Imperial College London

Demonstration of FPGA Acceleration of Monte Carlo Simulation

Marco Barbone
m.barbone19@imperial.ac.uk

Imperial College London

About me

- M.Sc. in Computer Science and Networking Scuola superiore Sant'Anna & Università di Pisa
- CERN OpenLab Summer student
- Research Software Engineer, Maxeler Technologies
- Ph.D. Student, Custom Computing research group, Imperial College London, Supervisor Prof. Wayne Luk
- Member of Imperial College CMS group
- Associate Ph.D. Student, The Institute of Cancer Research, London

Research Groups

The institute of cancer research, Division of Radiotherapy and Imaging:

- James Bedford, Uwe Oelfke

Imperial College London, HEP & Custom Computing research groups:

- Marco Barbone, Alexander Howard, Kenneth Long, Wayne Luk, Alexander Tapper

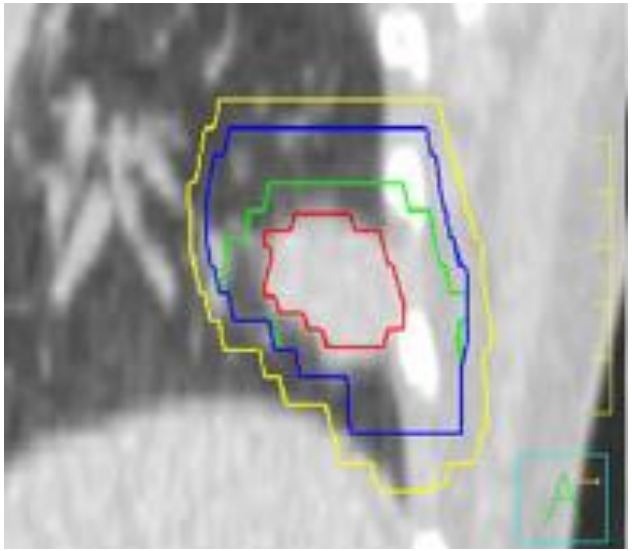
CERN:

- Mihaly Novak

NHS:

- (upcoming)

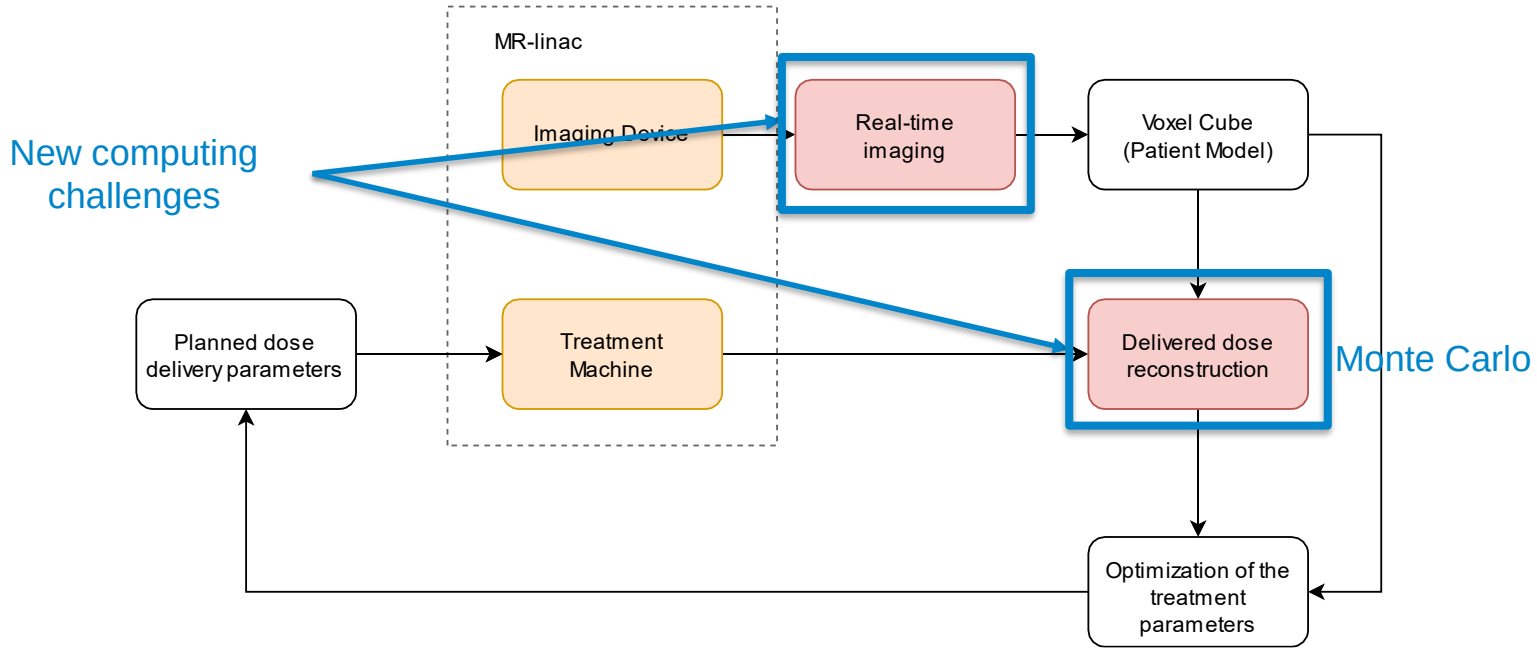
Real-time adaptive radiotherapy is a cancer treatment that takes variations in patient anatomy and cancer movements into account and adjusts the beam to better target the cancer



