The interface of BLOB with Geant4

Carlo Mancini Terracciano
carlo.mancini.terracciano@roma1.infn.it

24th April 2019
ENSAR2 workshop:
Geant4 in Nuclear Physics
CIEMAT Madrid, Spain

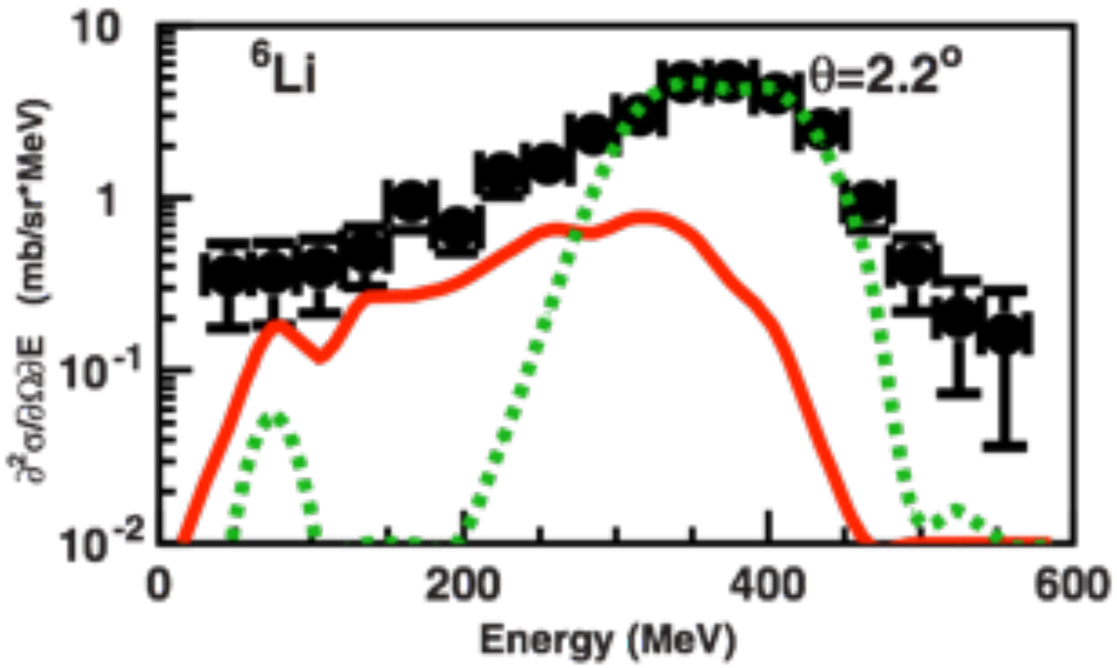
INFN
Istituto Nazionale di Fisica Nucleare
SAPIENZA
UNIVERSITÀ DI ROMA

Problems below 100MeV/A

- Despite the numerous and relevant application would use it, there is no dedicated model to nuclear interaction below 100 MeV/n in Geant4
- Many papers showed the difficulties of Geant4 in this energy domain:
  - Braunn et al. have shown discrepancies up to one order of magnitude in ^{12}C fragmentation at 95 MeV/n on thick PMMA target
  - De Napoli et al. showed discrepancy specially on angular distribution of the secondaries emitted in the interaction of 62 MeV/n ^{12}C on thin carbon target
  - Dudouet et al. found similar results with a 95 MeV/n ^{12}C beam on H, C, O, Al and Ti targets

- **Exp. data**
- **G4-BIC**
- **G4-QMD**

[Plot from De Napoli et al. Phys. Med. Biol., vol. 57, no. 22, pp. 7651–7671, Nov. 2012]



Cross section of the ^{6}Li production at 2.2 degree in a ^{12}C on ^{nat}C reaction at 62 MeV/n.

