Accelerating Scientific Machine Learning with AI Accelerators

Murali Emani,
Argonne Leadership Computing Facility
memani@anl.gov

ARGONNE TRAINING PROGRAM ON EXTREME-SCALE COMPUTING (ATPESC)
July 31, 2023
Surge of Scientific Machine Learning

- Simulations/surrogate models
  Replace, in part, or guide simulations with AI-driven surrogate models

- Data-driven models
  Use data to build models without simulations

- Co-design of experiments
  AI-driven experiments

Design infrastructure to facilitate and accelerate AI for Science (AI4S) applications
Integrating AI Systems in Facilities

Experimental Facility

Computing Facility

AI-Accelerators

Simulations

Data-driven Models
ALCF AI Testbed

https://www.alcf.anl.gov/alcf-ai-testbed

- Infrastructure of next-generation machines with AI hardware accelerators
- Provide a platform to evaluate usability and performance of AI4S applications
- Understand how to integrate AI systems with supercomputers to accelerate science
ALCF AI Testbed

https://www.alcf.anl.gov/alcf-ai-testbed

- Cerebras: 2 CS-2 nodes, each with 850,000 Cores, compute-intensive models
- SambaNova: DataScale SN30 8 nodes (8 SN30 RDUs per node) - 1TB mem per device, models with large memory footprint
- Graphcore: Bow Pod64 4 nodes (16 IPUs per node) - MIMD, irregular workloads such as graph neural networks
- GroqRack: 8 nodes, 8 GroqNodes per node - inference at batch 1
- Habana Gaudi1: 2 nodes, 8 cards per node - On-chip integration of RDMA over Converged Ethernet (RoCE2), scale-out efficiency
Director’s Discretionary (DD) awards support various project objectives from scaling code to preparing for future computing competition to production scientific computing in support of strategic partnerships.

Getting Started on ALCF AI Testbed:
Apply for a Director’s Discretionary (DD) Allocation Award

Cerebras CS-2, SambaNova Datascale SN30 and Graphcore Bow Pod64 are available for allocations

Allocation Request Form
AI Testbed User Guide
AI Testbed Community Engagement

- **AI training workshops**
  - Cerebras: [https://events.cels.anl.gov/event/420/](https://events.cels.anl.gov/event/420/)
  - SambaNova: [https://events.cels.anl.gov/event/421/](https://events.cels.anl.gov/event/421/)
  - Graphcore: [https://events.cels.anl.gov/event/422/](https://events.cels.anl.gov/event/422/)

**Tutorial at SC23** on Programming Novel AI accelerators for Scientific Computing *in collaboration with Cerebras, Intel Habana, Graphcore, Groq and SambaNova*
Dataflow Architectures

The old way: kernel-by-kernel
Bottlenecked by memory bandwidth and host overhead

The Dataflow way: Spatial
Eliminates memory traffic and overhead

Image Courtesy: SambaNova
SambaNova Cardinal SN30 RDU

7nm TSMC, 86B transistors
102 km of wire
640 MB on-chip, 1,024 GB external
688 TFLOPS (bf16)

RDU-Connect™

as-a-SERVICE
Pre-trained Foundation Models

SYSTEMS
DataScale®

SOFTWARE
SambaFlow™

SILICON
RDU

Image Courtesy: SambaNova
Cardinal SN30: Chip and Architecture Overview

- RDU broken up into 8-tiles
  - 160 PMU and PCUs per tile
  - Additional sub-components like coalescing units (CU) for connectivity to other tiles and off-chip components, switches to set up communication between PMU, PCUs, and CU

- Tile resource management: Combined or independent mode
  - Combined: Combine adjacent to form a larger logical tile for one application
  - Independent: Each tile controlled independently, allows running different applications on separate tiles concurrently.

