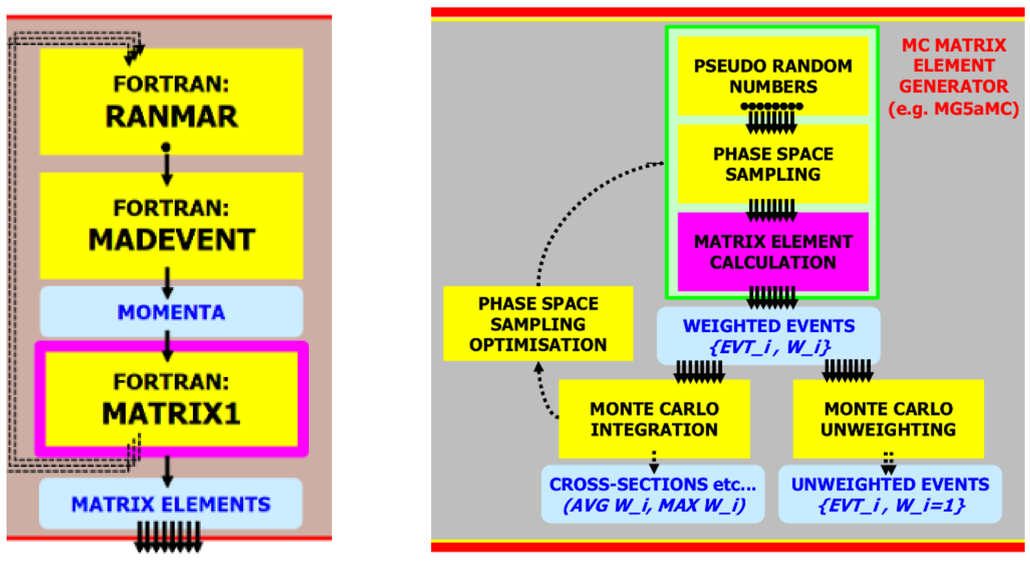
Porting MADGRAPH to FPGA using High-Level Synthesis (HLS)

Héctor Gutiérrez, Luca Fiorini, Alberto Valero, Francisco Hervas
IFIC (Universitat de València - CSIC)
HIGH-LOW TED2021-130852B-I00

MADGRAPH CPU

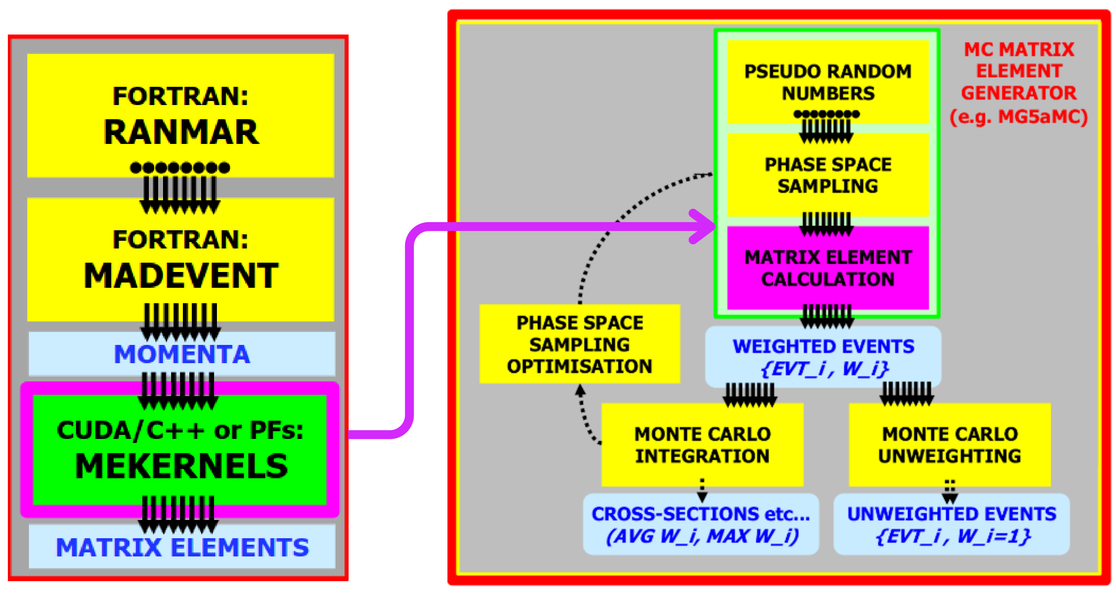
MadGraph5_aMC@NLO is a framework that aims at providing all the elements necessary for SM and BSM phenomenology, such as the computations of cross sections, the generation of hard events and their matching with event generators. Processes can be simulated to LO accuracy for any user-defined Lagrangian, and the NLO accuracy in the case of QCD (Quantum Chromo Dynamics) corrections to SM processes. Matrix elements at the tree- and one-loop-level can also be obtained.



MADGRAPH's functioning

MADGRAPH GPU

The acceleration of MADGRAPH on GPU focuses on three key components: pseudo-random number generation, phase space sampling, and matrix element calculation. These tasks, originally executed on the CPU, have been ported to the GPU, leveraging its massive parallel processing capabilities. This enables the simultaneous handling of thousands of events, drastically reducing execution times and improving efficiency in large-scale event simulations.