Gross tumour volume
Clinical target volume
Internal target volume
Planning target volume

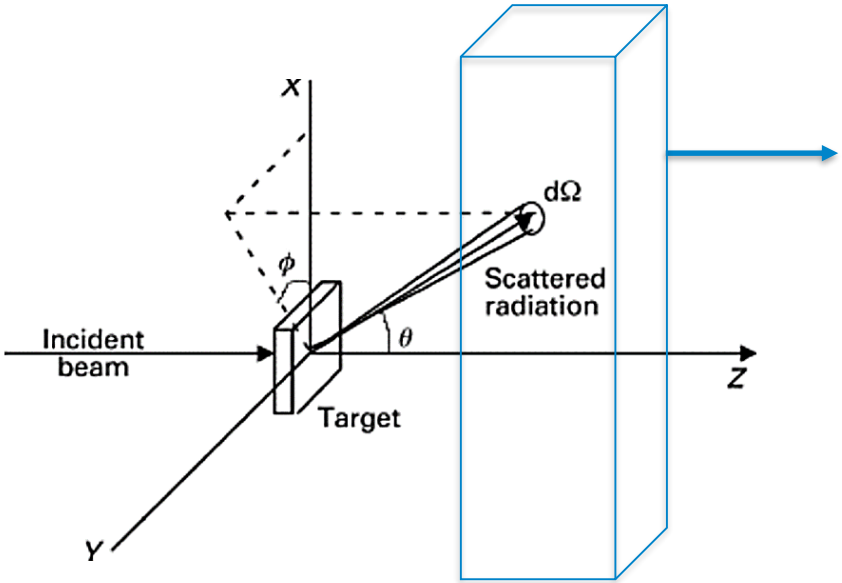
An area wider than the tumor is irradiated

Imperial College London



Imperial College London

Monte Carlo Simulation



Scattered radiation final position

Many particles are simulated to achieve statistical significance

Source: Kotlarchyk M. Scattering theory. Encyclopedia of Spectroscopy and Spectrometry, 1999.

Imperial College London

FPGA acceleration

Imperial College London

Methodology

No agile development! Compilation might take days

A methodology is needed to minimize unneeded compilations:

1. Application analysis
2. Performance and resource modelling
3. Decide the acceleration target
4. Model the target in software
5. FPGA programming

Imperial College London

Workload selection

Not all workloads benefits from FPGA acceleration!

FPGA have higher throughput than CPUs (and GPUs) if the workload:

- Has many branch mispredictions
- Has many cache misses
- (optional) Embarrassingly parallel
- (optional) Independency

→ Monte Carlo fits all!

Imperial College London

FPGA Performance

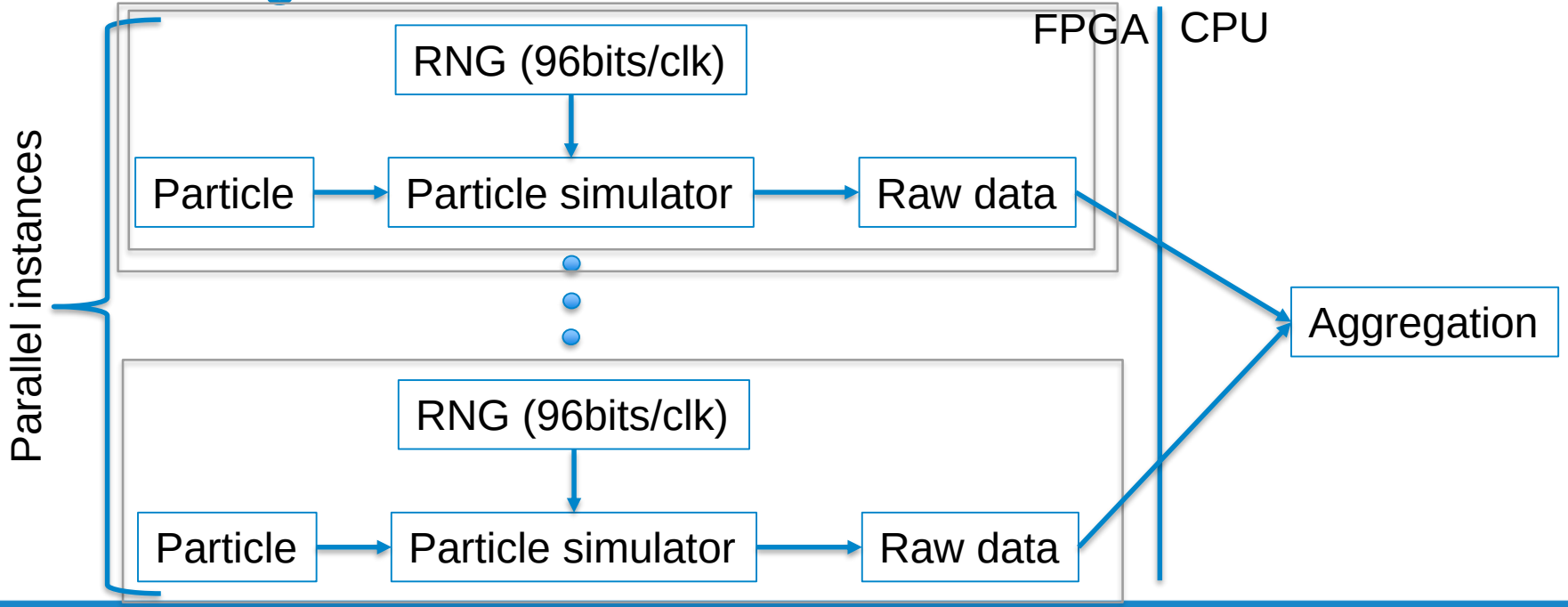
- FPGA Performance is predictable
- There is no context switch, garbage collector or any background process
- The bitstream will be executed the same number of clock cycles every time
- The number of clock cycles needed can be computed easily