Suitable models

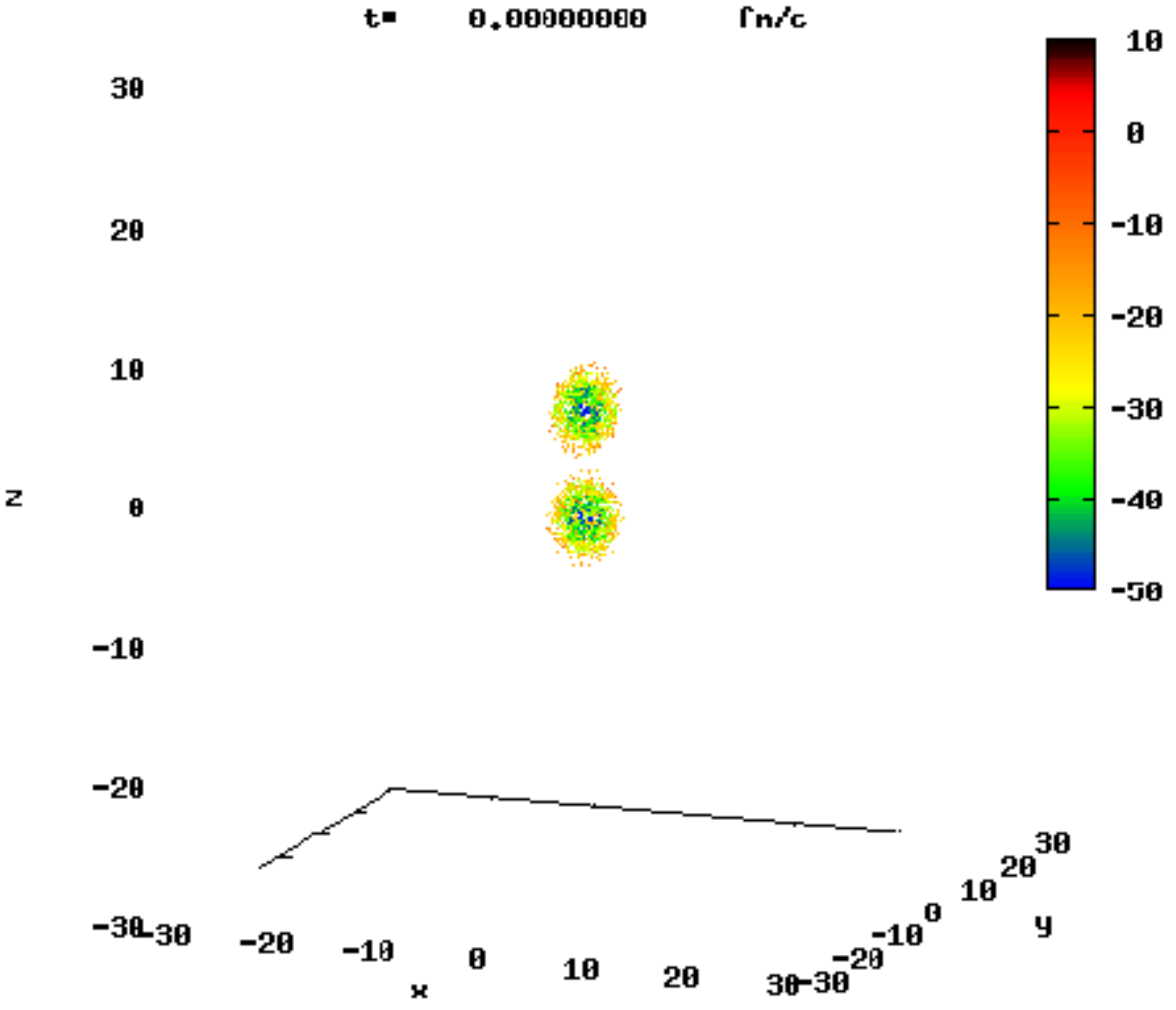
SMF (Stochastic Mean Field)

- Developed by Maria Colonna (INFN LNS, Catania)
- describes the time evolution of the density distribution
- involves the implementation of an effective attractive mean-field nuclear interaction
- mean-field is self-consistent, depends on the density
- includes two-bodies correlations through nucleon-nucleon collisions

BLOB (Boltzmann-Lagevein One Body)

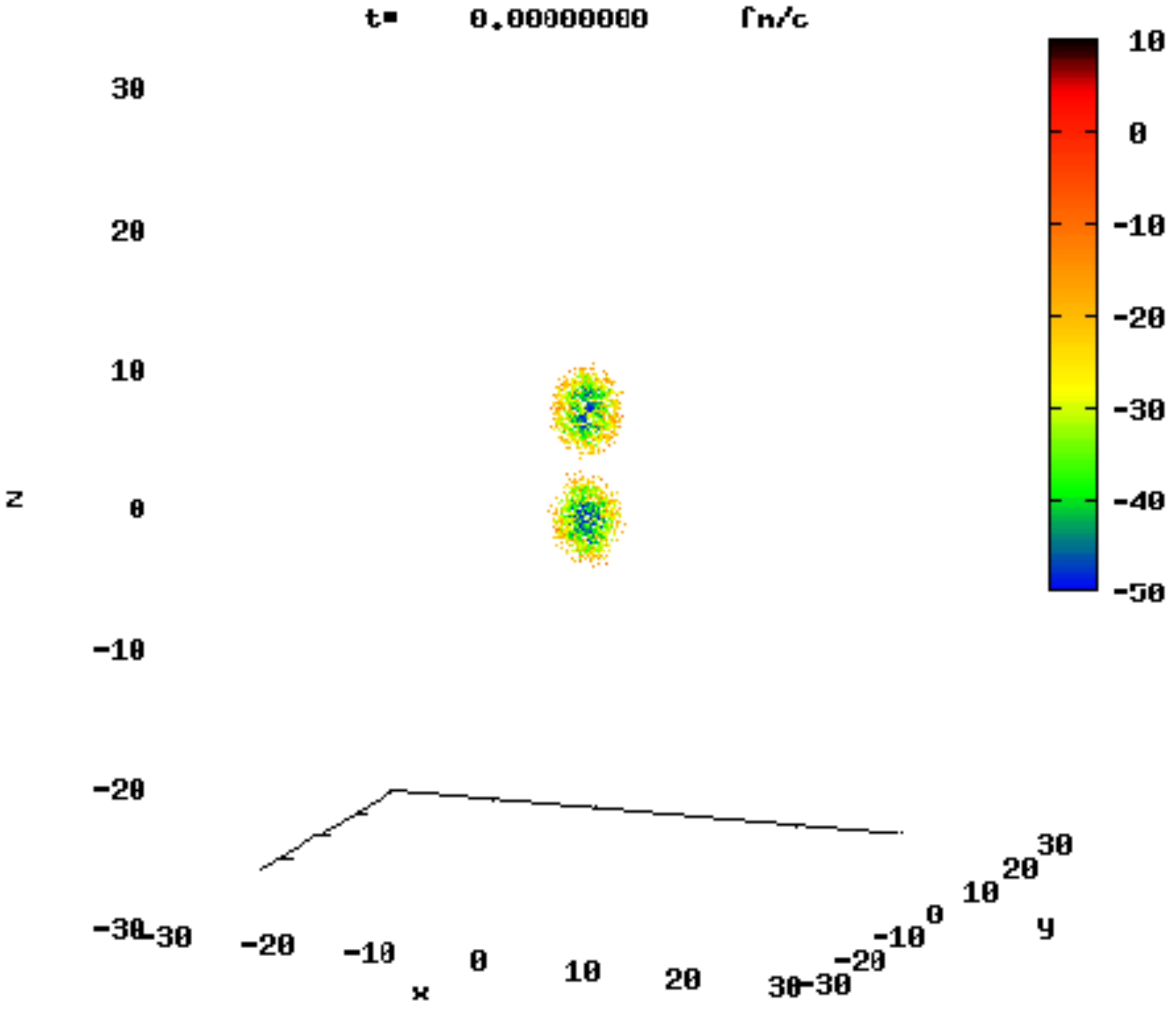
- Implemented by Paolo Napolitani (IPN, Orsay)
- Derived from SMF
- Adds fluctuations in the dynamics treating the nucleon-nucleon collisions as a stochastic process

