Tuning the HIJING++

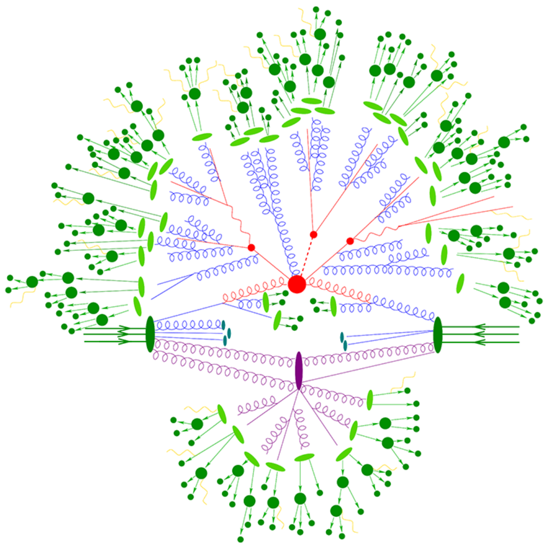
2021.11.10 GPU DAY 2021

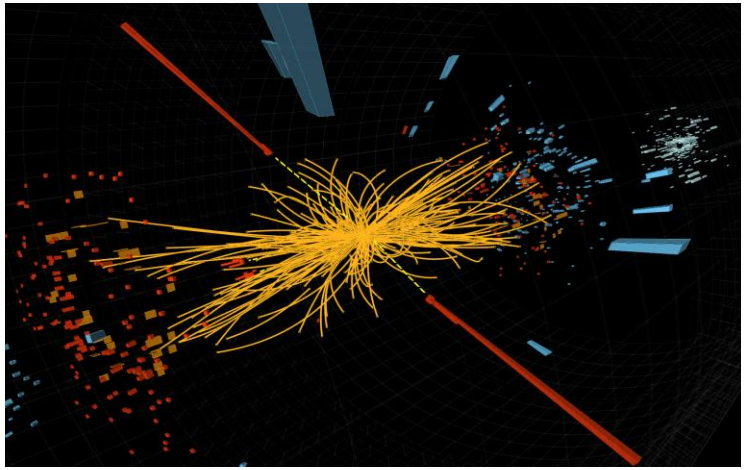
Balázs Majoros, ELTE, Wigner FK
Gábor Bíró, Wigner FK
Gergely Gábor Barnaföldi, Wigner FK
Gábor Papp, ELTE

ELTE
EÖTVÖS LORÁND
UNIVERSITY

Why MCEGs important in HEP?

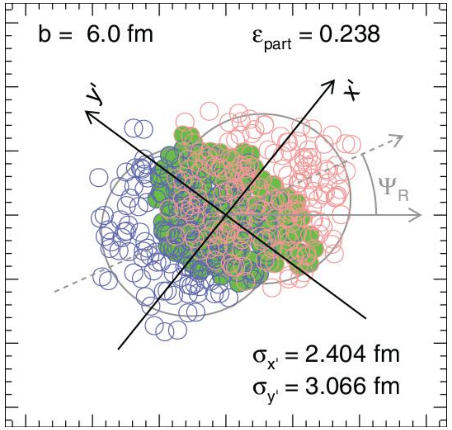
- It provides detailed simulations of high-energy collisions
- Monte Carlo Event Generators are widely used by theorists and the experimental community
- Used in data analysis, designing new experiments
- And to make predictions, and put our models to test
- Computationally **expensive**

