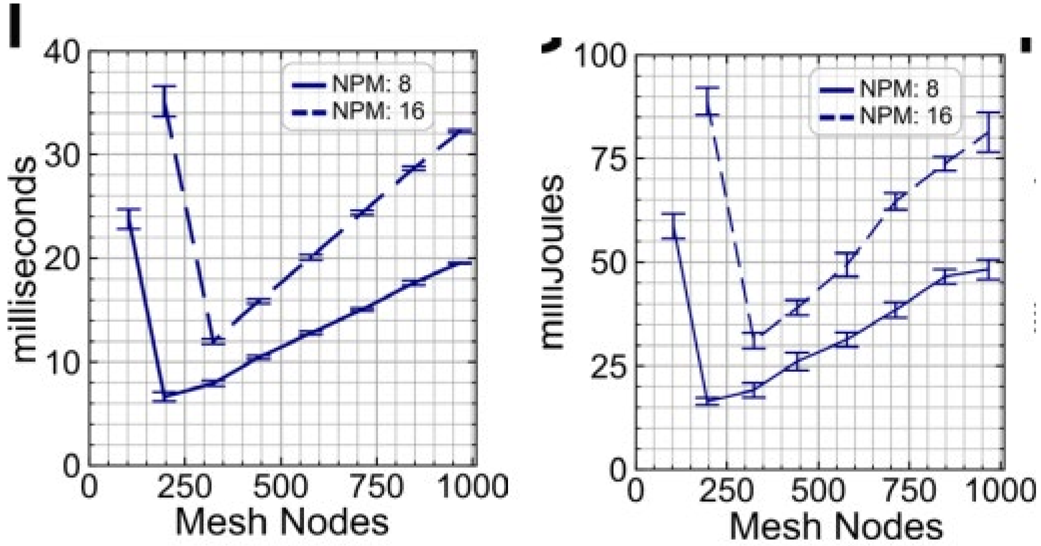


Theilman and Aimone, in press 2025

NEUROFEM IS SIMILAR IN SPEED TO GMRES AND CONJUGATE GRADIENT

Neuromorphic solution is general



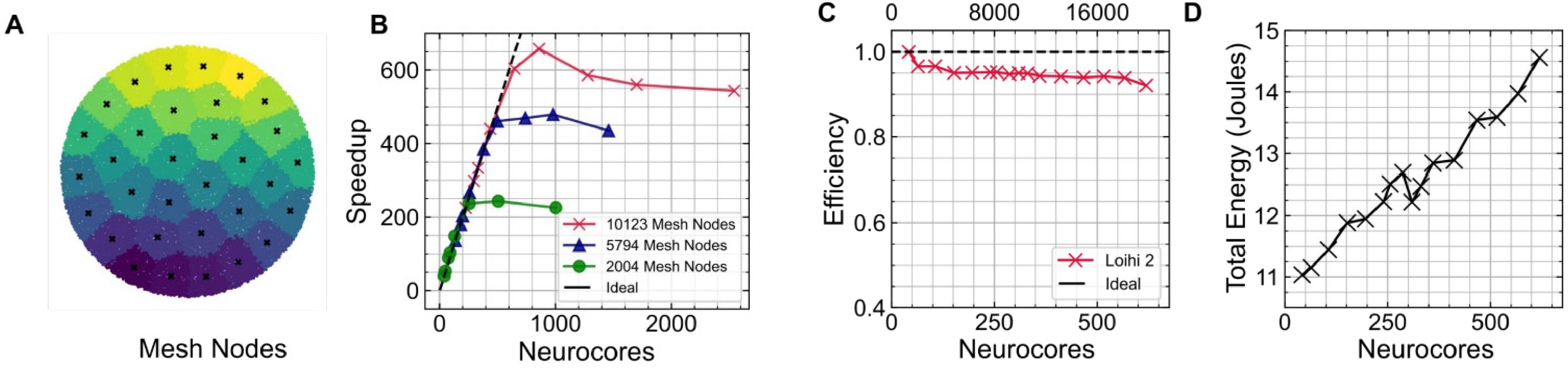


Conjugate gradient is faster at small scales, but can only be used for symmetric matrices

Theilman and Aimone, in press 2025

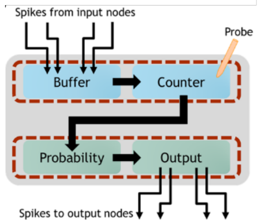
NEUROFEM SHOWS COMPELLING STRONG AND WEAK SCALING

