

Optimized machine learning inference on heterogeneous architectures using SOFIE

Sanjiban Sengupta^{1,2} and Lorenzo Moneta¹

1 CERN, Switzerland

2 The University of Manchester, United Kingdom

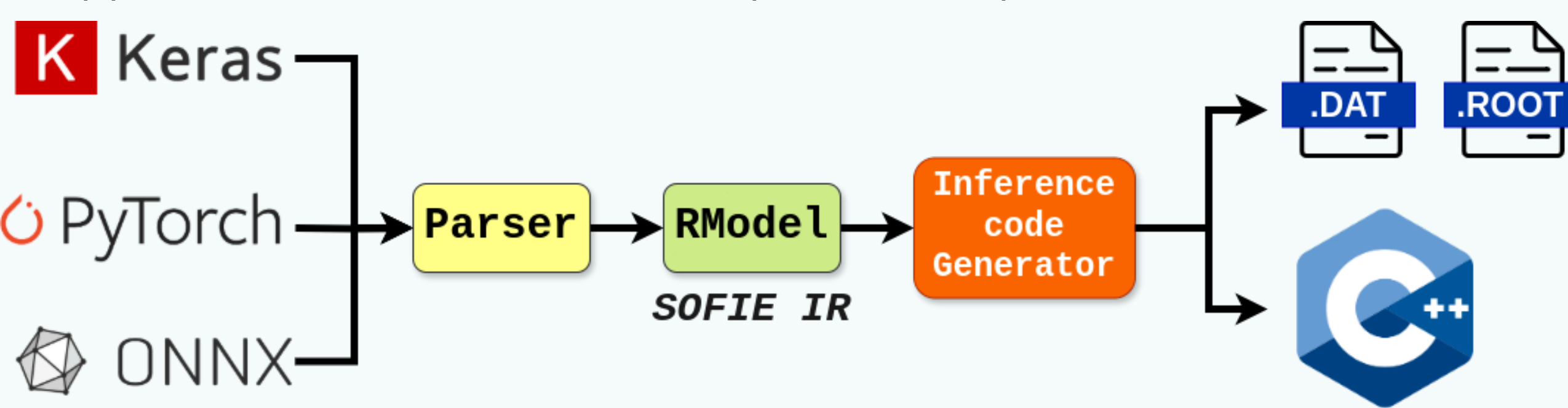


Motivation

- Popular libraries, such as Keras and PyTorch, provide functionality for inference, but **support only their own models**, whereas inference in ONNXRuntime is constrained by **heavy dependency**.
- SOFIE creates standalone C++ inference code, which can be included in any other C++ project with **limited dependencies**.
- In this work, we add optimizations in SOFIE that further reduces its memory usage making it more suitable for deployment in constrained environments. We further present SOFIE's latest developments for generating inference code for heterogeneous architectures.

Background

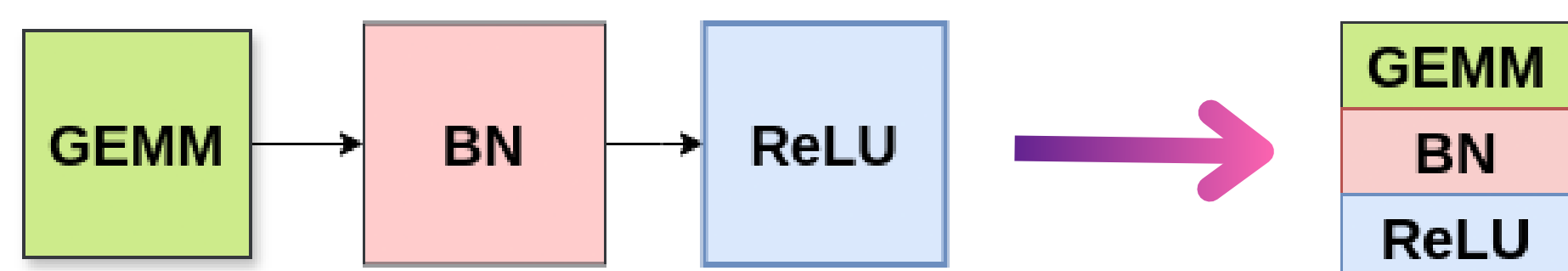
- SOFIE parses a trained model in ONNX, Keras or PyTorch format to its IR (based on ONNX standard).
- Generates inference code in the form of C++ functions.
- Supports several ONNX operators, including Transformer models used by LHC experiments.
- Supports also GNNs trained in DeepMind's Graph Nets.