SMF and BLOB



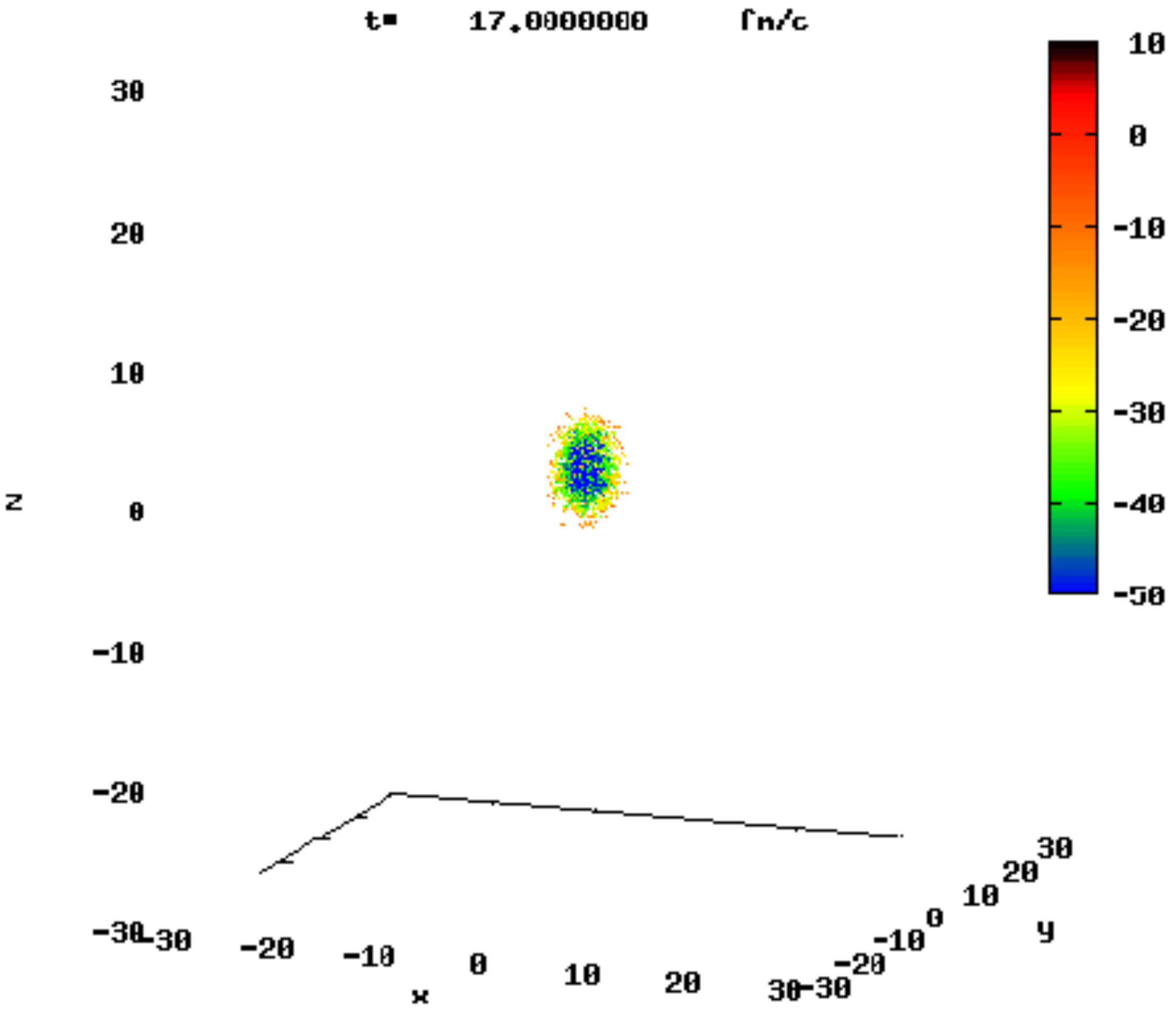
SMF

100 test particles per nucleon
^{12}C on ^{12}C at 62 MeV/n