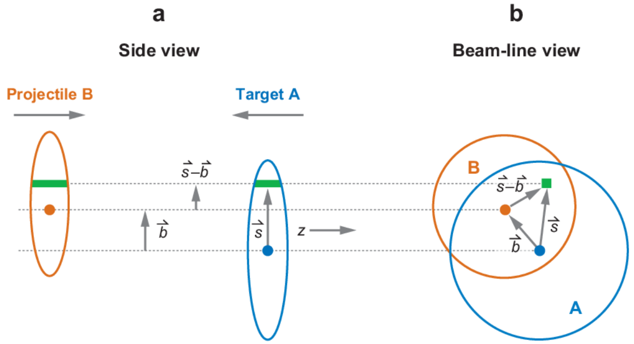




HIJING++

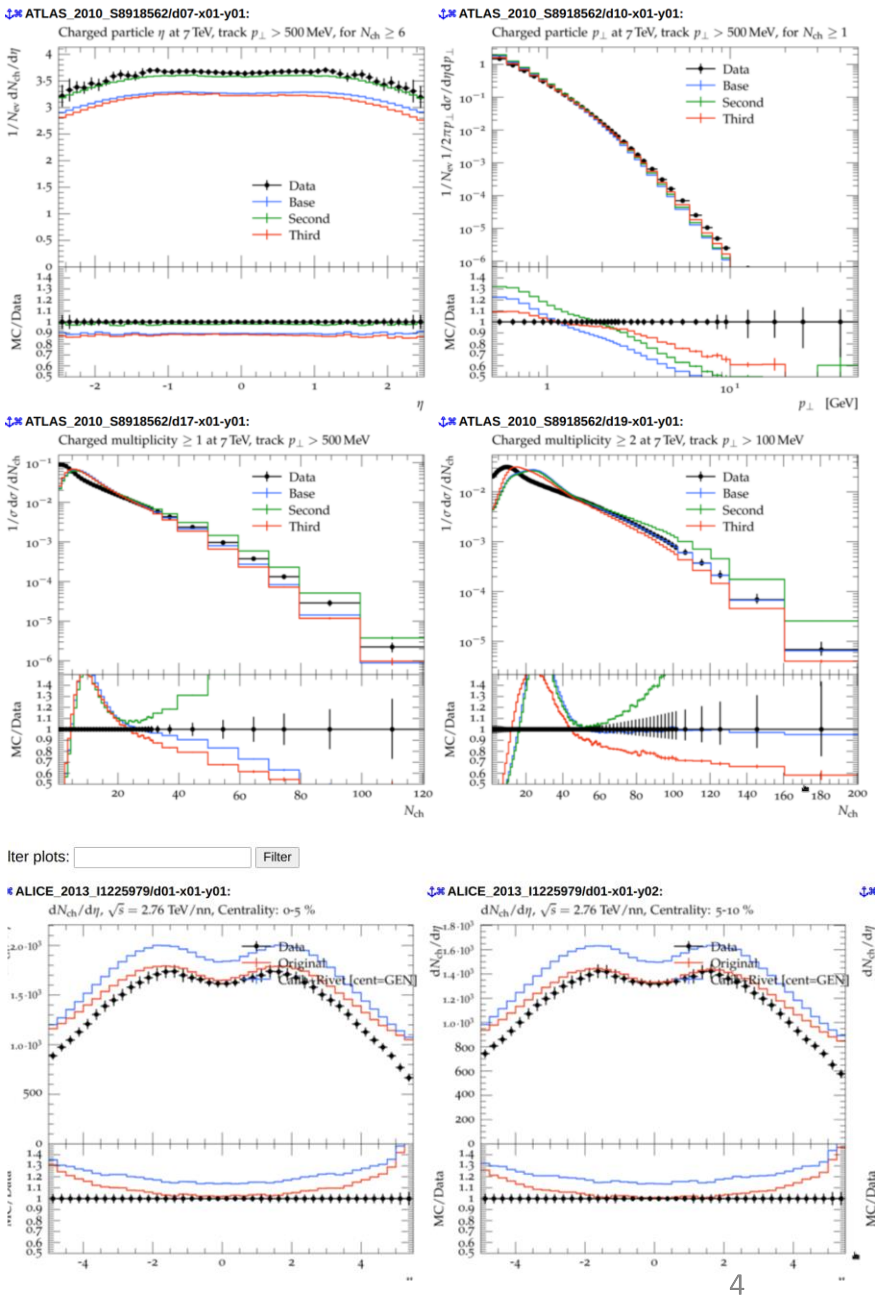
- Monte Carlo Event Generator for proton-proton and heavy-ion collisions (pre-release)
- Successor of the old Fortran HIJING, rewritten in C++
- Support for CPU multithreading
- The output depends on several internal parameters
- We had to examine around **40** physical parameters
- For one system it's a relatively easy task
- But we have to find a working combination for a range of energies from 20 GeV to 13 TeV in p-p, p-A, A-A collisions