- Direct access to TBs of DDR4 off-chip memory

- Memory-mapped access to host memory

- Scale-out communication support

Image Courtesy: SambaNova
**Cardinal SN30: Tile**

<table>
<thead>
<tr>
<th>TILE 0</th>
<th>TILE 1</th>
<th>TILE 2</th>
<th>TILE 3</th>
<th>TILE 5</th>
<th>TILE 6</th>
<th>TILE 7</th>
<th>TILE 8</th>
</tr>
</thead>
</table>

**Virtual Memory Manager**

**Top-Level Interconnect**

**DDR**

**PCle**

**DRAM (TBs)**

**Host Scale-Out**

---

**Software-Driven Architecture**

Tiled architecture with reconfigurable SIMD pipelines, distributed scratchpads, and programmed switches.

Image Courtesy: SambaNova
Dataflow Architecture for Terabyte Sized Models

Dataflow Efficiency + Compute Capability + Large Memory Capacity

DataScale SN30-8R

Image Courtesy: SambaNova
SambaNova DataScale SN30-8 System

- 8 x Cardinal SN30 Reconfigurable Dataflow Unit
- 8 TB total memory (using 64 x 128 GB DDR4 DIMMs)
- 6 x 3.8 TB NVMe (22.8 TB total)
- PCIe Gen4 x16
- Host module

Image Courtesy: SambaNova
SambaNova Datascale SN30

https://www.alcf.anl.gov/alcf-ai-testbed

- 4 Racks
- 8 nodes of SN30
- 8 RDUs or 4 XRDUs per node
- 8 Tiles per RDU
- Group of 4 tiles
SambaFlow Architecture

- Samba PyTorch API
  - Graph Compiler
    - Kernel Library
    - Kernel Compiler
  - Compilation Path
  - PEF
- Samba Runtime
  - Run Path
  - Runtime
    - RDUs
    - CPU

Image Courtesy: SambaNova
Cerebras Wafer-Scale Engine (WSE-2)

850,000 cores optimized for sparse linear algebra
46,225 \text{mm}^2 \text{ silicon}
2.6 trillion transistors
40 gigabytes of on-chip memory
20 PByte/s memory bandwidth
220 Pbit/s fabric bandwidth
7\text{nm} \text{ process technology}
Wafer-Scale Cluster

Input preprocessing servers stream training data

MemoryX - Stores and streams model’s weights

SwarmX – weight broadcasts and gradient across multiple CS2s

Compilation (maps graph to kernels) Execution (training)
Cerebras CS-2 Cluster
https://www.alcf.anl.gov/alcf-ai-testbed

ALCF’s CS-2 Cluster
- 2 CS-2 Appliances (each chip 46225 mm^2)
- 1 Management node
- 16 Worker nodes
- 24 MemoryX nodes
- 6 SwarmX nodes
- 3 user login nodes
Cerebras Weight Streaming Technology

Disaggregate storage and compute
Enable scaling model size

Image Courtesy: Cerebras
# Graphcore Intelligence Processing Unit (IPU)

<table>
<thead>
<tr>
<th></th>
<th>CPU</th>
<th>GPU</th>
<th>IPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parallelism</td>
<td>Designed for scalar processing</td>
<td>SIMD/SIMT architecture. Designed for large blocks of dense contiguous data</td>
<td>Massively parallel MIMD architecture. High performance/efficiency for future ML trends</td>
</tr>
<tr>
<td>Memory</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Processor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Memory</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Memory Bandwidth</td>
<td>Off-chip memory</td>
<td>Model and Data spread across off-chip and small on-chip cache and shared memory (2TB/s for A100 HBM)</td>
<td>Main Model &amp; Data in tightly coupled large locally distributed SRAM (~65 TB/s for Bow IPU)</td>
</tr>
</tbody>
</table>

Slide Courtesy: Graphcore
IPU-Tiles™
1472 independent IPU-Tiles™ each with an
IPU-Core™ and In-Processor-Memory™

IPU-Core™
1472 independent IPU-Core™
8832 independent program threads
executing in parallel

In-Processor-Memory™
900MB In-Processor-Memory™ per IPU
65TB/s memory bandwidth per IPU

IPU-Exchange™
11 TB/s all to all IPU-Exchange™
Non-blocking, any communication pattern

