# Accelerating Scientific Discovery with SCAIGATE Science Gateway

Chao Jiang[1], David Ojika[1], Bhavesh Patel[2], Ann Gordon-Ross[1], Herman Lam[1]
[1]NSF Center for Space, High-Performance and Resilient Computing
[2]DELL EMC
jc19chaoj, davido, anngordonross, hlam{@ufl.edu} bhavesh.a.patel@dell.com

***Abstract*— The demand for computational accelerators (GPUs, FPGAs, ASICs, etc.) is growing due to the widening variety of datacenter applications fueled by recent scientific breakthroughs that leverage artificial intelligence (AI). As much as these applications (e.g., cosmology, physics, etc.) have continued to witness record-breaking accuracy in predictive capabilities due to AI widespread influence, the infrastructure and workflow to take these applications out of research labs into production and business use-cases continues to lag. To address these important infrastructural challenges, we present SCAIGATE, a prototype science gateway with a simplified workflow aimed at facilitating model building/validation workflows in large-scale scientific applications.**

## I. Introduction

Deep neural networks (DNNs) are witnessing explosive growth in big data analytics applications [1]. While CPUs and GPUs have been widely used for DNN inference (the task of predicting fast, accurate results) inference engines accelerated with FPGAs have recently emerged. Recent improvements in FPGA technologies greatly increased the performance for DNN applications, e.g., with a reported performance of 9.2 TFLOPS for Intel Agilex with Stratix 10 FPGA [2]. Furthermore, FPGAs have other advantages important to many mission-critical applications such as deterministic low latency, energy efficiency, and re-configurability. As a result, the amount of research and development on accelerating DNNs on FPGAs and other accelerators in recent years has grown, demonstrating great interest in both academia and industry.

While some of these DNN acceleration works focus on optimizing DNN graphs for accelerator devices (e.g., TVM), other commercial ones focus on providing a more generalized FPGA platform for developers to build their custom applications [3]. Yet, other academic research efforts such as HGC framework [4] focus on the tools for accelerating domain-specific applications, e.g., cosmology, physics, etc.

As much as these applications have continued to witness record-breaking accuracy in predictive capabilities, the workflow to take these applications out of research labs into production use-cases continues to lag. Science gateways with community-based access to shared, distributed, advanced technologies and workflows present an opportunity to address these domain-centric infrastructural challenges.

## II. The SCAIGATE Science Gateway

SCAIGATE is a science gateway that integrates FPGAs and DNNs to facilitate machine learning through data preprocessing, training, and inference [5]. Using a set of software building blocks (Fig 1), SCAIGATE helps computational scientists and researchers accelerate their data analyses workflows at a fraction of the processing time and effort as compared to existing systems. The gateway also supports the integration of custom scientific workflows, allowing for the rapid acceleration of scientific applications with reconfigurable architectures.

The SCAIGATE prototype, depicted in Fig 1, consists of three main layers: (1) FPGA accelerators, (2) a workflow management framework, and (3) a gateway interface for community-based access. The rest of this paper focuses on using the HGC framework as the workflow management layer within SCAIGATE.

## III. Managing Workflows with HGC

A major shortcoming of many scientific workflows is limited interoperability, lack of component reusability, and curbed portability to new, advanced hardware (e.g.,

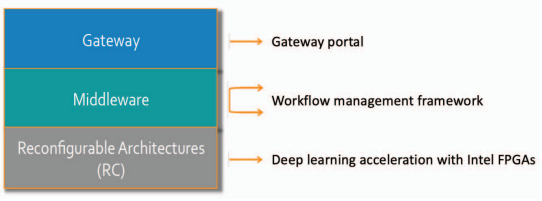


Fig 1. SCAIGATE's ecosystem.

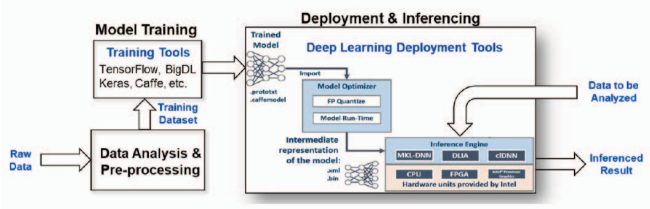


Fig 2. Overview of the HGC framework.

978-1-7281-2451-3/19/$31.00 ©2019 IEEE
DOI 10.1109/eScience.2019.00085

FPGAs). By abstracting key data processing workflows (data preprocessing, deep learning training, and deep learning inference) the HGC framework facilitates scientific deep learning model development, end-to-end, from data preparation and model training to model deployment as summarized in Fig 2. As shown, the framework consists of three stages. First, the Data Analysis and Pre-processing stage converts raw data from an application of interest into a form that is suitable for model training using any of the training frameworks. Next, the preprocessed data is used as inputs to the training tools in the Model Training stage. The output of the Training stage are the trained models which are then forwarded to the inference engines in the Deployment and Inferencing stage, the final stage of the workflow.