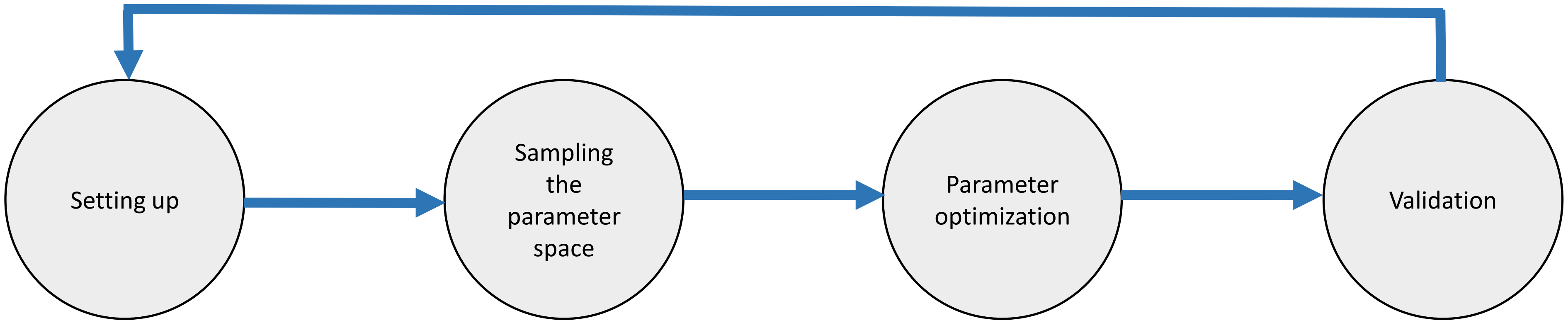


Statistics

- Simulations take place in HPC clusters provided by the Wigner Data Center [1]
- 7 nodes with 16 core Intel Skylake CPU-s and 32 GB RAM
- 5 TB Disk space
- In almost a year (a rough estimate):
  - **22 000** HIJING++ runs
  - Generated ~**340** TB of data
  - In around half a year of CPU time
  - And we need even MORE...

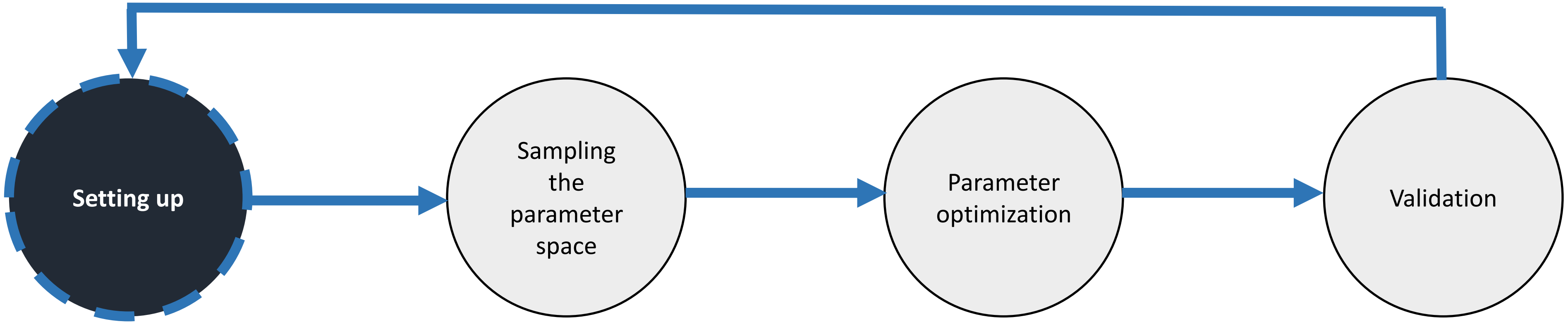


Tools and the tuning process



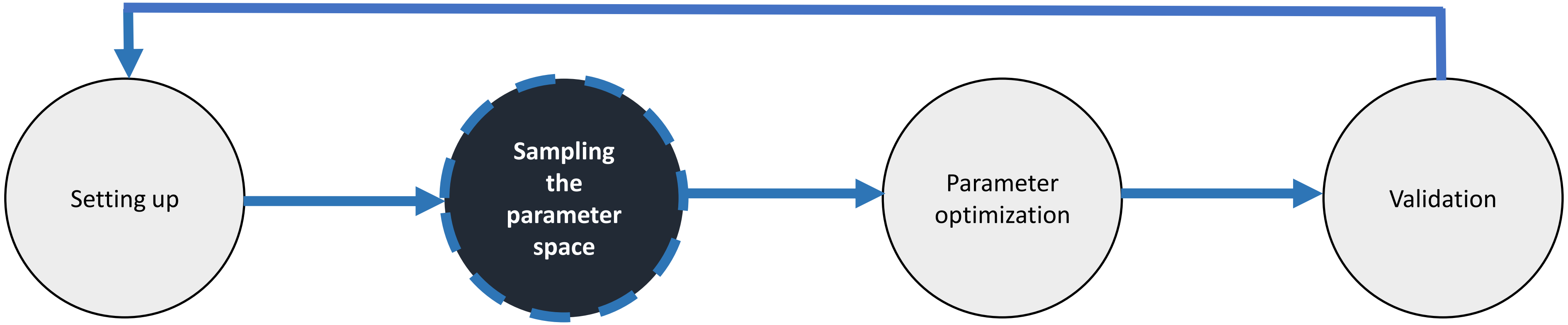
- Stack:
  - Rivet toolkit[2] (**R**obust Independent **V**alidation of **E**xperiment and **T**heory)
  - YODA [3] (**Y**et more **O**bjects for **D**ata **A**nalysis)
  - Professor [4] – parameterisation-based tuning tool, later we compare it to MCNNTUNES [5]
  - HIJING++ (**H**eavy **I**on **J**et **IN**teraction **G**enerator)