PCIe
PCI Gen4 x16
64 GB/s bidirectional bandwidth to host

IPU-Links™
10 x IPU-Links,
320GB/s chip to chip bandwidth

Slide Courtesy: Graphcore
SCALING ACROSS DEVICES

EXCHANGE
- THE IPU EXCHANGE MODEL EXTENDS OFF DEVICE
- ALLOWS TILE MESSAGING ACROSS IPU DEVICES
- SUPPORTS OPTIMIZED CROSS IPU SYNCHRONISATION
- DRIVEN BY COMPILER SUPPORT IN SOFTWARE

IPU-LINK™
- PROVIDES 320 GB/S IPU TO IPU BANDWIDTH
- SUPPORTS COMMUNICATION BETWEEN IPUS
- LAYOUT FULLY SOFTWARE CONFIGURABLE
- SUPPORTS POINT TO POINT TILE MESSAGING

GCD
- GRAPH CONTROL DOMAIN FOR APPLICATIONS
- CREATES A SINGLE LARGE IPU SOFTWARE TARGET
- FULLY CONFIGURABLE PARTITIONING OF IPUS
- BOTH DATA PARALLEL AND MODEL PARALLEL

UP TO 64 IPU DEVICES USABLE AS A SINGLE LARGE IPU FROM APPLICATIONS
565,248 FULLY INDEPENDENT WORKERS, 57.6GB IN-PROCESSOR MEMORY™, LEVERAGING OVER 3.8 TRILLION TRANSISTORS

Slide Courtesy: Graphcore
SCALING ACROSS SYSTEMS

- IPU EXCHANGE SUPPORT ACROSS DOMAINS
  - DRIVEN BY COMPILER SUPPORT IN SOFTWARE
  - ENABLES APPLICATION COLLECTIVES SUPPORT
  - ALLOWS SCALING UP TO 64000 IPU DEVICES

- IPU-LINK™ CAN BE EXTENDED ACROSS DOMAINS
  - SUPPORTS OPTIMIZED IPU LINK™ COLLECTIVES
  - ALLOWS REPLICATION ACROSS SYSTEMS
  - SUPPORTS A STANDARD IPU SOFTWARE MODEL

- IPUS CAN ACCESS MEMORY AND DEVICES OVER PCIE
  - ALLOWS INTERFACING WITH HOST BASED SOFTWARE
  - APPLICATIONS CAN BUILD ON HOST NETWORKING
  - ALLOWS SCALING IN STANDARD SERVER PLATFORMS

Slide Courtesy: Graphcore
BOW-2000: THE BUILDING BLOCK OF LARGE PODS

4x Bow IPUs
- 1.4 PFLOP\textsubscript{sp} compute
- 5,888 processor cores
- > 35,000 independent parallel threads

Exchange Memory
- 3.6GB In-Processor Memory @ 260 TB/s
- 128GB Streaming Memory DRAM (up to 256GB) @ 20 GB/s

IPU-Fabric managed by IPU-GW
- Host-Link - 100GE to Poplar Server for standard data center networking
- IPU-Link - 2D Torus for intra-POD64 communication
- GW-Link - 2x 100Gbps Gateway-Links for rack-to-rack - flexible topology

