
ATLAS
EXPERIMENT


EVROPSKÁ UNIE
Evropské strukturální a investiční fondy
Operační program Výzkum, vývoj a vzdělávání

Libraries for ATLAS fast simulation

Investigation in progress

Henry Day-Hall

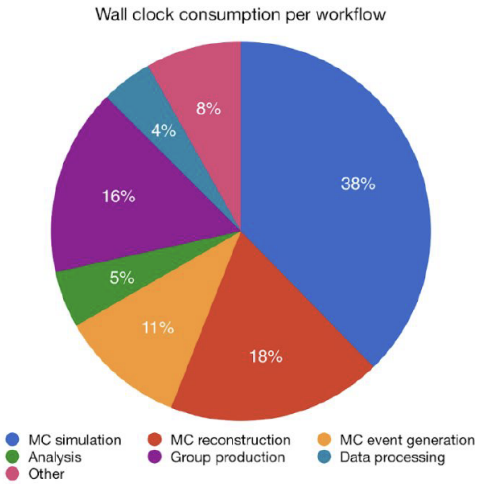
What is ATLAS fast simulation?

What are the NN libraries?

Structure of the upgrade

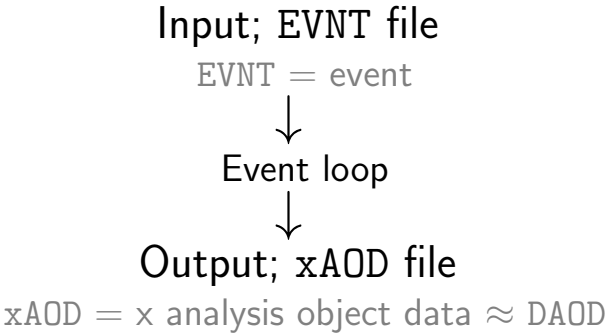
Conclusions, and feedback request.

What is ATLAS fast simulation?



ATLAS fast simulation (AtlFast3) is a composite simulation of the detector calorimeter with muon punch through.
It is designed to alleviate the computational cost of simulation, while maintaining a good emulation of the physics in the full simulation.

Data formats and frameworks



Data formats and frameworks

Input; EVNT file

Contains collision data pre detector simulation. The data is stored within a root file, using streamers to persist it as C++ objects.

↓

Event loop

↓

Output; xAOD file

xAOD = x analysis object data ≈ DAOD

Data formats and frameworks

Input; `EVNT` file

`EVNT` = event

Event loop

Applies operations to each event. Designed to reduce boilerplate code for looping on events in multiple CPU threads. Cannot be used for vectorised computation across events.

Output; `xAOD` file

`xAOD` = x analysis object data $\approx$ `DAOD`

Data formats and frameworks

Input; EVNT file

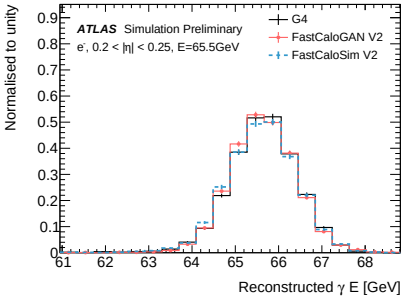
EVNT = event

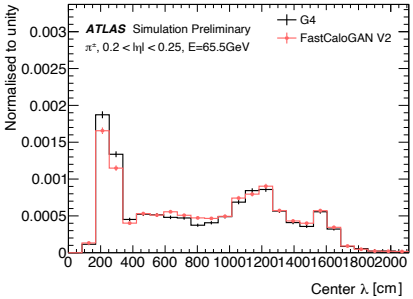
Event loop

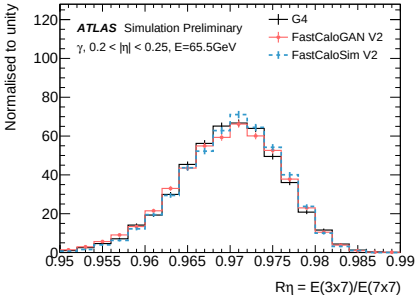
Output; xAOD file

Contains the detectors response to the events + the reconstructed particles. Like EVNT files, these are stored as C++ objects using streamers. Standard input format for analysis.

Comparison to the full simulation







Preliminary results achieve good accuracy in a range of variables.

What changes are we considering?