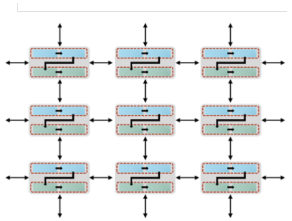
Strong Scaling **Weak Scaling**

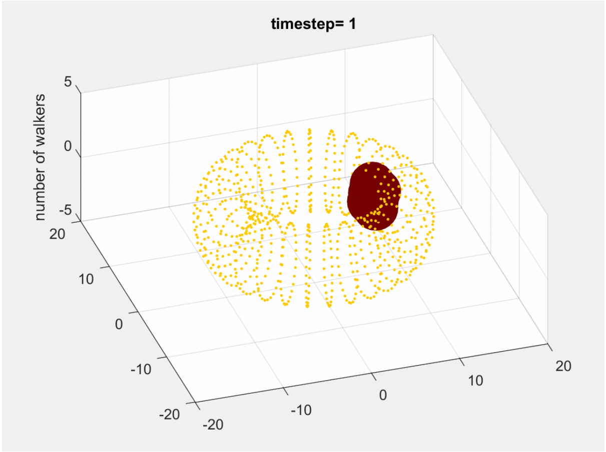


Theilman and Aimone, in press 2025

LOW-POWER SCIENTIFIC COMPUTING ON NEUROMORPHIC

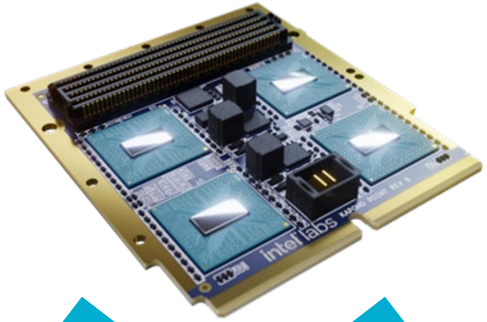




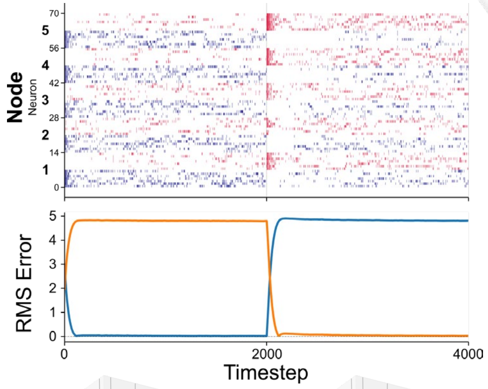


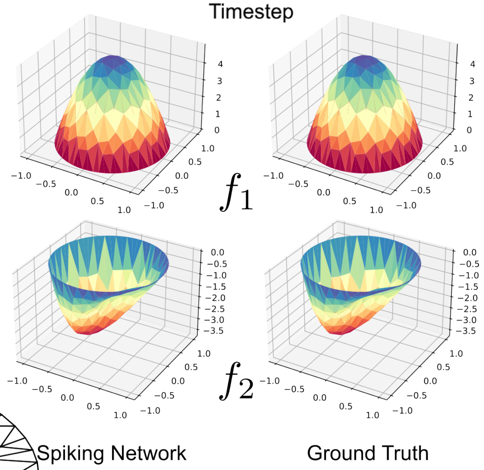
Severa et al., IJCNN 2018
Smith et al., ICONS 2020
Aimone et al., ICRC 2021
Smith et al., Nature Electronics, 20

Neuromorphic hardware can *efficiently* solve stochastic Monte Carlo simulations

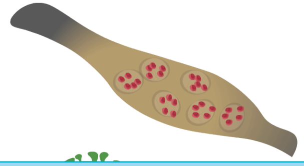


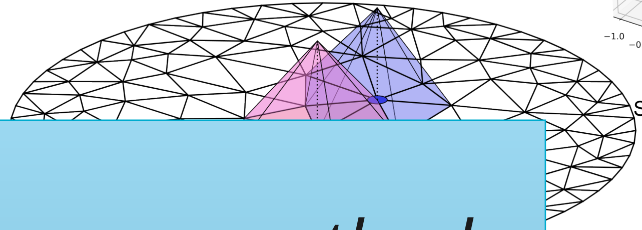
Neuromorphic hardware can *efficiently* solve finite element method simulations