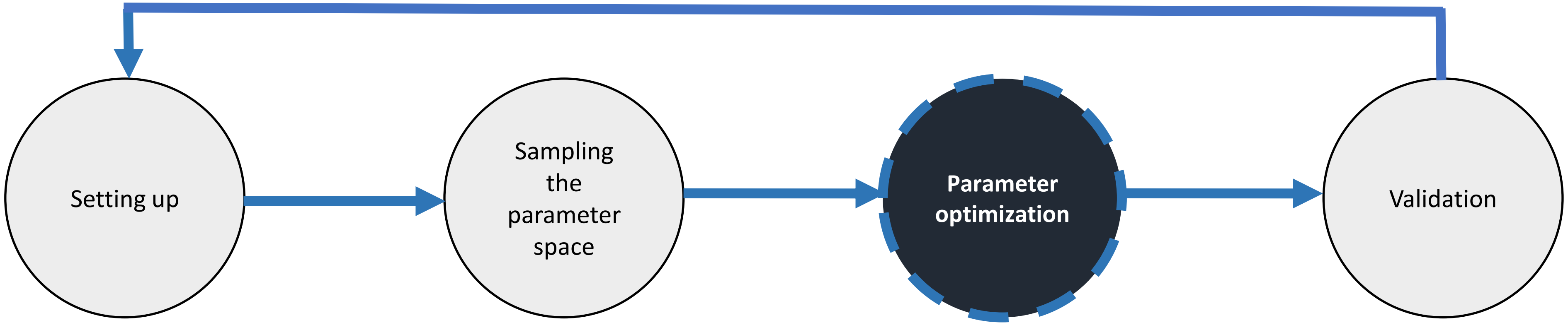
We have to:

1. Choose the energy, the collision system and the experimental reference data
2. Figure out the most relevant parameters -> less parameters = less runs
3. Find a "good" sampling range for the parameters e.g. HardCut [2.0, 3.0]
   This can be quite difficult
4. Write some scripts to run the MC
5. Example: 5.02 TeV, Pb-Pb, ALICE_2016_I1507090
   (Charged particle pseudorapidity density in centrality classes.)

