Department: Novel Architectures
Editors: Volodymyr Kindratenko, kindr@ncsa.uiuc.edu
Anne Elster, anne.elster@gmail.com

Accelerating Scientific Applications with SambaNova Reconfigurable Dataflow Architecture

Murali Emani,Venkatram Vishwanath, Corey Adams, Michael E. Papka, Rick Stevens
Argonne National Laboratory

Laura Florescu, Sumti Jairath, William Liu, Tejas Nama, Arvind Sujeeth
SambaNova Systems

ARTIFICIAL INTELLIGENCE (AI)-DRIVEN SCIENCE is an integral component in several science domains such as Materials, Biology, High Energy Physics and Smart Energy. Science workflows can span one or more computational, observational and experimental systems. The AI for Science report [1] put forth by a wide community of stakeholders from national laboratories, academia and industry collectively stress the need for a tighter integration of AI infrastructure ecosystem with experimental and leadership computing facilities. The AI component in science applications, which generally deploy deep learning (DL) models, are unique and exhibit different characteristics from traditional industrial workloads. They implement complex models and consists typically hundreds of millions of model parameters. Data from simulations is usually sparse, multi-modal, multi-dimensional and exhibit temporal and spatial correlations. Moreover, AI-driven science applications benefit from flexible coupling of simulations with DL training or inference.

Such complexity of the AI for science workloads with increasingly large deep learning models is typically limited by the traditional computing architectures. The adoption of novel AI architectures and systems aimed to accelerate machine learning models is critical to reduce the time-to-discovery for science.

Argonne Leadership Computing Facility (ALCF), a US Department of Energy Office of Science user facility, provides supercomputing resources to power scientific breakthroughs. Applications with significant DL components are increasingly being run on existing supercomputers. Adoption of novel AI-hardware systems, such as SambaNova, and leveraging

 Published by the IEEE Computer Society

their unique capabilities will try to address the challenges in scaling the performance of AI models in the science applications and hence, further accelerate scientific insights.

KEY ATTRIBUTES FOR A NEXT-GENERATION ARCHITECTURE

Through academic research, analysis of technology trends and knowledge developed in the design process, SambaNova identified the following key attributes to enable highly efficient dataflow processing.

- Native dataflow — Commonly occurring operators in machine learning frameworks and domain-specific languages (DSL) can be described in terms of parallel patterns that capture parallelizable computation on both dense and sparse data collections along with corresponding memory access patterns. This enables exploitation and high utilization of the underlying platform while allowing a diverse set of models to be easily written in any machine learning framework of choice.
- Support for terabyte-sized models — A key trend in deep-learning model development uses increasingly large model sizes to gain higher accuracy and deliver more sophisticated functionality. For example, leveraging billions of datapoints (referred to as parameters) enables more accurate natural language generation. In the life sciences field, analyzing tissue samples requires the processing of large, high-resolution images to identify subtle features. Providing much larger on-chip and off-chip memory stores than those that are available on core-based architectures will accelerate deep-learning innovation.
- Efficient processing of sparse data and graph-based networks — Recommender systems, friend-of-friends problems, knowledge graphs, some life science domains and more involve large sparse data structures that consist of mostly zero values. Moving around and processing large, mostly empty matrices is inefficient and degrades performance. A next generation architecture must intelligently avoid unnecessary processing.
- Flexible model mapping — Currently, data and model parallel techniques are used to scale workloads across the infrastructure. However, the programming cost and complexity are often prohibiting factors for new deep-learning approaches. A new architecture should automatically enable scaling across infrastructure without this added development and orchestration complexity and avoid the need for model developers to become experts in system architecture and parallel computing.
- Incorporate SQL and other pre-/post data processing — As deep learning models grow and incorporate a wider variety of data types, the dependency on pre-processing and post-processing of data becomes dominant. Additionally, the time lag and cost of ETL operations impact real-time system goals. A new architecture should allow the unification of these processing tasks on a single platform.

