4th Inter-experiment Machine Learning Workshop

Argonne NATIONAL LABORATORY

Pre-Learning a Geometry Using Machine Learning To Accelerate High Energy Physics Detector Simulations

Vangelis Kourlitis, Walter Hopkins, Doug Benjamin

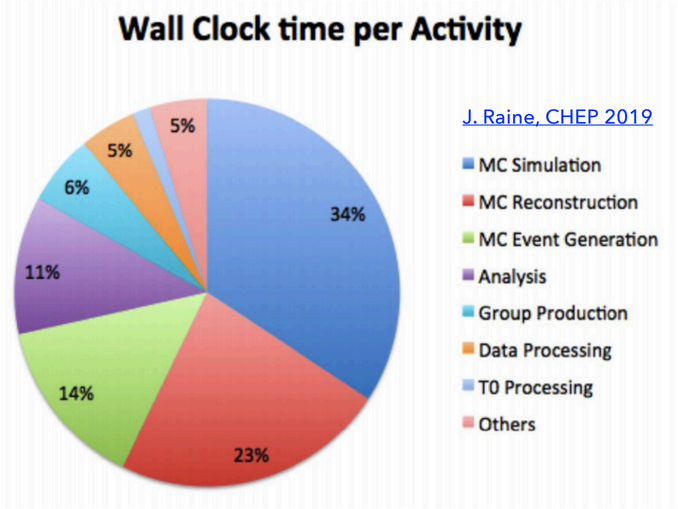
Oct 22nd 2020

HEP Detector Simulations

The ATLAS detector paradigm

Facts

1. Detector simulation the largest CPU consumer.
2. EM calorimeters dominate the simulation load (steps).

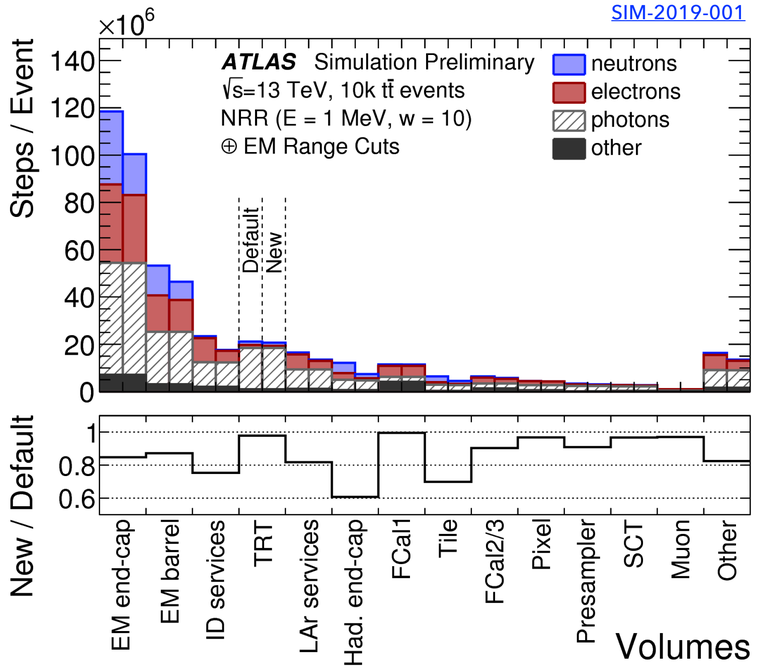


Geant4 Profile: photons on end-cap

Callees	CPU Time: Total
G4SteppingManager::Stepping	100.0%
G4SteppingManager::DefinePhysicalStepLength	66.7%
G4VProcess::AlongStepGPIL	58.2%
G4Transportation::AlongStepGetPhysicalInteractionLength	51.7%
G4Navigator::ComputeStep	34.1%
G4NormalNavigation::ComputeStep	20.9%
G4VoxelNavigation::ComputeStep	10.8%