There are 3 NNs in AtlFast3;

- The voxel GAN; the calorimeter is partitioned into discreet areas called voxels. The distribution of a particles energy between these voxels is predicted from its kinematics.
- The lateral weight NN; the depth of the actual hit within a layer is random, but not uniform. NN defined the distribution of the lateral shape of the shower within the voxel.
- Muon punch through NN; sometimes decay products escape the calorimeter into the muon system. The frequency and kinematics of this is also simulated with an NN.

Currently they all use the library LWTNN for interpolation, we are investigating a move to ONNX.

Current situation

- AtlFast3 runs in roughly 1/5 of the time required for the full simulation.
- They have very similar memory requirements.
- We would like to reduce both these requirements, but in particular, we think memory could be reduced.

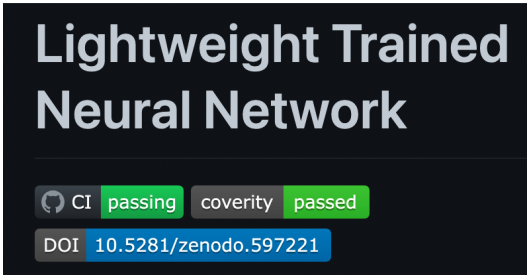
How can we change this?

Jobs are run on the In run 3, everything ran on CPU, but there are GPUs available, and we should be using them.
ATLAS has access to a range of GPUs for jobs, total 10.

- Mostly Nvidia.
- Models include V100, A100, V100S, T4.
- Varying CPUs associated.

LWTNN

- Originally written for LHCb.
- With CI and some documentation, 15 contributors.



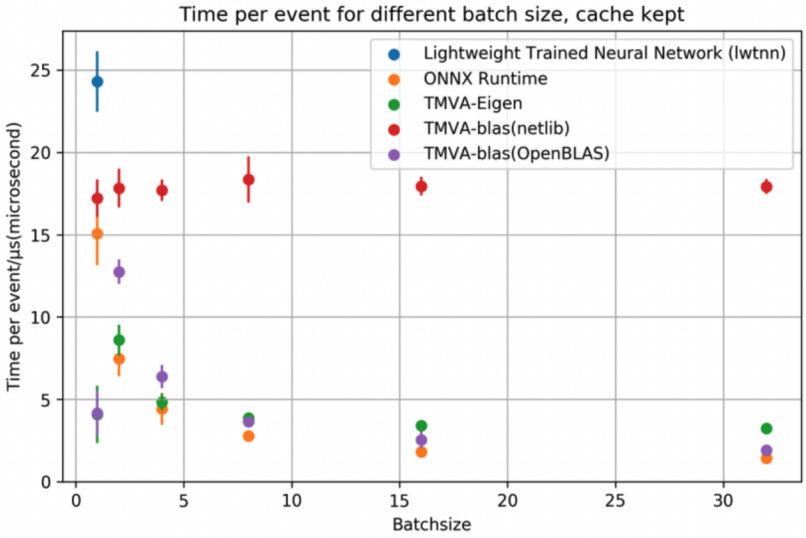
- Pure C++ interpolation, dependencies on boost's property tree and Eigen.
- Eigen is a matrix algebra library, CUDA compatible.

Open Neural Network Exchange

- Standard for specifying the geometry and parameters of a network.
- Subset of the proto format (from Google), which can be saved as a binary.
- Actually doing inference requires another library.

- ONNX runtime is developed by Microsoft.
- CI in 4 operating systems (3 with GPU), 434 contributors.
- Only dependencies are CUDA libraries for GPU.

Benchmarking



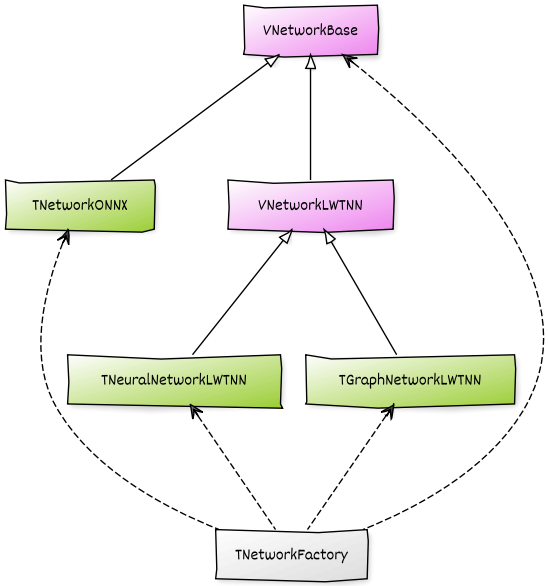
Given the great disparity in resources used to produce each library, it's impressive how well LWTNN holds up. None the less ONNX runtime does improve on it.