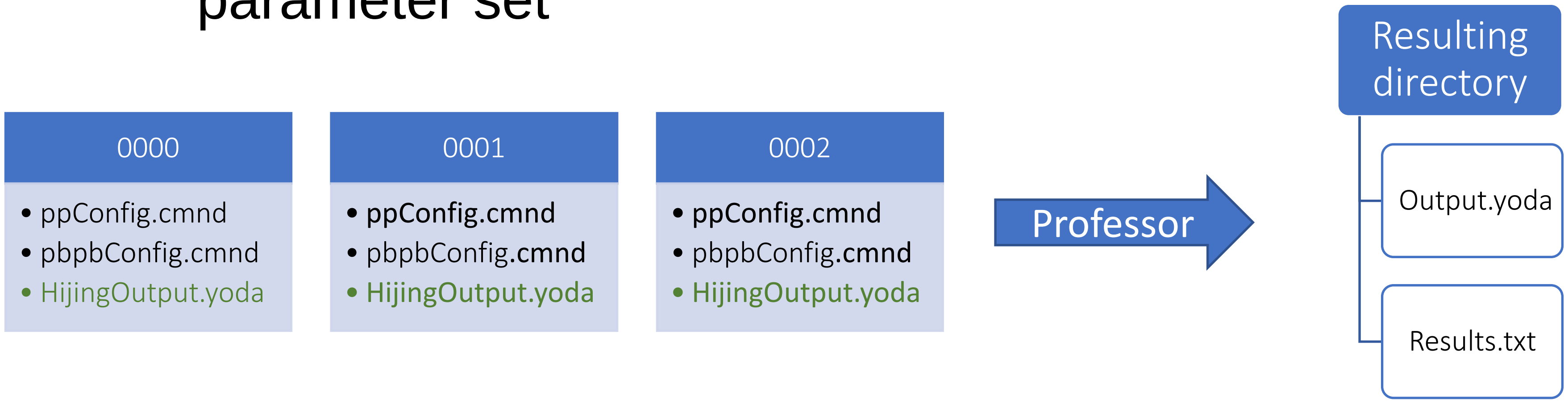


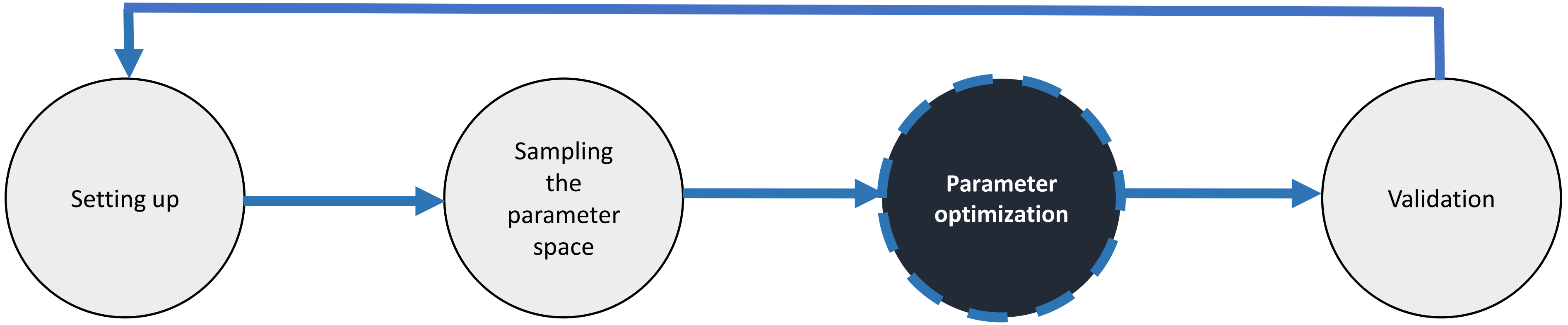
- The sampling is done by Professor (prof2-scan)
- Each subdirectory is a different configuration for HIJING++
- After sampling we run HIJING++ in each subdirectory
- Depending on the sample size it can take a few days to complete

./main_hepmc2 ***pbpbConfig.cmnd*** *valid_2760pp & rivet --quiet --analyses=ALICE_2015_I1394854,MC_GENERIC,MC_IDENTIFIED,MC_XS --analysis-path-append /data/majorosb_data/rivet/ALICE_2015_I1394854/ valid_2760pp.hepmc -o* ***HijingOutput.yoda;***



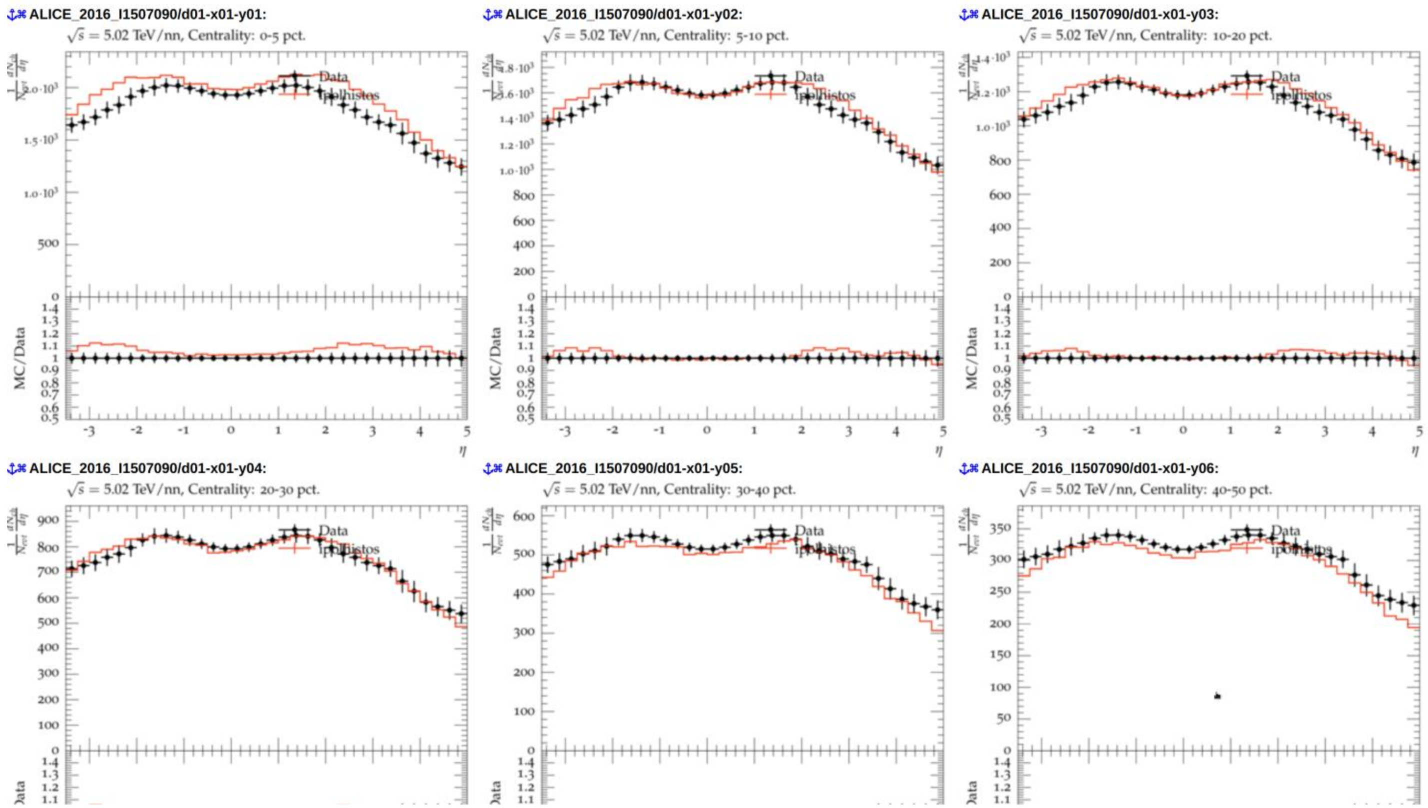
- Lots of intermidiate steps are involved
- Finally we call (prof2-tune) with a weight file
- We try find the best fit, and continue with that parameter set