Slide Courtesy: Graphcore
Graphcore POD-64

https://www.alcf.anl.gov/alcf-ai-testbed

- 4 Nodes
- 64 IPUs
<table>
<thead>
<tr>
<th></th>
<th>Cerebras CS2</th>
<th>SambaNova Cardinal SN30</th>
<th>Groq GroqRack</th>
<th>GraphCore GC200 IPU</th>
<th>Habana Gaudi1</th>
<th>NVIDIA A100</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Compute Units</strong></td>
<td>850,000 Cores</td>
<td>640 PCUs</td>
<td>5120 vector ALUs</td>
<td>1472 IPUs</td>
<td>8 TPC + GEMM engine</td>
<td>6912 Cuda Cores</td>
</tr>
<tr>
<td><strong>On-Chip Memory</strong></td>
<td>40 GB L1, 1TB</td>
<td>&gt;300MB L1, 1TB</td>
<td>230MB L1</td>
<td>900MB L1</td>
<td>24 MB L1, 32GB</td>
<td>192KB L1, 40-80GB</td>
</tr>
<tr>
<td><strong>Process System Size</strong></td>
<td>7nm</td>
<td>7nm</td>
<td>7 nm</td>
<td>7nm</td>
<td>7nm</td>
<td>7nm</td>
</tr>
<tr>
<td><strong>System Size</strong></td>
<td>2 Nodes incluing Memory-X and Swarm-X</td>
<td>8 nodes (8 cards per node)</td>
<td>9 nodes (16 cards per node)</td>
<td>4 nodes (8 cards per node)</td>
<td>2 nodes (8 cards per node)</td>
<td>Several systems</td>
</tr>
<tr>
<td><strong>Estimated Performance of a card (TFlops)</strong></td>
<td>&gt;5780 (FP16)</td>
<td>&gt;660 (BF16)</td>
<td>&gt;250 (FP16)</td>
<td>&gt;250 (FP16)</td>
<td>&gt;150 (FP16)</td>
<td>312 (FP16), 156 (FP32)</td>
</tr>
<tr>
<td><strong>Software Stack Support</strong></td>
<td>Tensorflow, Pytorch</td>
<td>SambaFlow, Pytorch</td>
<td>GroqAPI, ONNX</td>
<td>Tensorflow, Pytorch, PopArt</td>
<td>Synapse AI, TensorFlow and PyTorch</td>
<td>Tensorflow, Pytorch, etc</td>
</tr>
<tr>
<td><strong>Interconnect</strong></td>
<td>Ethernet-based</td>
<td>Ethernet-based</td>
<td>RealScale™</td>
<td>IPU Link</td>
<td>Ethernet-based</td>
<td>NVLink</td>
</tr>
</tbody>
</table>
Challenges

• Understand how these systems perform for different workloads given diverse hardware and software characteristics
• What are the unique capabilities of each evaluated system
• Opportunities and potential for integrating AI accelerators with HPC computing facilities
Approach

• Perform a comprehensive evaluation with a diverse set of Deep Learning (DL) models:
  ▪ **DL primitives**: GEMM, Conv2D, ReLU, and RNN
  ▪ **Benchmarks**: U-Net, BERT-Large, ResNet-50
  ▪ **AI4S applications**: BraggNN and Uno
  ▪ Scalability and Collective communications

• Evaluated SambaNova, Cerebras, Graphcore, Groq systems and Nvidia A100 as a baseline*

---


* run out-of-box.
Scaling UNet-2D Training

- 256x256 BMRI dataset
- A100, SN10 – Pytorch
- Graphcore – Tensorflow
- CS2 -- TF estimator
- Accelerators capable of handling much larger image sizes.

GraphCore uses data-prefetching optimization, CS-2 uses 1 wafer-scale engine.

Scale across 1, 2, 4, and 8 devices with two batch sizes (BS)

GraphCore uses data-prefetching optimization, CS-2 uses 1 wafer-scale engine
Scaling UNet-2D Training

Increased Throughput over 8 A100s

<table>
<thead>
<tr>
<th>Batch Size</th>
<th>8 SN10 - RDUs</th>
<th>1 CS2</th>
<th>8 GC 200 IPUs</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>2.1x</td>
<td>4.9x</td>
<td>10x</td>
</tr>
</tbody>
</table>

GraphCore uses data-prefetching optimization, CS-2 uses 1 wafer-scale engine.
Scaling UNet-2D Training

Scaling efficiency

<table>
<thead>
<tr>
<th>Batch Size</th>
<th>A100</th>
<th>SN10</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>18.8%</td>
<td>42%</td>
<td>79.5%</td>
</tr>
<tr>
<td>256</td>
<td>52%</td>
<td>28%</td>
<td>79.6%</td>
</tr>
</tbody>
</table>
GPT Small (1.5B / 2B / xl)

**SN30**

Sequence length = 1024, fits on 4 tiles.