In the following section, we describe two representative scientific applications and their inference performance at the completion of the workflow stages.

## IV. Application Benchmarks

### A. HEP-CNN

HEP-CNN is a variation of the AlexNet model for high-energy physics (HEP) [6]. Trained with the ADAM optimizer, it comprises 5 convolution layers with ReLU activation functions. The kernel and stride sizes are 3x3 and 1x1 respectively, and it employs 128 filters per layer. The final set of layers consists of an average pooling across the dimensions output image followed by a fully connected layer with softmax activation which performs binary classification.

### A. CosmoGAN

CosmoGAN is a deep convolutional generative adversarial network (DC-GAN) which was designed to serve as an emulator for cosmology and weather simulations [7]. The network input is a 64-dimensional vector of uncorrelated gaussian noise, followed by a fully-connected layer to cross-correlate all inputs, followed by a series of 4 transpose convolutions, leading to a single 256x256 output image. Each inner layer is batch-normalized and uses ReLU activation while the output layer uses a Tanh activation.

## IV. Results

Leveraging SCAIGATE's microservices design, we note that the design introduces little to no overhead in an event of a rare occurrence of component/microservice failure. It takes less than a second to restart a component.

We setup the HGC framework on an Intel-based Skylake Gold CPU on a Dell EMC R740 server. The FPGA accelerator image was modified with several optimizations to the compute kernels within the underlying Intel Deep Learning Accelerator (DLA) stack, including optimizations to the deconvolution, normalization, and pooling layers as well as to the PE array configurations.

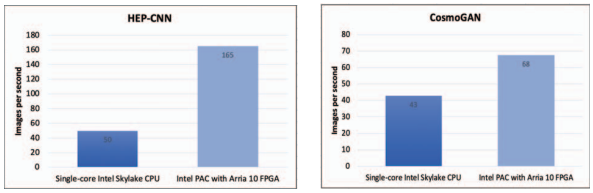


Fig 3. Throughput of HEP-CNN and CosmoGAN.

In the HEP-CNN benchmark, the FPGA achieved a throughput of 165 frames/sec, a speedup of 3.3x against a single core (multi-thread) Skylake processor representing a mission-critical application with space, weight and power (SWaP) constraints. In the CosmoGAN benchmark, the FPGA achieved 68 frames/sec throughput, a speedup of 1.6x compared to the CPU. In this particular scenario, the application is bottlenecked by data ingress into the FPGA accelerator. With planned optimizations to the DLA stack and enhancements to its data organization architecture, we anticipate improved performance of both CosmoGAN and HEP-CNN benchmarks.

## IV. Conclusions and Future work

We presented the HGC framework for managing scientific workflows in SCAIGATE science gateway. Specifically, using the HGC framework, we demonstrated the inference performance of two scientifically important applications to spur interests in the community in the usage of science gateways enabled with FPGAs and other accelerators. Future work will incorporate studies from more scientific applications (beyond HEP-CNN and CosmoGAN), explore more complex DNN models such as 3D-GANs, and seek to further develop the SCAIGATE infrastructure to support access by the scientific communities.

## References

[1] V. Kumar and M. L. Garg, "Deep learning in predictive analytics: A survey," 2017 International Conference on Emerging Trends in Computing and Communication Technologies (ICETCCT), Dehradun, 2017.

[2] Intel Agilex. https://newsroom.intel.com/news/intel-driving-data-centric-world-new-10nm-intel-agilex-fpga-family/#gs.flvnht

[3] Amazon EC2 F1 Instances for Educators https://aws.amazon.com/education/F1-instances-for-educators/

[4] C. Jiang, D. Ojika, T. Kurth, Mr Prabhat, S. Vallecorsa, B. Patel, H. Lam, Acceleration of Scientific Deep Learning Models on Heterogeneous Computing Platform with Intel FPGAs, International Supercomputing Conference (ISC), Frankfurt, Germany, 2019.

[5] D. Ojika, H. Lam, A. Gordon-Ross, B. Patel, SCAIGATE: Science Gateway for Scientific Computing with Artificial Intelligence and Reconfigurable Architectures, Gateways, Austin, Texas, 2018.

[6] HEP-CNN. https://docs.nersc.gov/analytics/machinelearning/benchmarks/#hep-cnn

[7] M. Mustafa; B. Deborah, B. Wahid, L. Zarija; Al-Rfou, R. Kratochvil, J. M., "CosmoGAN: creating high-fidelity weak lensing convergence maps using Generative Adversarial Networks", Computational Astrophysics and Cosmology, Volume 6, Issue 1, article id. 1,