Figure: An, Sitong and Moneta, Lorenzo. *doi: 10.1051/epjconf/202125103040.*

A wrapper for NN libraries

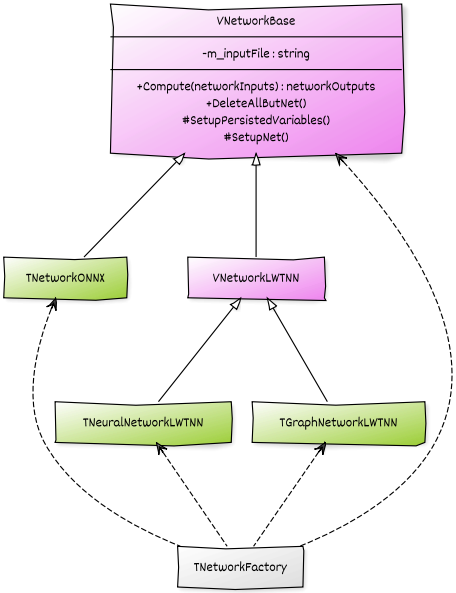
- Verify that changes do improve computational efficiency.
- Guarantee changes don't impact physics output.
- Facilitate saving inside a root file (Streamers).

A wrapper for NN libraries



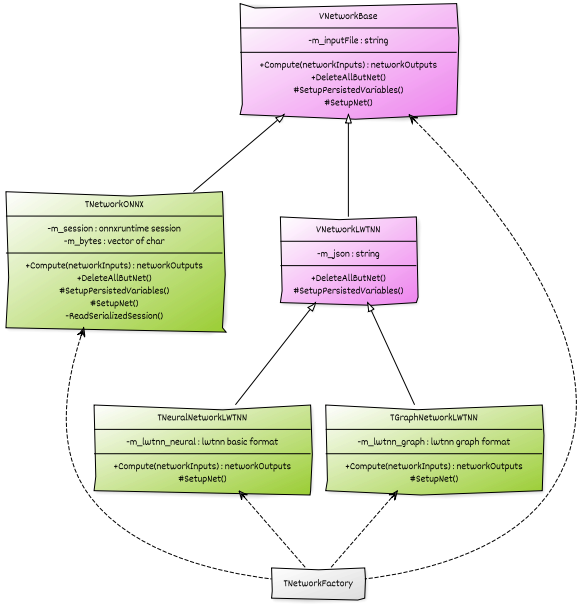
The factory method pattern will meets all requirements neatly.

A wrapper for NN libraries



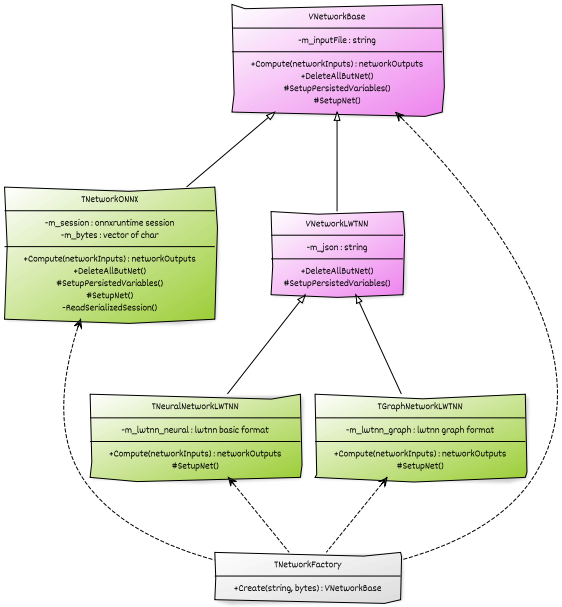
The abstract `VNetworkBase` offers a common interface for all network interactions.

A wrapper for NN libraries



`TNetworkONNX`, `TNeuralNetworkLWTNN` and `TGraphNetworkLWTNN` inherit from `VNetworkLWTNN` and handle the specifics of each library.

A wrapper for NN libraries



`TNetworkFactory` hosts a static function that uses whatever data is given (file path, json string, vector of bytes) to form the corresponding net. It return a unique pointer to `VNetworkBase`.

Conclusions

- Strong motive to reduce compute time of AtlFast3, and due to extensive use of NNs, changing the library doing the interpolation could be a major advantage.
- Current objective is to test ONNX runtime, but also create flexibility.
- Feedback request; are there other libraries we should consider?
- Feedback request; are there other low-hanging fruit (improvements) we should look at?


ATLAS
EXPERIMENT