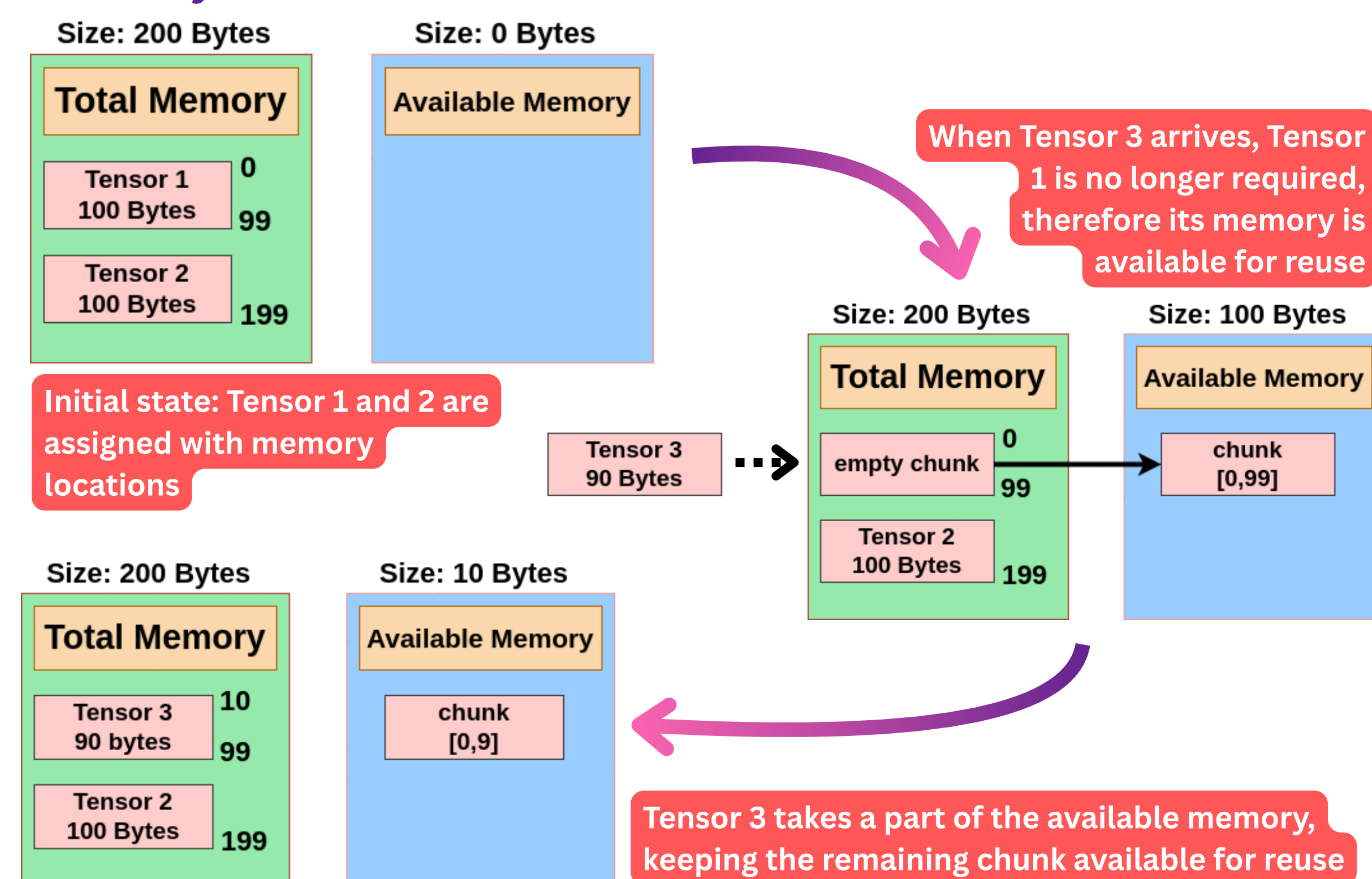
Optimization Methods

- Benchmarking studies^[1] against ONNXRuntime suggested SOFIE performs better for smaller models and single event evaluation in time and memory, but takes longer time and intensive memory for large models.
- Several enhancements can be made to further optimize SOFIE for larger models, which include:

Multi-Operator Fusion

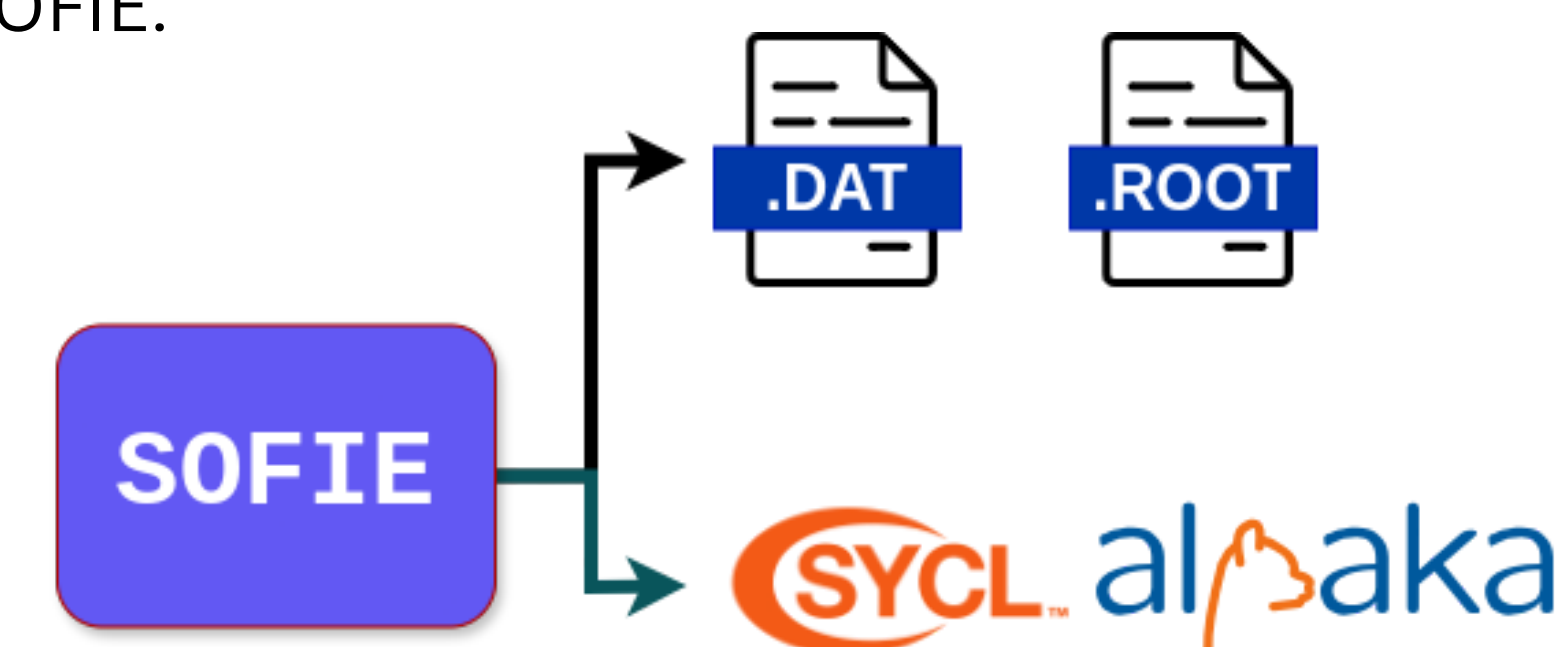


Memory Reuse



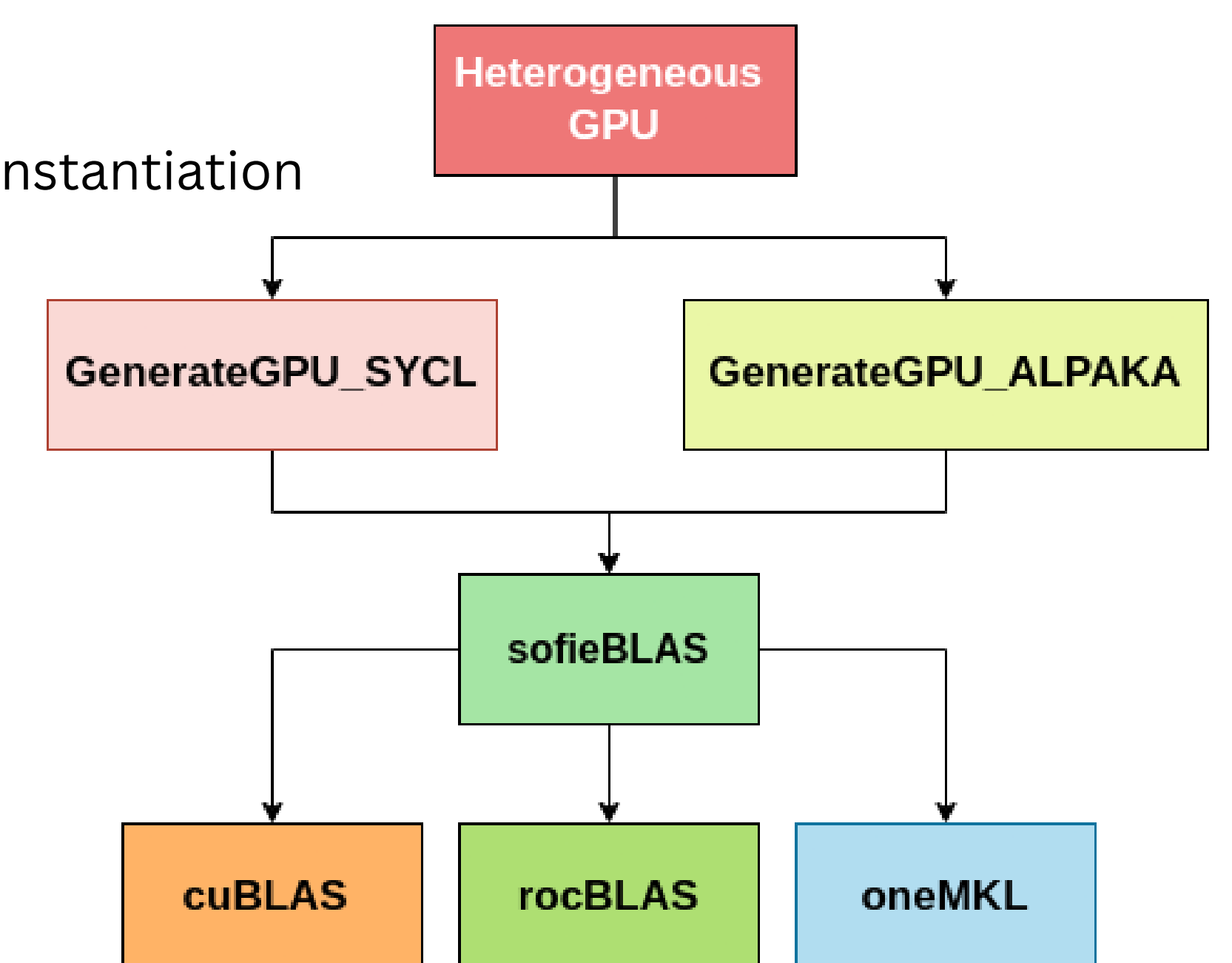
Portability

- Support for inference on heterogeneous architectures provides flexibility to the user for easier integration in complex pipelines.
- Abstraction libraries like SYCL and ALPAKA^[2] provides a platform agnostic interface for managing memory on heterogeneous architectures, making them suitable to be supported by SOFIE.
- Development continues with prototypes implemented for support of SYCL^[3] and ALPAKA in SOFIE.