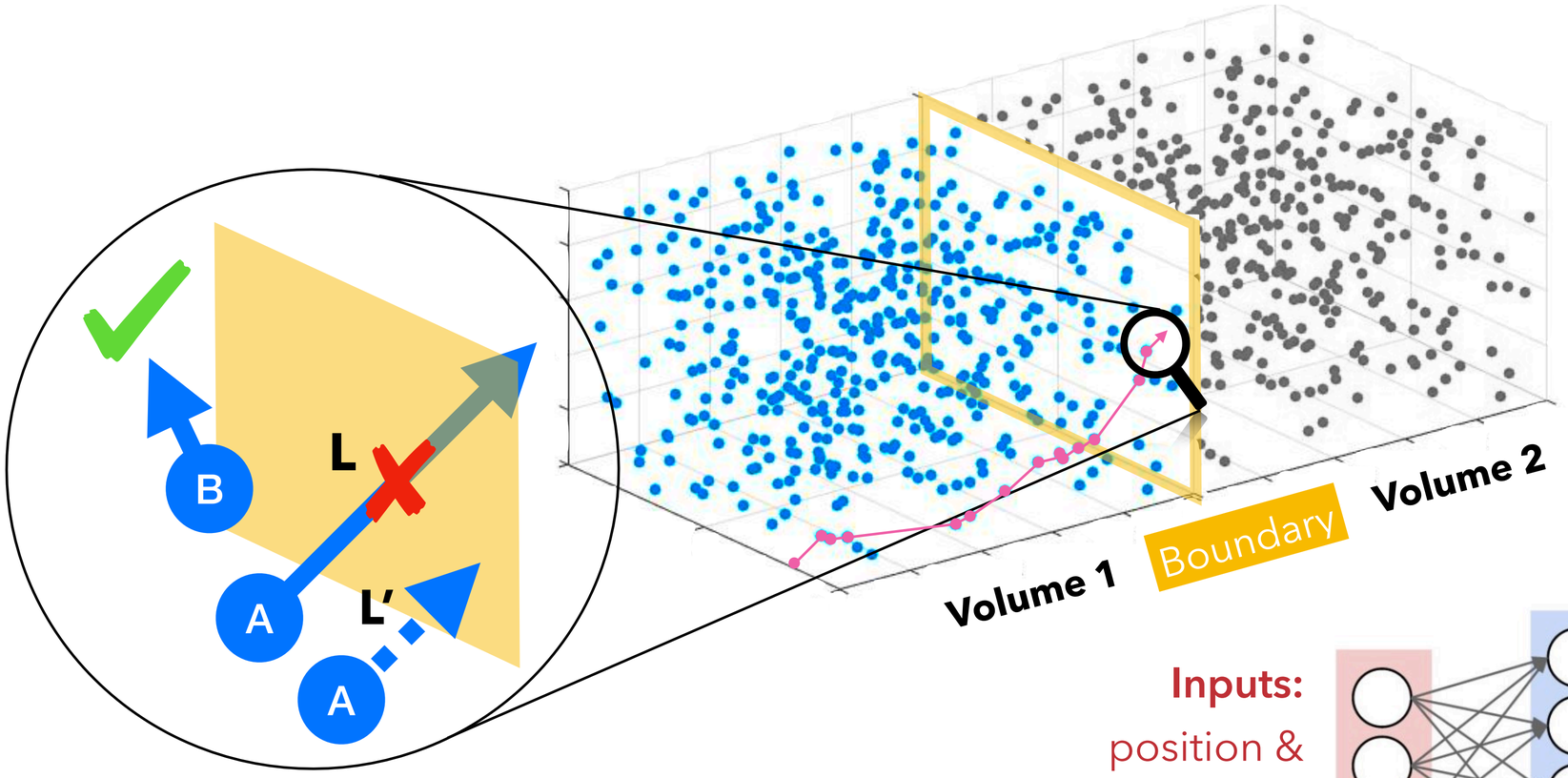
The point: methods exploring the geometry* are taking significant amount of the simulation time.

* Locate position into geometry tree and calculate distance to next boundary in order to limit step.