A New Approach: SambaNova Reconfigurable Dataflow Architecture™

The SambaNova Reconfigurable Dataflow Architecture (RDA) is a computing architecture designed to enable the next generation of machine learning and high performance computing applications. The Reconfigurable Dataflow Architecture is a complete, full-stack solution that incorporates innovations at all layers including algorithms, compilers, system architecture and state-of-the-art silicon.

The RDA provides a flexible, dataflow execution model that pipelines operations, enables programmable data access patterns and minimizes excess data movement found in fixed, core-based, instruction set architectures. It does not have a fixed Instruction Set Architecture (ISA) like traditional architectures, but instead is programmed specifically for each model resulting in a highly optimized, application-specific accelerator.

The Reconfigurable Dataflow Architecture is composed of the following:

SambaNova Reconfigurable Dataflow Unit (RDU) is a next-generation processor designed to provide native dataflow processing and programmable acceleration. It has a tiled architecture that comprises a network of reconfigurable functional units. The architecture enables a broad set of highly parallelizable patterns contained within dataflow graphs to be efficiently programmed as a combination of compute, memory and commu-

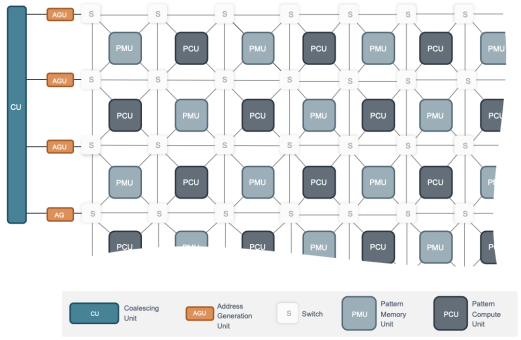

Figure 1. SambaNova Reconfigurable Dataflow Unit (RDU)

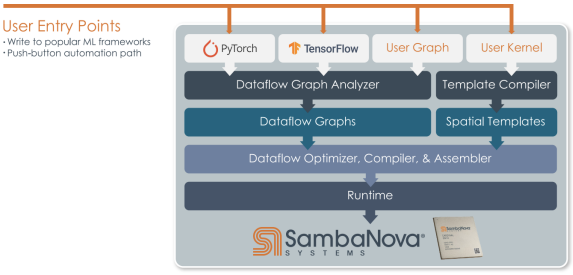

Figure 2. SambaFlow

nication networks.

The RDU is the engine that efficiently executes dataflow graphs. It consists of a tiled array of reconfigurable processing and memory units connected through a high-speed, three-dimensional on-chip switching fabric. When an application is started, SambaNova Systems SambaFlow™ software configures the RDU elements to execute an optimized dataflow graph for that specific application. Figure 1 shows a small portion of an RDU with its components described below.

Pattern Compute Unit (PCU) — The PCU is designed to execute a single, innermost-parallel operation in an application. The PCU datapath is organized as a multi-stage, reconfigured SIMD pipeline. This design enables each PCU to achieve high compute density and exploit both loop-level parallelism across lanes and pipeline parallelism across stages.

Pattern Memory Unit (PMU) — PMUs are highly specialized scratchpads that provide on-chip memory capacity and perform a number of specialized intelligent functions. The high PMU capacity and distribution throughout the PCUs minimizes data movement, reduces latency, increases bandwidth and avoids off-chip memory accesses.

Switching Fabric — The high-speed switching fabric that connects PCUs and PMUs is composed of three switching networks: scalar, vector and control. These switches form a three-dimensional network that runs in parallel to the rest of the units within an RDU. The networks differ in granularity of data being transferred; scalar networks operate at word-level granularity, vector networks at multiple word-level granularity and control at bit-level granularity.