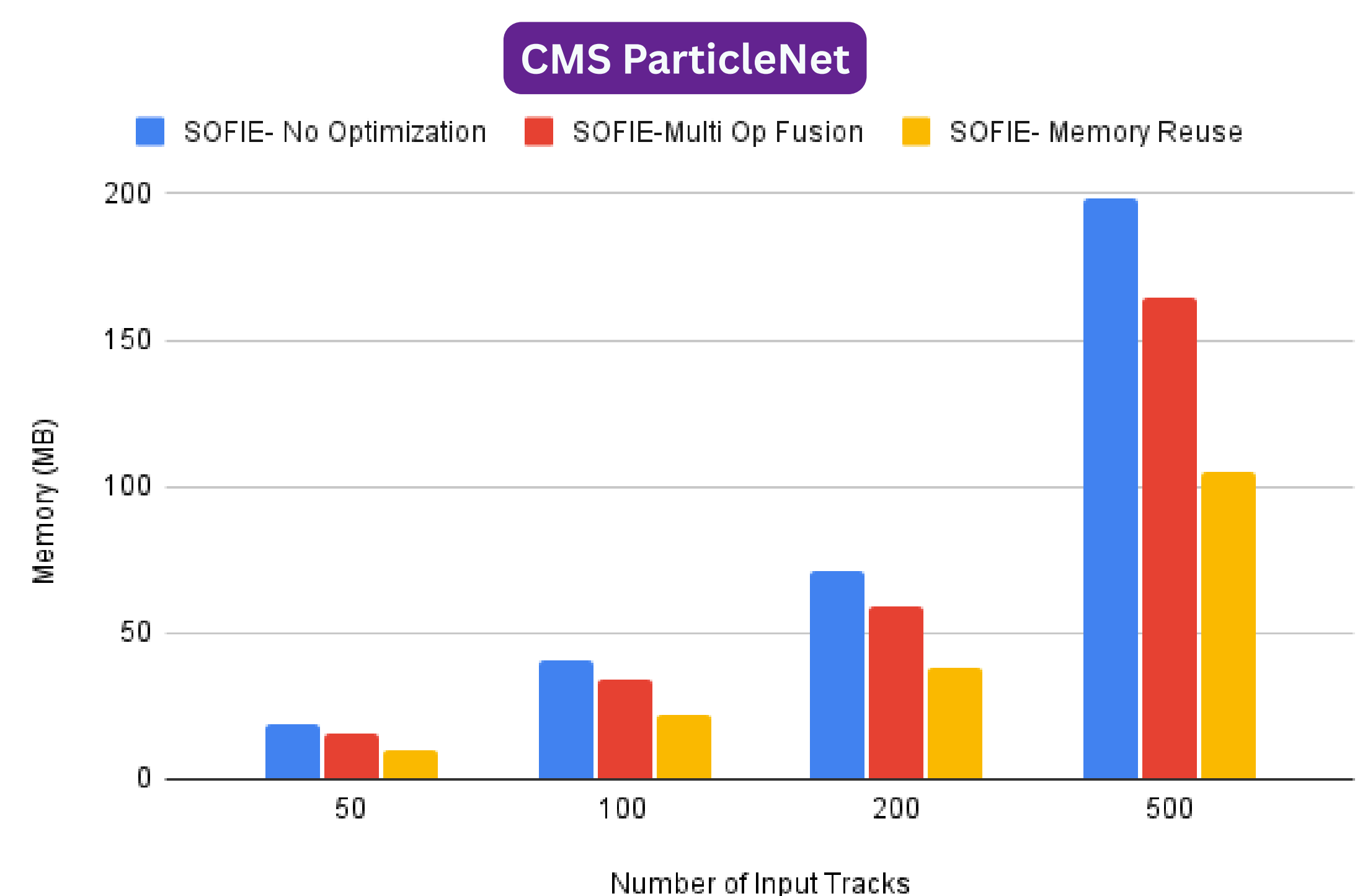
Architecture

- Using buffer-accessor model
- Initializes buffers during session instantiation
- Accepts buffers as inputs
- Returns buffers as outputs
- Abstract Infer function
 - user has the control of running on Intel, NVIDIA, or AMD GPUs
 - BLAS methods chosen automatically as per the chosen execution architecture.