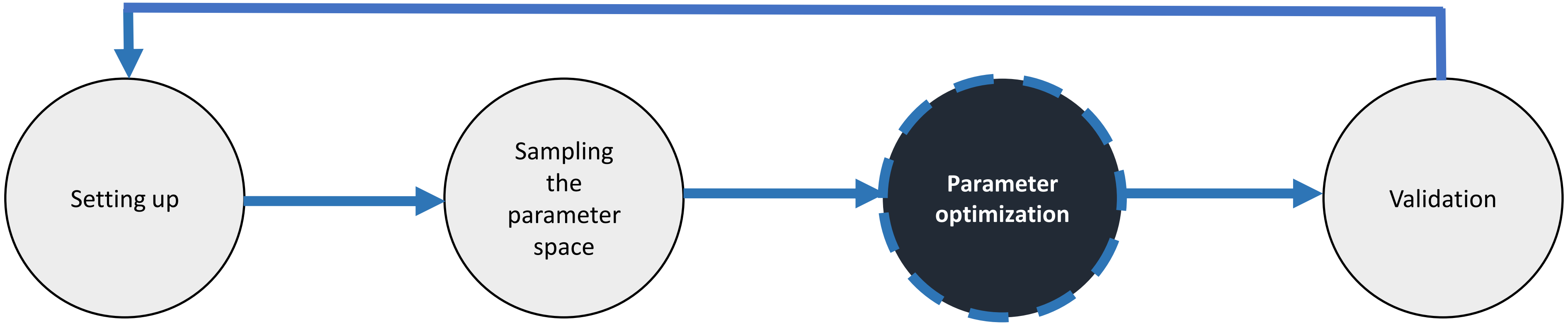



```
# ProfVersion: 2.3.3
# Date: 2021-11-09 00:07:11
# InterpolationFile: /mnt/wigner/majorosb_data/2021/Delta/ipol2.dat
# DataDirectory: /home/bors/AUR/rivet/rivet/pkg/rivet/usr/share/Rivet
#
# Limits:
#       hardcut         3.415969 5.974541
#       minjetpt        1.019177 3.981295
#       formfactscale   1.067201 3.999901
#       mu0             3.500261 4.197911
#       quenchmindist   0.106022 0.976270
#       qhat            0.213240 2.991320
#
# Fixed:
#
# Minimisation result:
#
# GOF 1214.734232
# UNITGOF 1214.734232
# NDOF 334.000000
hardcut         3.714634
minjetpt        2.337209
formfactscale   3.999901
mu0             4.197911
quenchmindist   0.111619
qhat            0.866300
#
# MIGRAD errors:
#
# hardcut       1.557659e-02
# minjetpt      6.102153e-02
# formfactscale 3.746009e-03
```

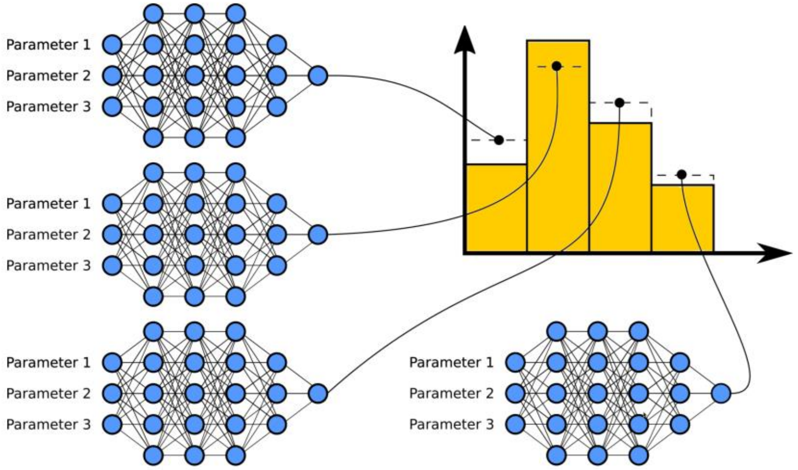
[Result.txt]