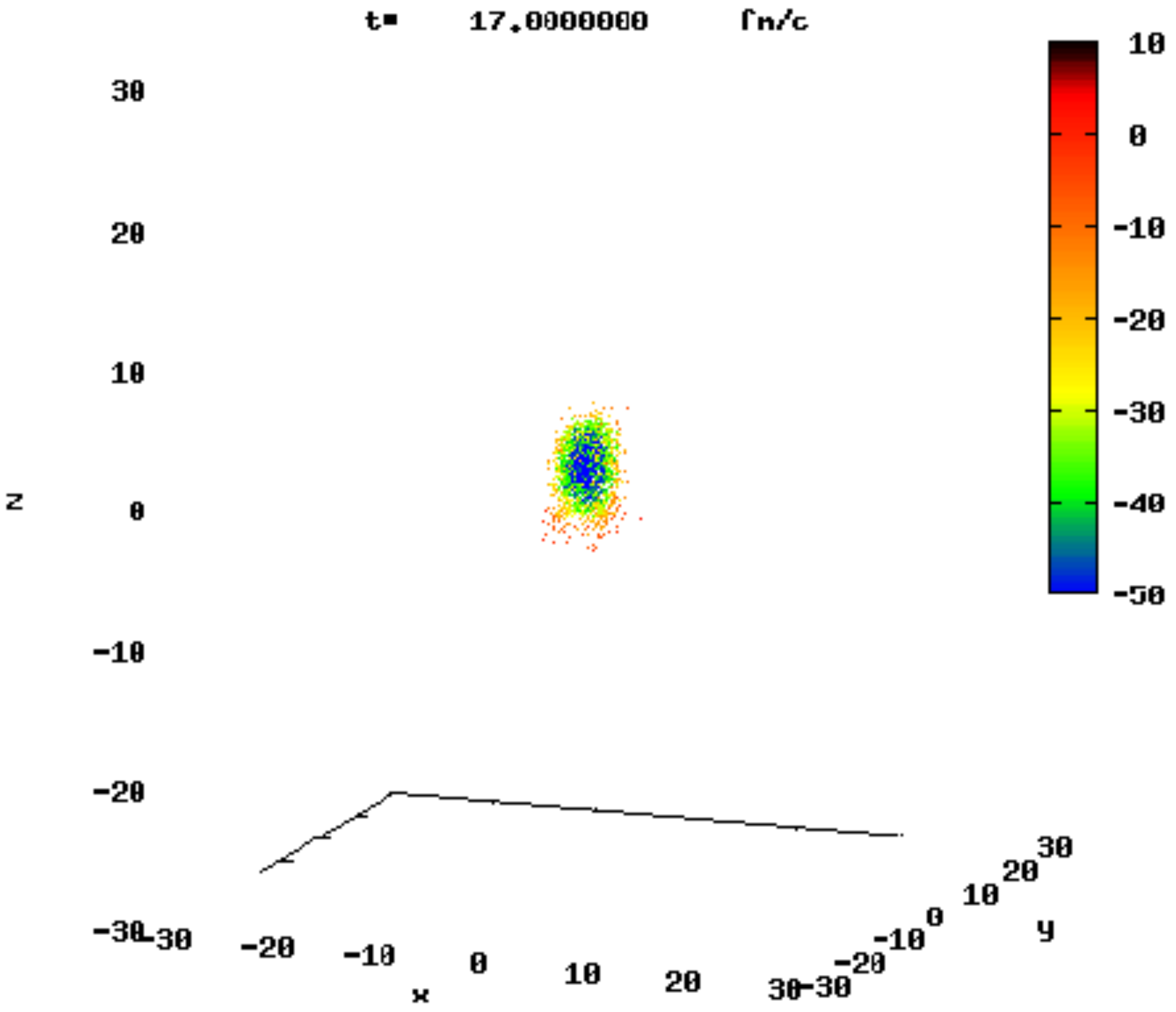


BLOB

SMF and BLOB



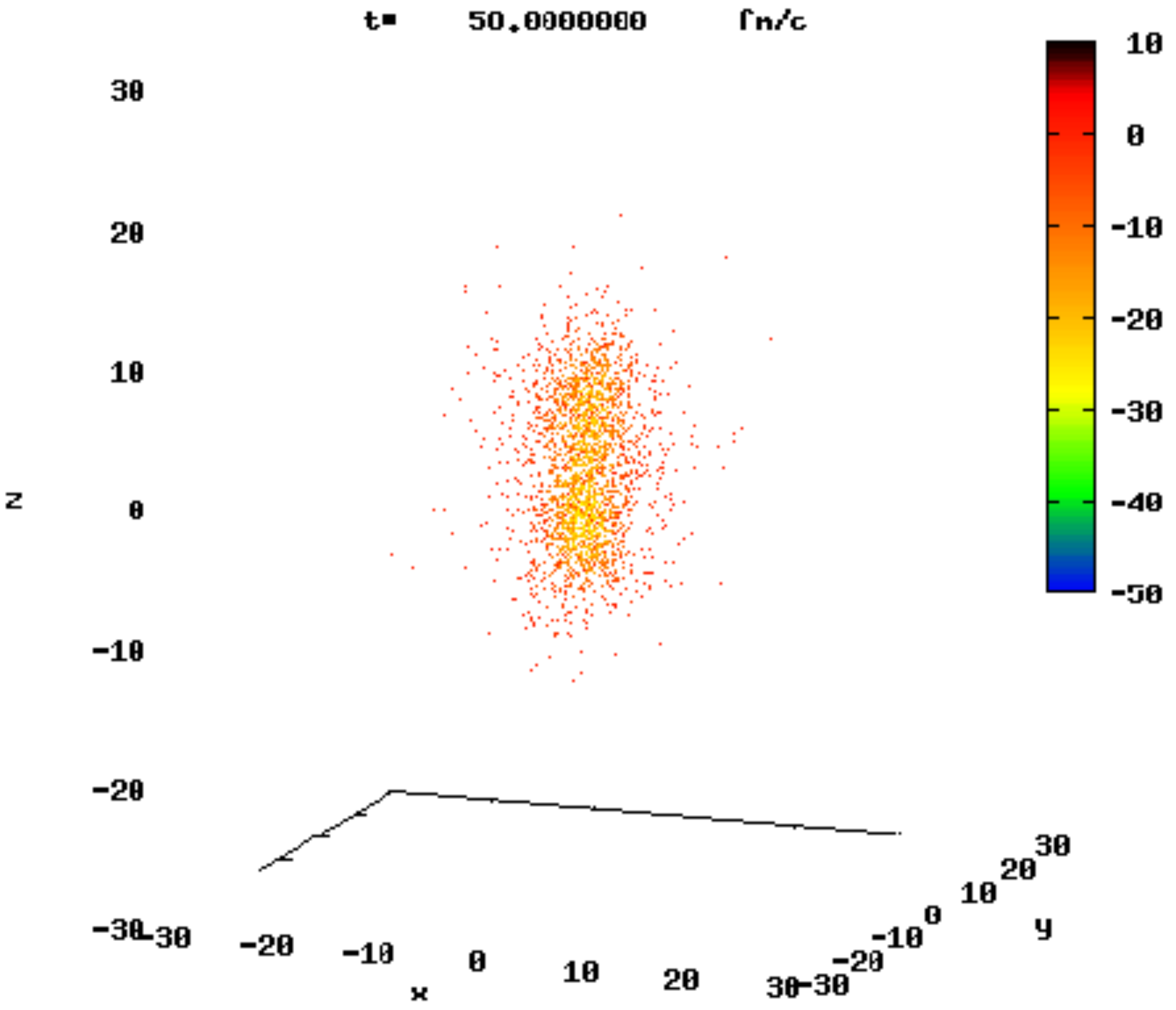