Address Generator Units (AGU) and Coalescing Units (CU) — AGUs and CUs provide the interconnect between RDUs and the rest of the system, including off-chip DRAM, other RDUs and the host processor. RDU-Connect™ provides a high-speed path between RDUs for efficient processing of problems that are larger than a single RDU. The AGUs and CUs working together with the PMUs enable RDA to efficiently process sparse and graph-based datasets.

Reconfigurability, exploitation of parallelism at multiple levels and the elimination of instruction processing overhead gives RDUs their significant performance advantages over traditional architectures.

SambaFlow is a complete software stack designed to take input from standard machine learning frameworks such as PyTorch and TensorFlow. SambaFlow automatically extracts, optimizes and maps dataflow graphs onto RDUs, allowing high performance to be obtained without the need for low-level kernel tuning. SambaFlow also provides an API for expert users and those who are interested in leveraging the RDA for workloads beyond machine learning. Figure 2 shows the components of SambaFlow and its components described below.

User Entry Points – SambaFlow supports the common open-source, machine learning frameworks, PyTorch and TensorFlow. Serialized graphs from other frameworks and tools are also imported here.

Dataflow Graph Analyzer and Dataflow Graphs — Accepts models from the frameworks then analyzes the model to extract the dataflow

graph. For each operator, the computation and communication requirements are determined, so the appropriate RDU resources can be allocated later. The analyzer determines the most efficient mappings of the operators and communication patterns to the RDU utilizing the spatial programming model. With knowledge of both the model architecture and the RDU architecture, the analyzer can also perform high-level, domain-specific optimizations like node fusion. The output of the Dataflow Graph Analyzer is an annotated Dataflow Graph that serves as the first intermediate representation (IR) passed to the Dataflow Compiler.

Template Compiler and Spatial Templates — For cases where operators are required but not available in the existing frameworks, new operators can be described via a high-level, tensor index notation API. The Template Compiler will then analyze the operator and generate an optimized dataflow implementation for the RDU, called a Spatial Template. The generated template includes bindings that enable the new operator to be used directly from application code in the same way as built-in framework operators.

Dataflow Compiler, Optimizer and Assembler — This layer receives annotated Dataflow Graphs and performs high-level transformations like meta-pipelining, multi-section support and parallelization. It also understands the RDU hardware attributes and performs low-level transforms, primarily placing and routing by mapping the graph onto the physical RDU hardware and then outputting an executable file. As before, a spatial programming approach is used to determine the most efficient location of RDU resources.

SambaNova Systems DataScale™ is a complete, rack-level, data-center-ready accelerated computing system. Each DataScale system configuration consists of one or more DataScale nodes, integrated networking and management infrastructure in a standards-compliant data center rack, referred to as the DataScale SN10-8R. Additionally, SambaNova DataScale leverages open standards and common form factors to ease adoption and streamline deployment.

Deployment at Argonne

SambaNova DataScale system deployed at Argonne Leadership Computing Facility (ALCF) is a DataScale SN108-R system consisting of two SN10-8 systems. Each SN10-T8 system consists of a host module and 8 RDUs. The RDUs on a system are interconnected via the RDU-Connect and the systems are interconnected using an Infiniband-based interconnect. These together enable both model parallelism as well as data parallelism across the RDUs in the system.

We evaluated the SambaNova system with a diverse range of deep learning application models of interest to science. These application models also exhibit diverse characteristics in terms of the model architectures and parameters. Additionally, BERT (Bidirectional Encoder Representations from Transformers) model was also evaluated.

CANDLE Uno: The Exascale Deep Learning and Simulation Enabled Precision Medicine for Cancer project (CANDLE) [2] implements deep learning architectures that are relevant to problems in cancer. These architectures address problems at three biological scales: cellular , molecular and population. The goal of the **Uno** model, part of the CANDLE project, is to build neural network-based models to predict tumor response to single and paired drugs, based on molecular features of tumor cells. It implements a deep learning architecture with 21 million parameters.

