VECTORFLOW

Arturo Garza Rodriguez

Mentors:
Andrei Gheata
Guilherme Amadio
INTEGRATE SIMD COMPONENTS IN SCALAR FLOWS

Many CPU-intensive high-energy physics algorithms may profit from the vector pipelines of modern processors, they don’t because they don’t have vectorizable inner loops.

Can we do better?

• Cannot rely on compilers to provide an optimal solution.
• No recipe for SIMD vectorization, but we have libraries like VecCore.
• How to integrate them with other data flows?
VECTORFLOW DESIGN: non-intrusive & smooth integration

Pipeline flow:

Complex flow:

State is “propagated” through the graph according to runtime decisions
COMPLEX FLOW: Generate + Propagate

Each layer is translated to a VectorFlow task that propagates a vector of tracks (previously gathered in the correct format for SIMD processing).

Each particle track can be propagated to an inner or outer layer in the tube, i.e. a task dispatches data to the other tasks' containers.

If tracks are propagated outside the geometry, they are scattered back to the original data flow.

The flow continues its execution until as long as there is still data in the buffers.

Up to ~2.0x speed-up in AVX2 processor, even though inherent overhead due to data transformations.