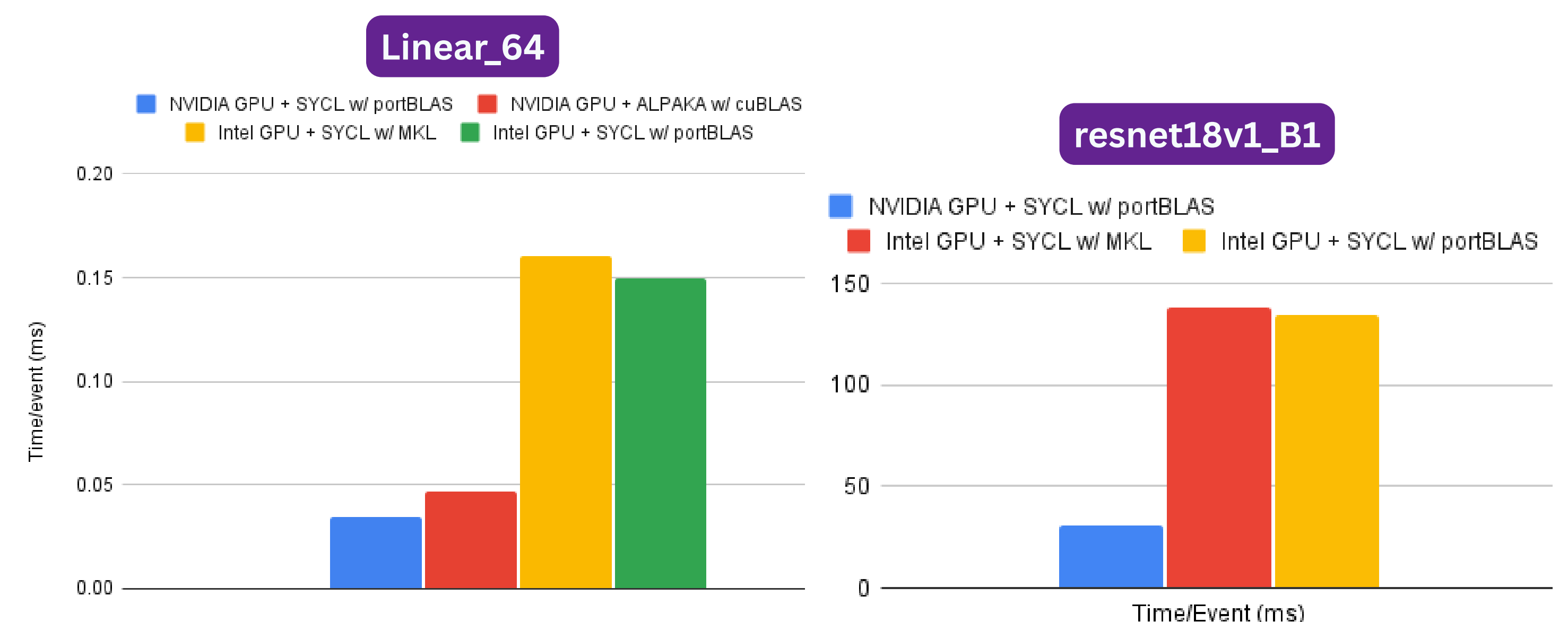


Experimental Evaluation

- Experimentation on inference code for ParticleNet was performed after adding the optimization methods, measuring particularly memory improvements.



- Experimentation on generated heterogeneous code was performed for various configurations and platforms, measuring particularly elapsed time per event.



Key references

- [1] Moneta L., Panagou I.M., Sengupta S. 2024. Benchmark Studies of ML Inference with TMVA SOFIE. In Proceedings of the 27th International Conference on Computing in High-Energy & Nuclear Physics. Krakow, Poland.
- [2] Matthes, A., Widera, R., Zenker, E., Worpitz, B., Huebl, A., & Bussmann, M. 2017. *Tuning and optimization for a variety of many-core architectures without changing a single line of implementation code using the Alpa library*. Retrieved from <http://arxiv.org/abs/1706.10086>
- [3] Panagou I.M., Bellas N., Moneta L., Sengupta S. 2024. Accelerating Machine Learning Inference on GPUs with SYCL. In Proceedings of the 12th International Workshop on OpenCL and SYCL. Association for Computing Machinery, New York, NY, USA, Article 17, 1-2. <https://dl.acm.org/doi/abs/10.1145/3648115.3648123>



Acknowledgements



Supported by the Eric & Wendy Schmidt Fund for Strategic Innovation (grant agreement SIF-2023-004)

More Information



SANJIBAN SENGUPTA
Doctoral Student
(CERN-EP/SFT, U of Manchester)
sanjiban.sengupta@cern.ch