Maximum of 128 instances run on 8 SN30 nodes.

~ 2x speed up observed over the nodes.

**Graphcore**

Scaling over replication factor better for smaller GAS values.

poprun for improved scaling efficiency.

**CS2**

Work in progress for higher sequence length and wider models.
Missing a real disruption (false negative) can be costly because of the damaging effects of a disruption, while triggering a false positive alarm wastes valuable experimental time and resources. Setting the threshold allows a tradeoff between these two and more..

Figure 1: Data flow and summary of the FRNN algorithm

4.2 Reconstruction Error Analysis

§4.1 discussed the direct model performance on peak localization. Since the 3D reconstruction is our final goal, we also do reconstruction using peaks position located by the proposed BraggNN and the conventional Voigt profiling separately. Figure 5 compares the positions of about 400 grains reconstructed separately using Bragg peaks localized by BraggNN and conventional 2D pseudo-Voigt profiling. The fact that the deviation directions are uniformly distributed indicates that BraggNN is

Figure 5: A comparison of grains in 3D space. Each ball represents one grain reconstructed by using the conventional method, with color indicating the grain size (µm). An arrow indicates a deviation from a grain to the corresponding grain reconstructed by using the BraggNN estimated peak.

Cancer drug response prediction

Imaging Sciences-Braggs Peak

Tokomak Fusion Reactor operations

Protein-folding (Image: NCI)

and more..
Fast X-Ray Bragg Peak Analysis

**Goal:** Enable rapid analysis and real-time feedback during an in-situ experiment with complex detector technologies

**Proposed Approach:** Deep learning-based method, BraggNN, for massive extraction of precise Bragg peak locations from far-field high energy diffraction microscopy data. BraggNN has achieved 200X improvement over conventional pseudo-Voigt profiling

**Challenges:** Model training capability is limited by the hardware

Application of the BraggNN deep neural network to an input patch yields a peak center position ($y$, $z$). All convolutions are 2D of size $3 \times 3$, with rectifier as activation function. Each fully connected layer, except for the output layer, also has a rectifier activation function.

A comparison of BraggNN, pseudo-Voigt FF-HEDM and NF-HEDM. (a) Grain positions from NF-HEDM (black squares), pseudo-Voigt FF-HEDM (red circles) and BraggNN FF-HEDM (blue triangles) overlaid on NF-HEDM confidence map

**Figure 6:** A comparison of BraggNN, pseudo-Voigt FF-HEDM and NF-HEDM. (a) Grain positions from NF-HEDM (black squares), pseudo-Voigt FF-HEDM (red circles) and BraggNN FF-HEDM (blue triangles) overlaid on NF-HEDM confidence map. (b-c) Difference in position of grains between pseudo-Voigt FF-HEDM (b), BraggNN (c) and NF-HEDM as a function of Grain Size. Color of markers in (b-c) represent the mean difference in position of expected and observed diffraction spots. Size of markers in (b-c) represent the mean Internal Angle (see text).

CNN layers better extract feature representation in the latent space for fully-connected layers to better approximate its center [Wang et al., 2018]. Here, we conduct an ablation study to show its effectiveness. We train two models, one with attention block one without, using the same datasets, i.e., attention block is the only difference, and then we evaluate their estimation accuracy. Fig. 7 shows the distribution of deviations. It is clear that both the 50th and 75th percentile deviations are more than 20% worse than Fig. 4(c) where BraggNN has the non-local self-attention block, the 95th percentile is about 15% worse.

**4.2 Data Augmentation**

We presented a novel data augmentation method to prevent model over-fitting and to address inaccurate patch cropping using the connect component in the model inference phase. In order to study its effectiveness, we trained BraggNN on a simulation dataset with and without augmentation. When trained with augmentation, we use an interval of $[1, 1]$ for both $m$ and $n$. Fig. 8 demonstrates three arbitrarily selected cases in our test dataset where the computed peak location deviated from the corresponding patch’s geometric center (i.e., $(5, 5)$ for a $11 \times 11$ pixel patch) in different directions. We can see from the demonstration that BraggNN is able to locate the peak values precisely even when the peak is deviated from the geometric center.