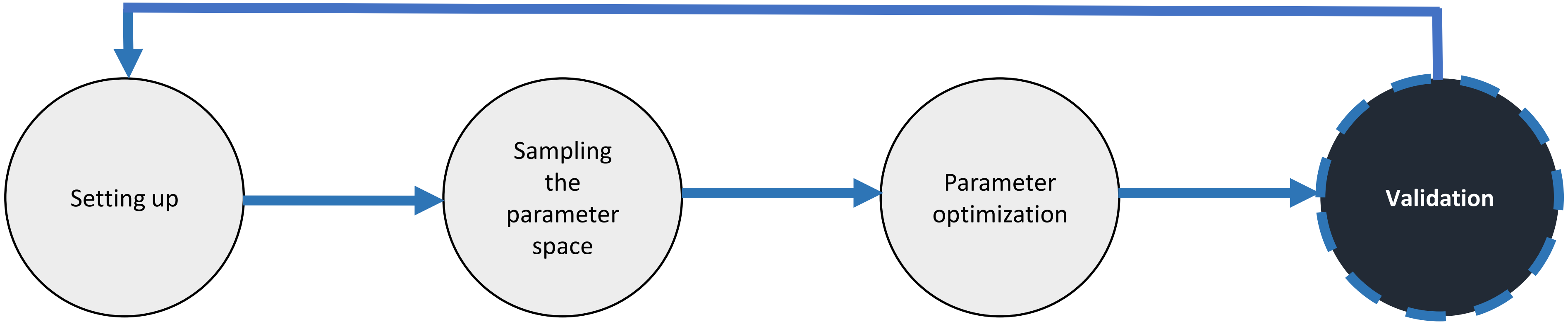
[Output.yoda]



- MCNNUTNES: A machine learning based optimization tool
- Per bin model
- Feedforward neural network for each bin
- Tuning step is the same with Professor: minimize the χ^2 function