The Uno model performs extremely well on the RDU for a variety reasons. First, the model has a large number of parameters, which can be served directly from on-chip SRAM. The RDU has 100s of TB/s of bandwidth for repeated use in the network, which is much higher bandwidth than what is provided in other architectures. Secondly, the model has a reasonable number of non-systolic operations. SambaFlow constructs a dataflow graph from these operations and schedules sections of the computational graph, providing very high efficiency in executing these operations, without requiring any manual kernel fusion.

UNet: UNet [6] is a modified convolutional network architecture for fast and precise segmentation of images with fewer training samples. The upsampling operators in the model layers increase the resolution of the output. This model

is commonly used in segmentation in imaging science applications, such as in accelerators and connectomics.

Similar to the Uno model, UNet also performs much better on the RDU than traditional architectures. The large memory capacity of the RDU, which starts at 3TB and goes to 12TB per 8 RDUs, enables the RDU to handle hi-resolution images natively, without any compromise on image quality or batch-size. Additional, the data-flow architecture of the RDU provides for computation over overlapping pixels on the same device, without introducing any communication latency.

CosmicTagger: CosmicTagger application [3] in high energy particle physics domain deals with detecting neutrino interactions in a detector overwhelmed by cosmic particles. The goal is to classify each pixel to separate cosmic pixels, background pixels, and neutrino pixels in a neutrinos dataset. This uses multiple 2D projections of the same 3D particle tracks and the raw data is 3 images per event. The training model is a modified UResNet architecture for multi-plane semantic segmentation and is available in both single node and distributed-memory multi-node implementations.

Due to the high resolution images in the neutrino detectors, the memory requirements for GPU training of the CosmicTagger application are high enough to exceed Nvidia V100 memory sizes in most configurations at full image resolution. In this work, we demonstrate that the spatial partition of SambaFlow allows training at full resolution (as opposed to downsampled images on GPUs), leading to an improvement of state-of-the-art accuracy (mIoU) as seen in Figure 3.

Gravwaves: Multimessenger Astrophysics project aims to observe astrophysical phenomena using gravitational waves and requires large-scale computing [4]. This is achieved by the development of algorithms that significantly increase the depth and speed of gravitational wave searches and one that can process terabyte-size datasets of telescope images in real-time. The model has a root-leaf architecture. The shared root component of the model is composed of seven convolutional layers, and its output is shared by the leaves. There are multiple leaf parts in the model for individual parameters. Each leaf part consists of multiple sequential fully connected layers with

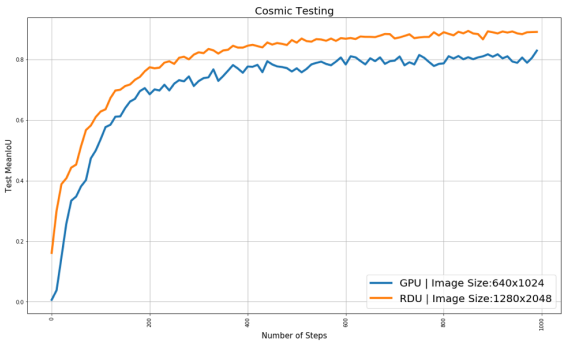

Figure 3. Using full resolution data that is out of memory on V100 GPUs, the CosmicTagger network is able to exceed state of the art accuracy on the test set.

ReLU, identity, and TanH activations. The neural network structures are composed of a general feature extractor for the first seven layers. The sub-networks learn specialized features for all different physical parameters.

Gravwaves is another network that performs well on the RDU since the compute to communication ratio is low. On traditional architectures, the kernel-by-kernel execution method loses a lot of efficiency in scheduling the kernels on the device. Since the RDU schedules the whole graph on the device, a much higher execution efficiency is achieved. Additionally, the RDU data-flow architecture implements convolutions from various building blocks, which allows for the execution of more exotic convolution operations with the same high efficiency as standard convolution operations.