In order to quantitatively evaluate the effectiveness of data augmentation, we sample $m$ and $n$ independently from $\{-1, 0, 1\}$ when preparing our test dataset to mimic imperfect patch cropping. That is, only $1/3 \times 1/3 = 1/9$ of the patches have maxima at the geometric center. Fig. 9 compares the prediction error on the test dataset in a statistical way. Comparing Fig. 9(a) with Fig. 9(b), we see clear improvement when augmentation is applied for model training. The 50th, 75th, and 95th percentile errors are all reduced to about 20% of those obtained when BraggNN is trained without data augmentation: a five times improvement.

**5 Conclusions and Future work**

We have described BraggNN, the first machine learning-based method for precisely characterizing Bragg diffraction peaks in HEDM images. When compared with conventional 2D pseudo-Voigt fitting and higher resolution nf-HEDM, BraggNN has achieved 200X improvement over conventional pseudo-Voigt profiling.
Fast X-Ray Bragg Peak Analysis

End-to-End Execution time (lower is better)

- Fixed Time (compile, I/O and pre-processing)
- Training Time

### TABLE II: BraggNN Throughput (in order of 1k samples/sec) with various batch sizes (BS)

<table>
<thead>
<tr>
<th>System</th>
<th>BS=512</th>
<th>BS=1024</th>
<th>BS=2048</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS-2 (FP16)</td>
<td>1365.4</td>
<td>2463.4</td>
<td>2787.9</td>
</tr>
<tr>
<td>GC200 IPU (FP16)</td>
<td>478.0</td>
<td>350.6</td>
<td>219.9</td>
</tr>
<tr>
<td>SN10 RDU (BF16)</td>
<td>369.7</td>
<td>449.8</td>
<td>518.0</td>
</tr>
<tr>
<td>A100 (FP16)</td>
<td>53.9</td>
<td>65.5</td>
<td>73.7</td>
</tr>
</tbody>
</table>

- SambaNova and Graphcore achieve lowest time to solution and achieve up to 1.55x and 1.46x speedup in comparison to Nvidia A100 respectively.
- Cerebras achieves up to 37.8x throughput improvement over A100.
Genome-scale Language Models (GenSLMs)

Goal:
- How new and emergent variants of pandemic causing viruses, (specifically SARS-CoV-2) can be identified and classified.
- Identify mutations that are VOC (increased severity and transmissibility)
- Extendable to gene or protein synthesis.

Approach
- Adapt Large Language Models (LLMs) to learn the evolution.
- Pretrain 25M – 25B models on raw nucleotides with large sequence lengths.
- Scale on GPUs, CS2s, SN30.

GenSLMs: Genome-scale language models reveal SARS-CoV-2 evolutionary dynamics
Winner of the ACM Gordon Bell Special Prize for High Performance Computing-Based COVID-19 Research, 2022,
DOI: https://doi.org/10.1101/2022.10.10.511571
Genome-scale Language Models (GenSLMs)

Challenges
Scaling LLMs with 25B parameters:
- $O(L^2)$ complexity in the attention computation
- Overcome communication overheads
- Sharding and the training time available on GPUs imposing limitations

Solution
Cerebras CS-2 wafer-scale cluster and Sambanova SN30 enables pre-training and finetuning.

<table>
<thead>
<tr>
<th>Model</th>
<th>Seq. length</th>
<th>#Parameters</th>
<th>Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>GenSLM-Foundation</td>
<td>2048</td>
<td>25M, 250M, 2.5B, 25B</td>
<td>110M</td>
</tr>
<tr>
<td>GenSLM</td>
<td>10240</td>
<td>25M, 250M, 2.5B, 25B</td>
<td>1.5M</td>
</tr>
<tr>
<td>GenSLM-Diffusion</td>
<td>10240</td>
<td>2.5B</td>
<td>1.5M</td>
</tr>
</tbody>
</table>
GenSLMs on CS2