Further read: Nils Voss et al. (2021), On Predictable Reconfigurable System Design

https://doi.org/10.1145/3436995

Imperial College London

Resulting architecture



Imperial College London

Test configurations

Hardware:

- AMD Ryzen 5900x 12-cores 3.7GHz (4.8GHz) CPU
- Xilinx's Alveo U200 Data Center accelerator card Xilinx VU9P FPGA

Toolchain:

- MaxCompiler 2021.1
- Vivado 2019.2
- OpenMP

Imperial College London

Test configurations (continued)

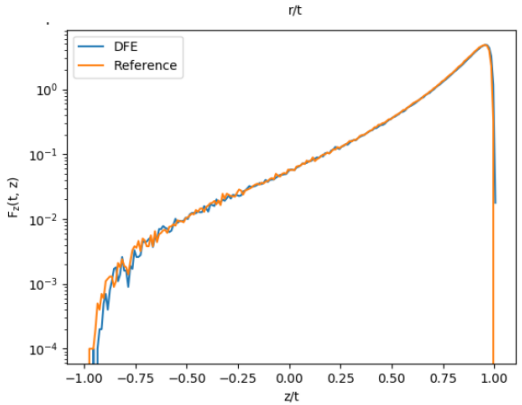
Simulation configuration:

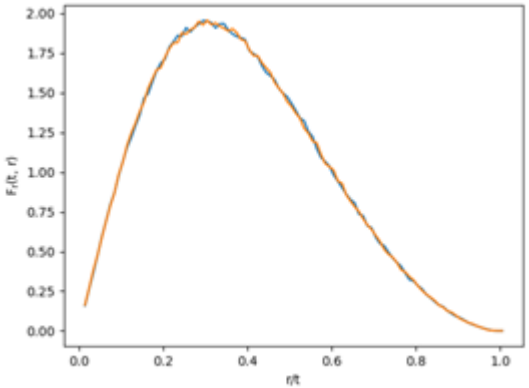
- 100M histories
- 6 MeV beam
- Water

The FPGA clock frequency was limited to 200MHz

Imperial College London

Histograms





It passes a KS-test

Imperial College London

Performance measurements

The FPGA accelerated version is:

- 270x faster than an optimized single-core implementation
- 110x faster than multi-core implementation (12-cores)
- 10x cost-equivalent speedup

The theoretical 128x does not account for overhead and CPU data aggregation

Using MaxCompiler the architecture achieves 300Mhz resulting in a 160x speedup

Imperial College London

Performance measurements

The FPGA accelerated version is:

- 270x faster than an optimized single-core implementation
- 110x faster than multi-core implementation (12-cores)
- 10x cost-equivalent speedup

The theoretical 128x does not account for overhead and CPU data aggregation

Using MaxCompiler the architecture achieves 300Mhz resulting in a 160x speedup

Imperial College London

Possible extensions

Integration in more complex simulation (Geant4)

Accelerating other simulations (EM simulation)

Imperial College London

EM shower simulation

Depending on the degree of parallelism the best accelerator is:

- GPU: the number is in the order of thousands
- FPGA: the number is less then 1k

If the EM shower simulation the parallelism depends on the number of high energy particles

Imperial College London

Conclusions

- FPGAs can be multiple orders of magnitude faster than CPUs and GPUs when accelerating suitable workloads
- Due to their stochastic nature Monte Carlo simulations greatly benefit from FPGA acceleration
- The results shows 110x speedup compared to high-end multicore consumer CPUs
- Following a formal methodology and HLS toolchains high performance can be achieved with no previous FPGA programming experience

Imperial College London

Worst-case for branch predictors

Assuming rng uniformly distributed:

```
if(rng.getRandom() < 0.5) then f(x) else g(y)
```

In the GPU case branches can significantly affect the instruction throughput by causing threads of the same warp to diverge.

The diverging branches are de-scheduled!