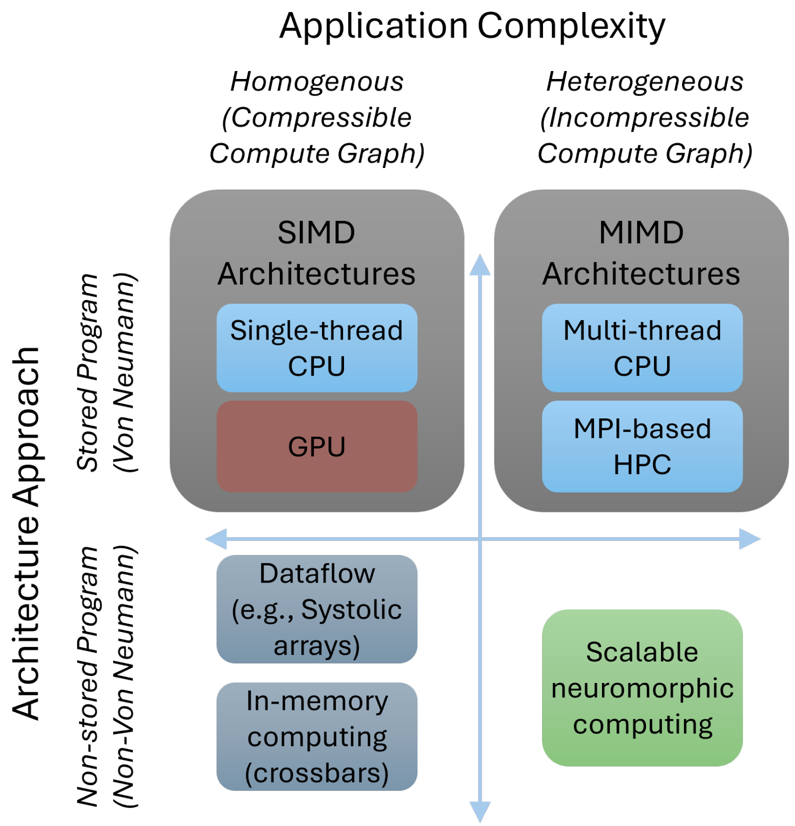
Theilman and Aimone, in press 2025





Both probabilistic and sparse methods are emerging frontiers in AI algorithms!

WE NEED MOD SIM TO DETERMINE HOW NEUROMORPHIC FITS INTO FUTURE HPC SYSTEMS



- Neuromorphic Computing offers distinct advantages *complementary* to existing accelerators and processors
- Effectively free from an energy and power perspective when suitably used
- *Back to basics* – this only makes sense if algorithms are co-designed to leverage neuromorphic advantages
- A path to energy-efficient AI?

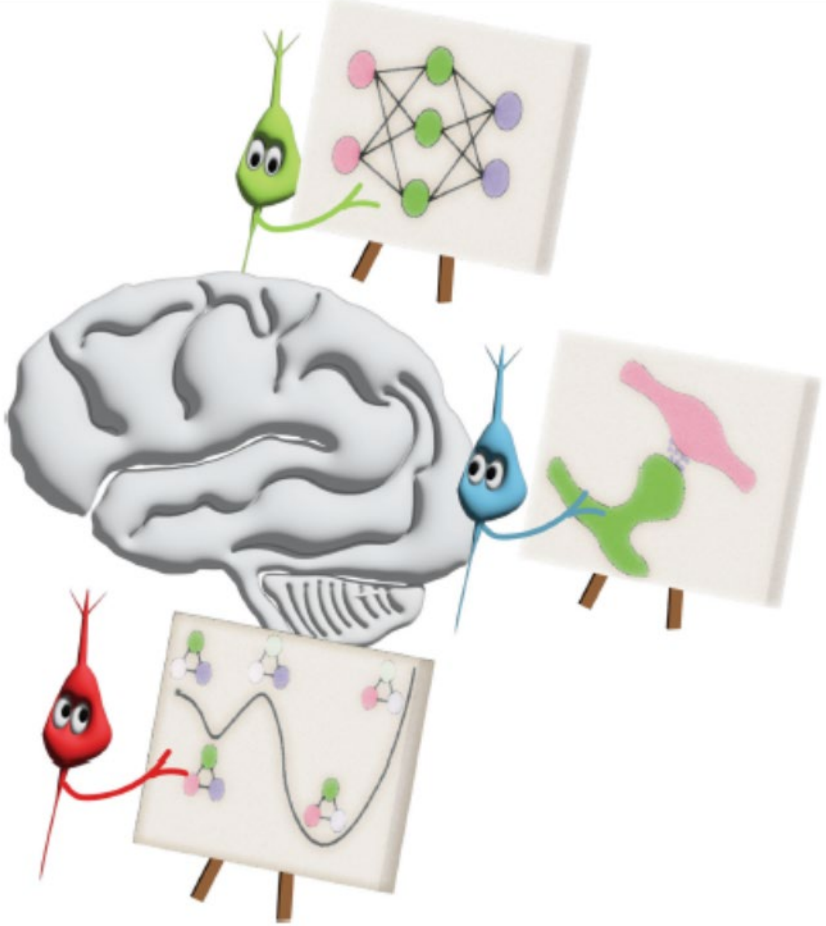
THANK YOU!

Any questions to jbaimon@sandia.gov

Neural Exploration & Research Lab
COGNITIVE & EMERGING COMPUTING

Craig M Vineyard, Suma G Cardwell, Corinne M Teeter, William Severa, **J Darby Smith**, Felix Wang, Fred Rothganger, Michael Krygier, Cale Crowder, **Bradley H Theilman**, William Chapman, Ryan Dellana, Mark Plagge, Efrain Gonzalez, Srideep Musuvathy

Aaron Hill, Shashank Misra, Yang Ho, Brady Taylor, Chris Allemang, Rich Lehoucq, Brian Franke, Ojas Parekh



UNIVERSITY of SOUTH FLORIDA
UC San Diego
OAK RIDGE National Laboratory
UTSA
NYU

U.S. DEPARTMENT OF ENERGY | Office of Science
LDRD LABORATORY DIRECTED RESEARCH & DEVELOPMENT
Sandia National Laboratories
ASC