BERT: BERT [5] is a neural network-based technique for natural language processing (NLP) pre-training called Bidirectional Encoder Representations from Transformers (BERT). BERT makes use of Transformer, an attention mechanism that learns contextual relations between words. These architectures are being pursued to mine scientific literature in domains including biosciences, material science, among others. BERT involves various deep learning kernels and has 110-340 million parameters.

Similar to the Uno model, BERT see benefits on the RDU due to the large number of parameters, which are served from on- chip SRAM, and how the RDU constructs and schedules the graph at a high level of efficiency.

The latest updates and innovations from SambaNova can be found at https://sambanova.ai/articles/.

CONCLUSION

SambaNova Reconfigurable Dataflow Architecture along with the SambaFlow software stack provides for an attractive system and solution to accelerate AI for science workloads. We have demonstrated the efficacy of using the system with a diverse set of science applications and reasoned their suitability for performance gains over traditional hardware. As the DataScale system provides for a very large memory capacity, the system can be used to train models that typically do not fit in a GPU. The architecture also provides for deeper integration with upcoming supercomputers at the Argonne Leadership Computing Facility to help advance science insights.

ACKNOWLEDGMENT

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

REFERENCES

1. AI for Science Technical Report. [Online]. Available: https://www.anl.gov/ai-for-science-report (URL)
2. Exascale Deep Learning and Simulation Enabled Precision Medicine for Cancer. [Online]. Available: https://candle.cels.anl.gov/ (URL)
3. Neutrino and Cosmic Tagging with UNet [Online]. Available: https://github.com/coreyjadams/CosmicTagger/ (URL)
4. Deep Learning at Scale for Multimessenger Astrophysics. [Online]. Available: https://www.alcf.anl.gov/science/projects/deep-learning-scale-multimessenger-astrophysics-through-ncsa-argonne-collaboration/ (URL)
5. J. Delvin, M. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding",NAACL-HLT, page 4171-4186. Association for Computational Linguistics, (2019)
6. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015 (May 2015)

Murali Emani is a computer scientist in the Datascience group with Argonne Leadership Computing Facility. He received his Ph.D. degree in Informatics from the University of Edinburgh, UK. Contact him at memani@anl.gov.

Venkatram Vishwanath is a computer scientist and head of the Datascience group with Argonne Leadership Computing Facility. He received his Ph.D. degree in Computer science from the University of Illinois, Chicago. Contact him at venkat@anl.gov.

Corey Adams is a computer scientist in the Datascience group with Argonne Leadership Computing Facility. He received his Ph.D. degree from Yale University. Contact him at corey.adams@anl.gov.

Michael E. Papka is a senior scientist at Argonne National Laboratory and division director of the Argonne Leadership Computing Facility. He received a Ph.D. degree in computer science from the University of Chicago. Contact him at papka@anl.gov.

Rick Stevens is the Associate Laboratory Director for Computing, Environment and Life Sciences directorate at Argonne National Laboratory. He received his Ph.D. degree in Computer Science from Northwestern University. Contact him at stevens@anl.gov.

Laura Florescu is a Principal Engineer at SambaNova Systems. She received her Ph.D. in Computer Science from New York University. She can be contacted at laura.florescu@sambanova.ai.

Sumti Jairath is a Chief Architect at SambaNova Systems. He can be reached at sumti.jairath@sambanova.ai.

William Liu is a Software Engineer at SambaNova Systems. He received his Bachelor's in Cognitive Science from Carnegie Mellon University. He can be reached at william.liu@sambanova.ai.

Tejas Nama is a Senior Machine Learning Engineer at SambaNova Systems. He received his Master's from Carnegie Mellon University in Computational Data Science. Contact him at tejas.nama@sambanova.ai.

Arvind Sujeeth is a Senior Director Software Engineering at SambaNova Systems. He received his Ph.D. in Electrical Engineering from Stanford University. Contact him at arvind.sujeeth@sambanova.ai.