- Sequence Length = 10,240
- Trainable upto GPT3-13b model.
- Training with 4CS2, less than $\frac{1}{2}$ day
GenSLMs on SN30

- Sequence Length = 1024
- Model Size 13B
- Achieves linear scaling across nodes.
- SN30 performance similar to 4 A100 on 1.17 release.
- Optimized on 1.18 to get 10x speed-up.
- Pretraining and FineTuning on larger sequence lengths.
Observations, Challenges and Insights

• Significant speedup achieved for a wide-gamut of scientific ML applications
  - Easier to deal with larger resolution data and to scale to multi-chip systems

• Room for improvement exists
  - Porting efforts and compilation times
  - Coverage of DL frameworks, support for performance analysis tools, debuggers
Observations, Challenges and Insights

- Good progress made in integration of AI accelerators, in production, at a national user facility and significant more work is needed for effective coupling

- Training and Outreach is critical to educate users to effectively use AI systems

- Close collaboration with vendors is necessary to realize the vision of AI for science
Ongoing Efforts

- Evaluate new AI accelerators offerings and incorporate promising solutions as part of the testbed
- Integrate AI testbed systems with the PBSPro scheduler to facilitate effective job scheduling across the accelerators
- Evaluate traditional HPC on AI Accelerators
- Understand how to integrate AI accelerators with ALCF’s existing and upcoming supercomputers to accelerate science insights
Recent Publications

• **GenSLMs: Genome-scale language models reveal SARS-CoV-2 evolutionary dynamics**
  **Winner of the ACM Gordon Bell Special Prize for High Performance Computing-Based COVID-19 Research, 2022,**
  DOI: https://doi.org/10.1101/2022.10.10.511571

• **A Comprehensive Evaluation of Novel AI Accelerators for Deep Learning Workloads**

• **Enabling real-time adaptation of machine learning models at x-ray Free Electron Laser facilities with high-speed training optimized computational hardware**
  Petro Junior Milan, Hongqian Rong, Craig Michaud, Naoufal Layad, Zhengchun Liu, Ryan Coffee, *Frontiers in Physics*
  DOI: https://doi.org/10.3389/fphy.2022.958120
Recent Publications

- **Intelligent Resolution: Integrating Cryo-EM with AI-driven Multi-resolution Simulations to Observe the SARS-CoV-2 Replication-Transcription Machinery in Action**

- **Stream-AI-MD: Streaming AI-driven Adaptive Molecular Simulations for Heterogeneous Computing Platforms**

- **Bridging Data Center AI Systems with Edge Computing for Actionable Information Retrieval**
  Zhengchun Liu, Ahsan Ali, Peter Kenesei, Antonino Miceli, Hemant Sharma, Nicholas Schwarz, Dennis Trujillo, Hyunseung Yoo, Ryan Coffee, Naoufal Layad, Jana Thayer, Ryan Herbst, Chunhong Yoon, and Ian Foster, 3rd Annual workshop on Extreme-scale Event-in-the-loop computing (XLOOP), 2021

- **Accelerating Scientific Applications With SambaNova Reconfigurable Dataflow Architecture**

* Finalist in the ACM Gordon Bell Special Prize for High Performance Computing-Based COVID-19 Research, 2021
Thank You

• This research was funded in part and used resources of the Argonne Leadership Computing Facility (ALCF), a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.

• Venkatram Vishwanath, Michael Papka, William Arnold, Bruce Wilson, Varuni Sastry, Sid Raskar, Zhen Xie, Rajeev Thakur, Anthony Avarca, Arvind Ramanathan, Alex Brace, Zhengchun Liu, Hyunseung (Harry) Yoo, Corey Adams, Ryan Aydelott, Kyle Felker, Craig Stacey, Tom Brettin, Rick Stevens, and many others have contributed to this material.

• Our current AI testbed system vendors – Cerebras, Graphcore, Groq, Intel Habana and SambaNova. There are ongoing engagements with other vendors.

Please reach out for further details
Venkat Vishwanath, Venkat@anl.gov
Murali Emani, memani@anl.gov