SMF



BLOB

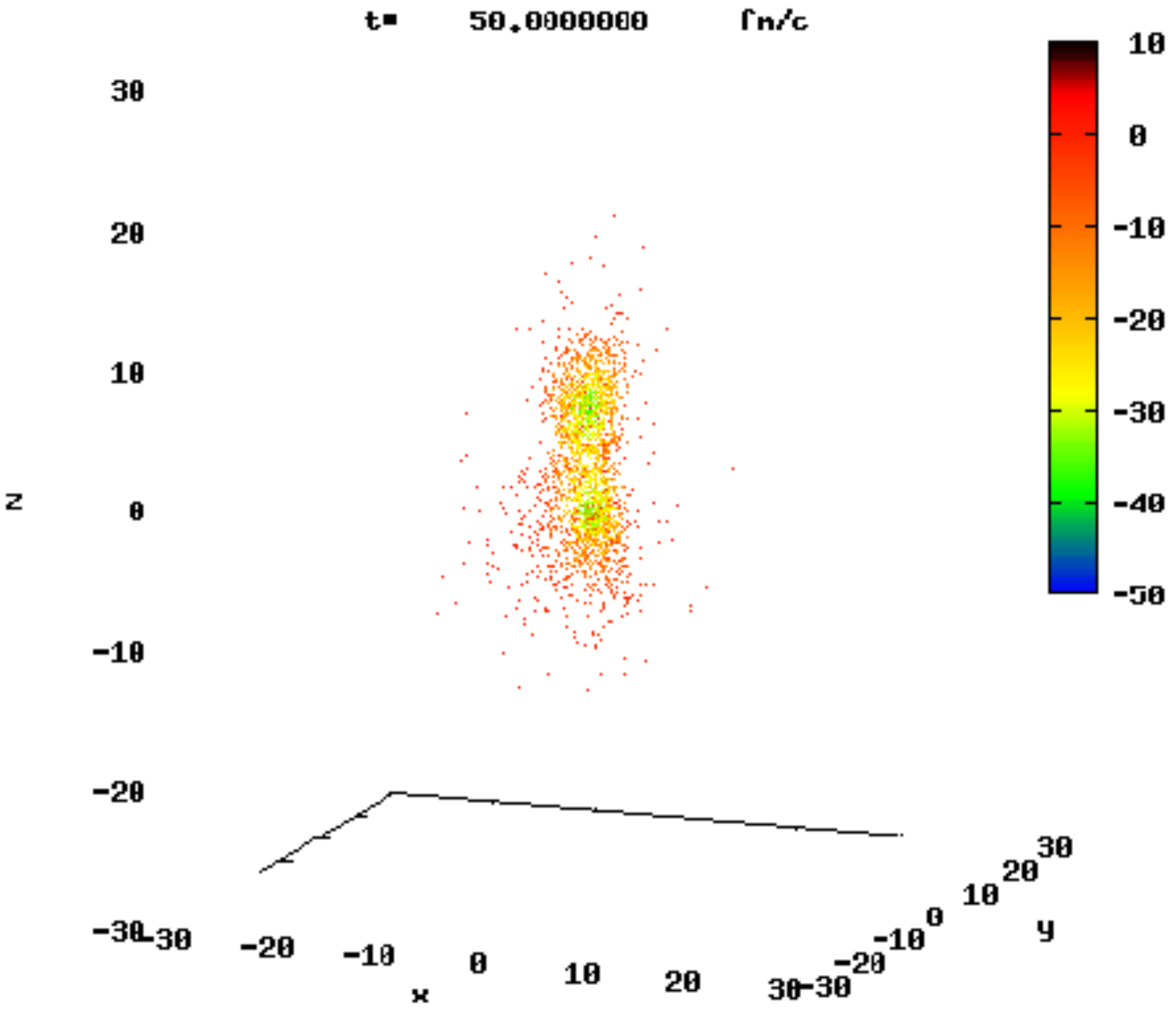
100 test particles per nucleon
^{12}C on ^{12}C at 62 MeV/n