The Idea

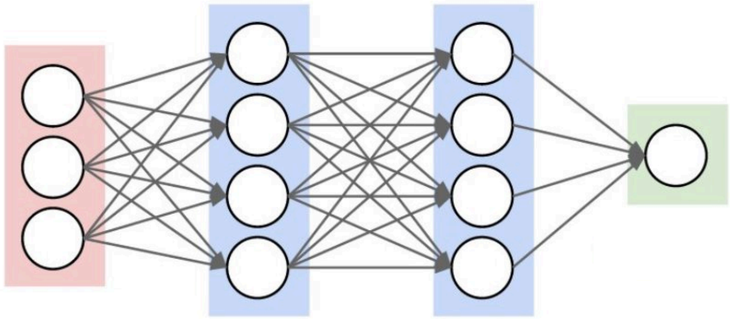
Surrogate modeling within Geant4: *Could we speed-up the geometry exploration by using a* pre-defined/learned map instead of algorithmic calculations *in each step?*



Machine learning regression technique trained for a particular geometry (e.g. ATLAS EMEC)

Hardware transparent industrial libraries optimized for different hardware architectures (CPU or GPU)

Inputs: position & direction

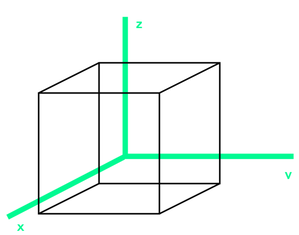


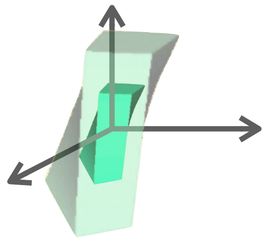
Output: Geometrically safe step length

Fully connected NNs, Point-nets, …

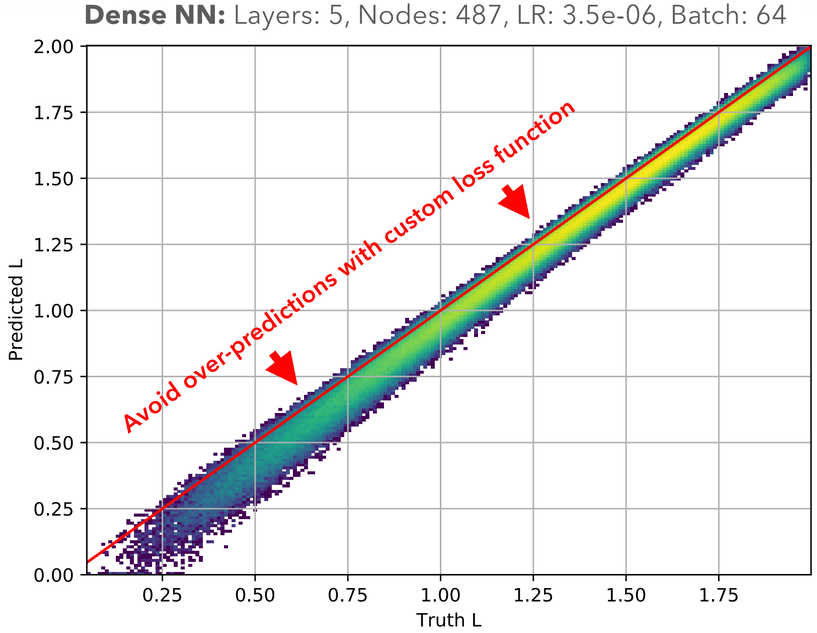
Results

Explore the idea in simplified geometries:





Training data collection via Geant4 application



How long Geant4 takes to explore the geometry vs ML inference?*

	Geometry	Time [μs]
Geant4	ATLAS EMEC	~ 5
	(nested) Twisted Trapezoids	50 - 100
Dense NN evaluation	-	~ 1000

Strong Remarks!

- No batch evaluation
- Evaluation on CPU
- Naive model structure
- Tests in Python

* Locate position into geometry tree and calculate distance to next boundary in order to limit step.