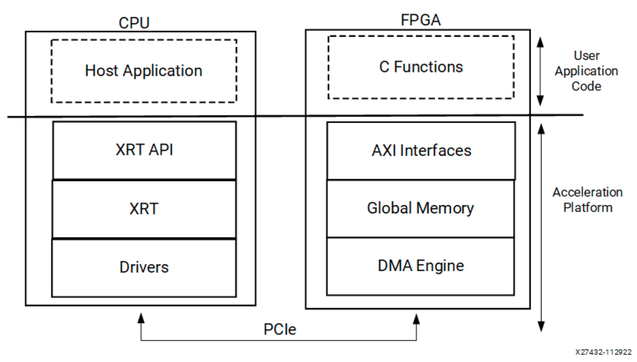


MADGRAPH's GPU port

Credits to: Madgraph5_aMC@NLO for GPUs group

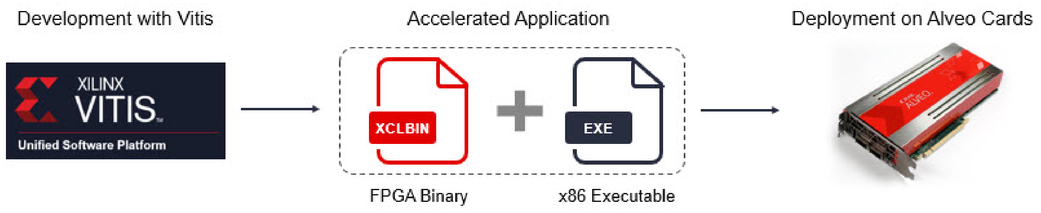
MADGRAPH FPGA

Developing applications for Alveo involves two parts: programming the host, which runs on x86 processors, and programming the FPGA, which accelerates specific functions. Host development is similar to regular software development, using C/C++ and the OpenCL API to manage tasks on the FPGA, transfer data, and program the FPGA in real-time, optimizing its resources.

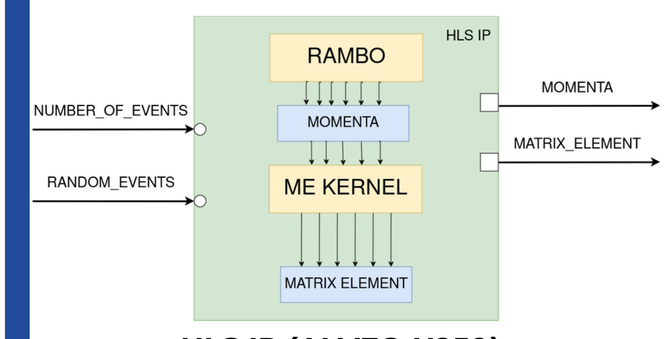


CPU/FPGA Interaction

FPGA application development is more complex, typically using low-level languages like Verilog or VHDL. However, the Vitis environment allows using C/C++/OpenCL C to design functions, called kernels, which are automatically transformed into RTL using High-Level Synthesis (HLS). Once the RTL is generated, Vitis handles the synthesis, mapping, and creation of the bitstream, packaged in an xclbin file, to program the FPGA.



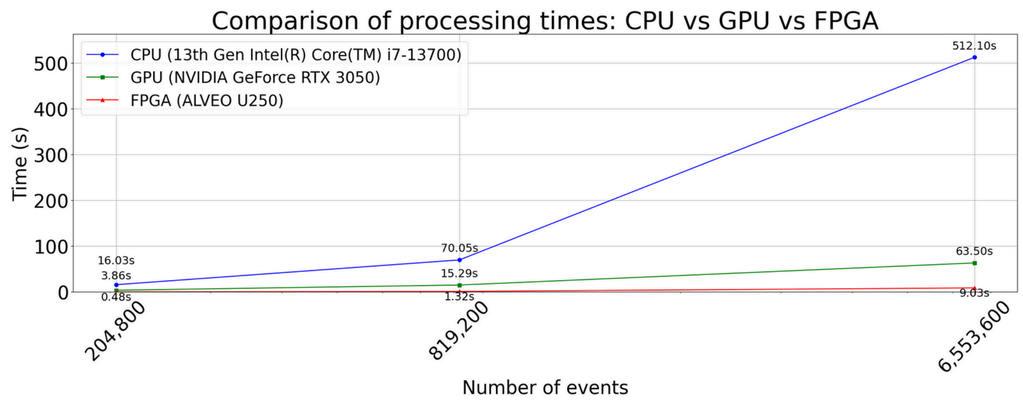
Vitis Development Flow



HLS IP (ALVEO U250)

The CPU sends the "NUMBER_OF_EVENTS" and "RANDOM_EVENTS" data to the FPGA, where the RAMBO block calculates the "MOMENTA" based on these events. Then, the Matrix Element kernel computes the "MATRIX_ELEMENT." Both results are sent back to the CPU for further processing. This setup accelerates complex operations on the FPGA using HLS IP, reducing the computational load on the CPU.

e+ e− > μ+ μ−



CPU vs GPU vs FPGA

This graph compares the processing times for running the MADGRAPH process e+e−→μ+μ− on three different platforms: CPU, GPU, and FPGA. These results highlight the efficiency of FPGA for accelerating this process, outperforming both CPU and GPU, particularly for large-scale computations.

REFERENCES

[1] Xilinx. (n.d.). Vitis development flow. Xilinx. https://xilinx.github.io/graphanalytics/vitis-dev-flow.html

[2] MadGraph. (n.d.). MadGraph: A program for event generation in high-energy physics. Retrieved September 19, 2024, from http://madgraph.phys.ucl.ac.be/