SMF and BLOB



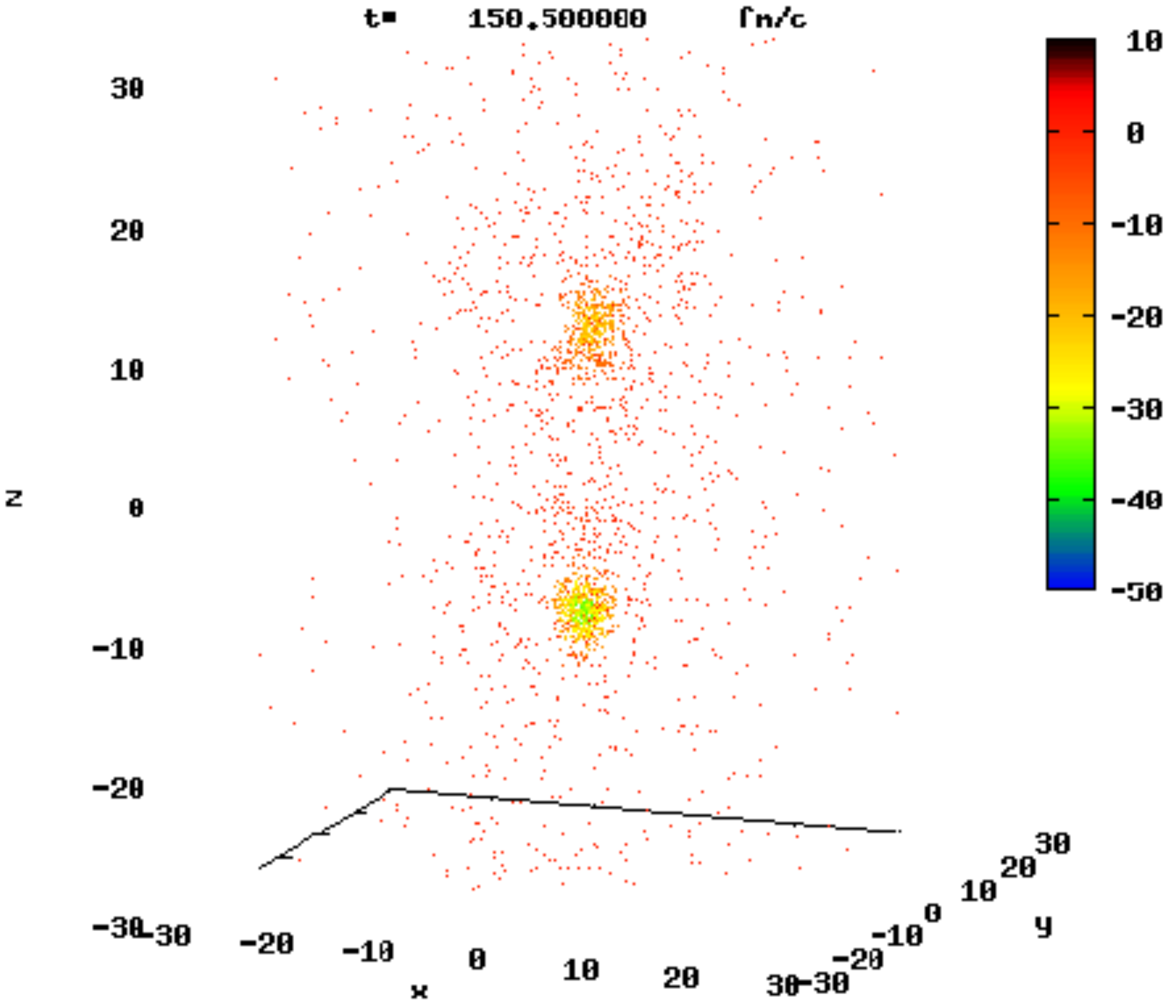
SMF

100 test particles per nucleon
^{12}C on ^{12}C at 62 MeV/n



BLOB

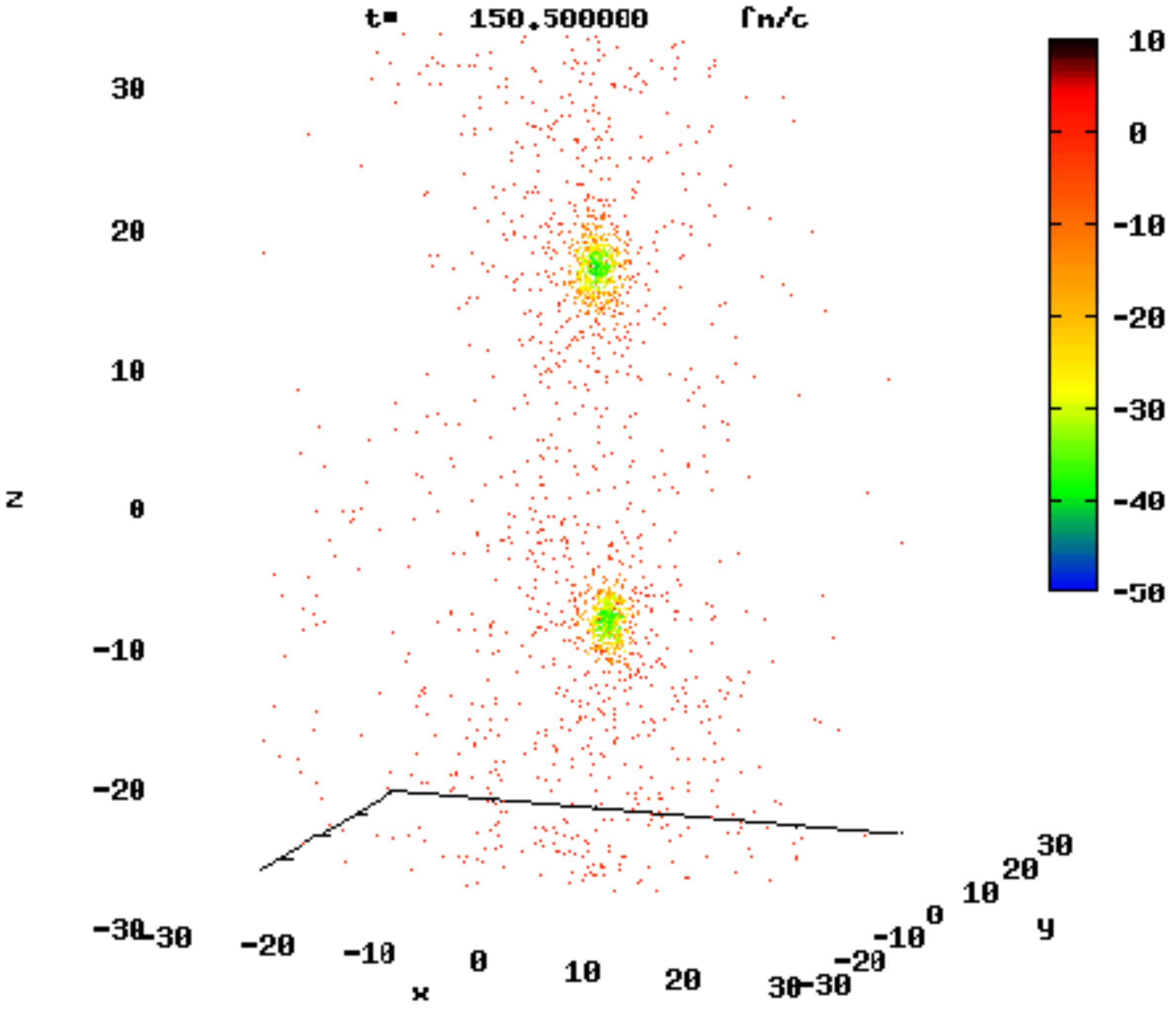
SMF and BLOB



SMF

100 test particles per nucleon

^{12}C on ^{12}C at 62 MeV/n

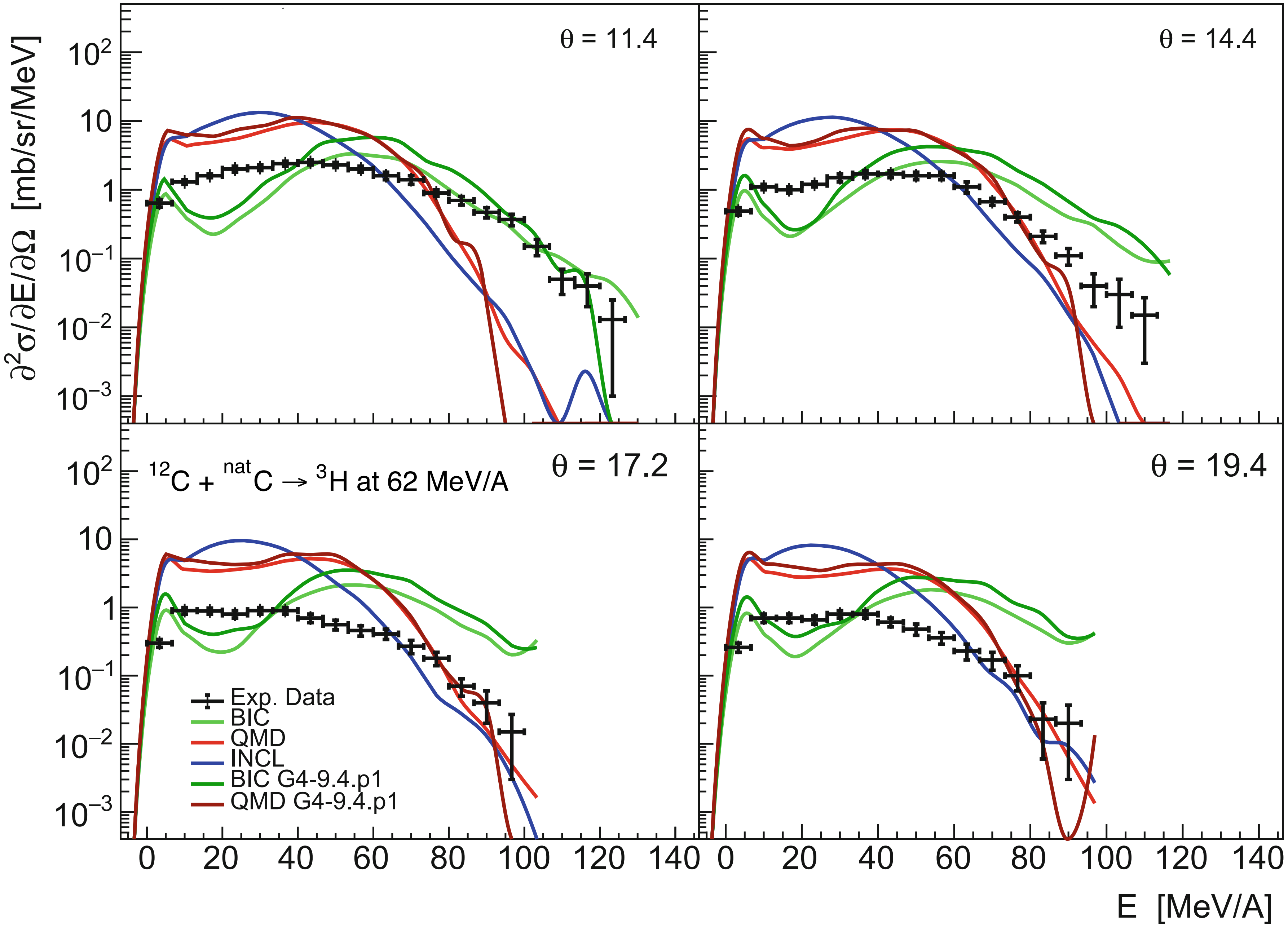


BLOB