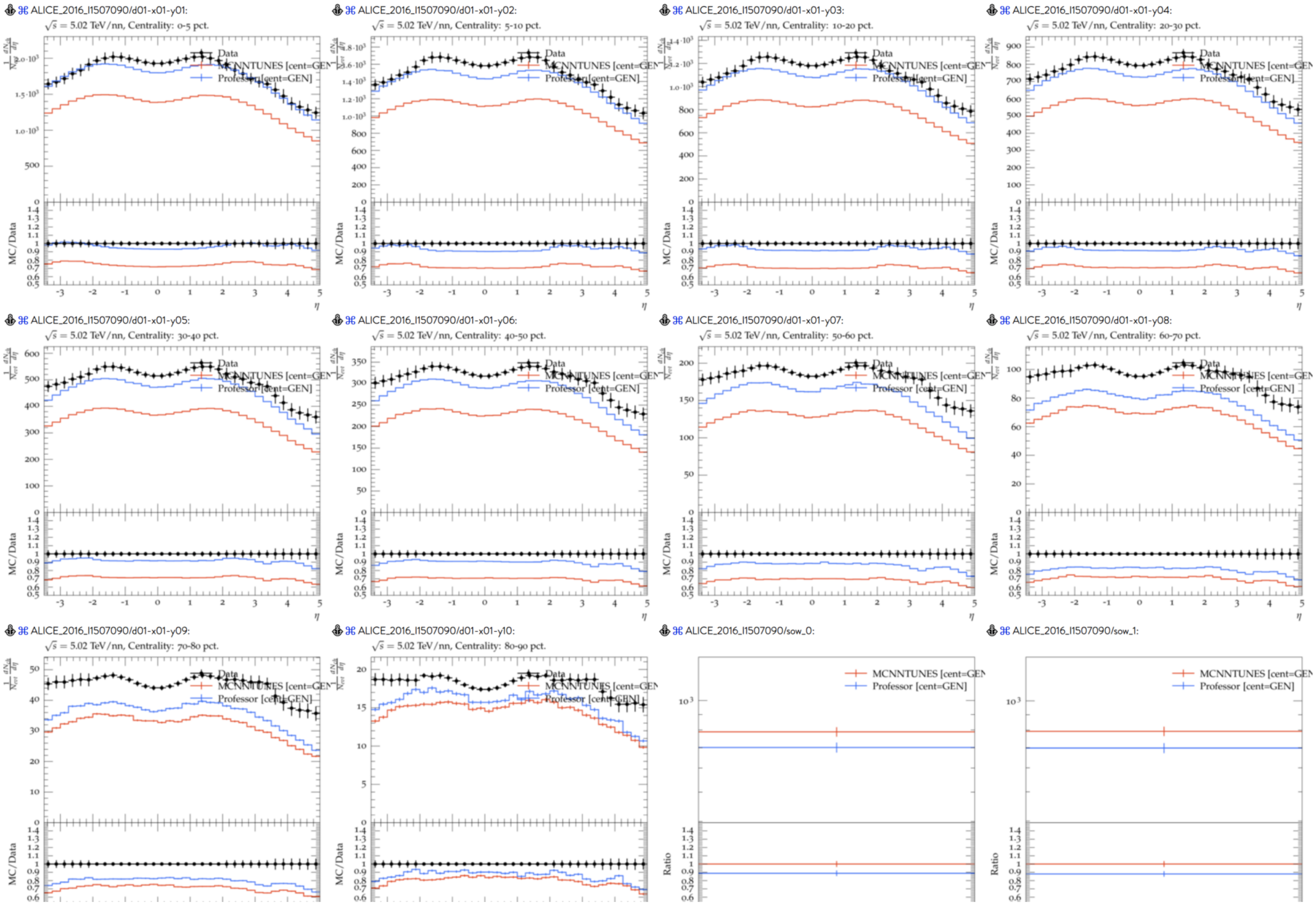


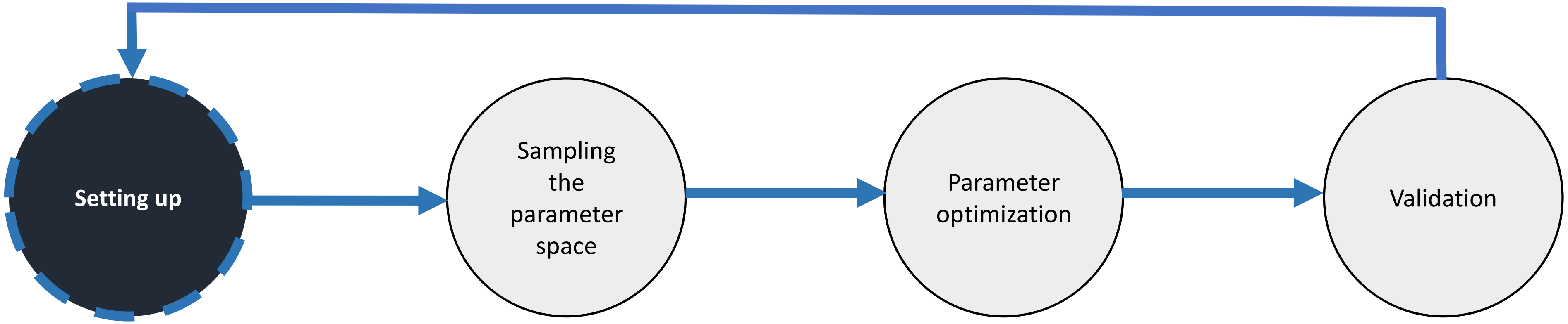
$$\chi^2(p) = \sum_{i=1}^{N} \frac{(h^{(i)}(p) - h_{exp}^{(i)})^2}{\sigma_{(i)}^2}$$



Energy: 5.02 TeV
System: Pb-Pb
Experimental analysis: ALICE_2016_I1507090
HEPData: https://www.hepdata.net/record/ins1507090

Parameters	Ranges	Professor	MCNNTUNES
HardCut	3.4 – 6.0	3.714634	4.655653
MinJetPt	1.0 – 4.0	2.337209	2.732256
FormFactScale	1.0 – 4.0	3.999999	2.506656
Mu0	3.2 – 4.2	4.197911	3.857968
QuenchMinDist	0.1 – 1.0	0.111619	1.574622
Qhat	0.2 – 3.0	0.866300	0.503265


ALICE_2016_I1507090/d01-x01-y01:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 0-5 pct.
ALICE_2016_I1507090/d01-x01-y02:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 5-10 pct.
ALICE_2016_I1507090/d01-x01-y03:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 10-20 pct.
ALICE_2016_I1507090/d01-x01-y04:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 20-30 pct.
ALICE_2016_I1507090/d01-x01-y05:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 30-40 pct.
ALICE_2016_I1507090/d01-x01-y06:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 40-50 pct.
ALICE_2016_I1507090/d01-x01-y07:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 50-60 pct.
ALICE_2016_I1507090/d01-x01-y08:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 60-70 pct.
ALICE_2016_I1507090/d01-x01-y09:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 70-80 pct.
ALICE_2016_I1507090/d01-x01-y10:
$\sqrt{s}$ = 5.02 TeV/nn, Centrality: 80-90 pct.
ALICE_2016_I1507090/sow_0:
ALICE_2016_I1507090/sow_1:
Data
MCNNTUNES [cent=GEN]
Professor [cent=GEN]
MC/Data
Ratio
$\frac{1}{N_{evt}}\frac{dN_{ch}}{d\eta}$
η



Summary

- The validation runs are in favour of Professor, but it seems MCNNTUNES needs more data or a different configuration, to make better predictions.
- MCNNTUNES is a bit slower than Professor (hardware probably can solve this problem)
- Predictions can still be improved, so back to step 1. ☺
- Currently the tune is still ongoing for the HIJING++

Special thanks to:

Wigner GPU Laboratory

Wigner Data Center

Thank You for the attention!

The research was funded by K135515, 2019-2.1.6-NEMZKI-2019-00011

References:

[1] https://wignerdc.wigner.hu/home

[2] https://rivet.hepforge.org

[3] https://yoda.hepforge.org

[4] https://professor.hepforge.org

[5] https://github.com/N3PDF/mcnntunes