Update of a ^{12}C fragmentation benchmark

Update of the benchmark originally published on De Napoli et al. Phys. Med. Biol., vol. 57, no. 22, pp. 7651–7671, Nov. 2012

- 62 MeV/n ^{12}C on thin carbon target
- doubly differential cross sections
- Tritium
- INCL was not available at the time of the original publication

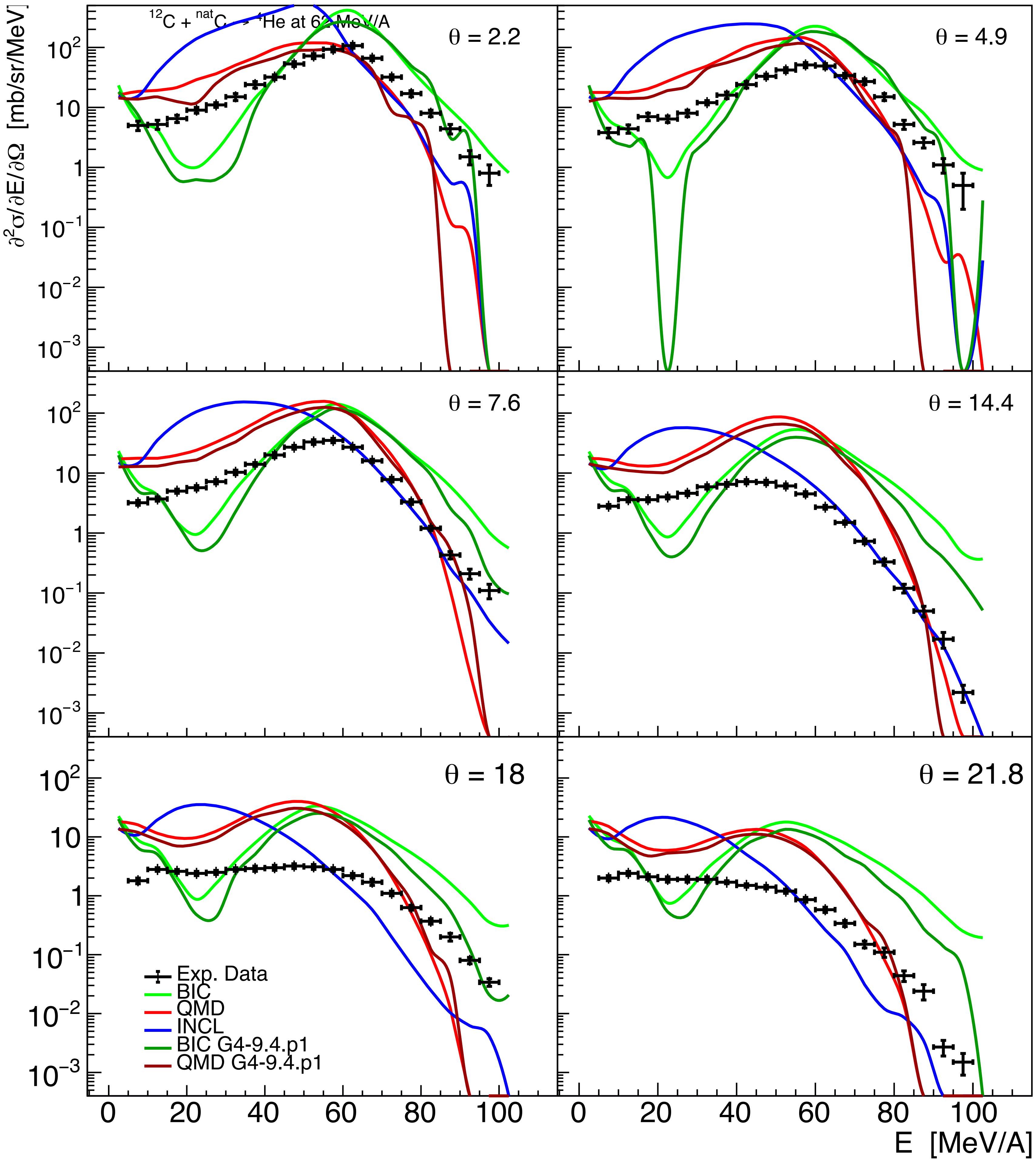


[C. Mancini-Terracciano et al. IFMBE Proceedings Series 68/1 (2018), pp. 675–685. doi: 10.1007/978-981-10-9035- 6_126]

Update of a ^{12}C fragmentation benchmark

Update of the benchmark
originally published on De Napoli
et al. Phys. Med. Biol., vol. 57, no.
22, pp. 7651–7671, Nov. 2012

- alpha



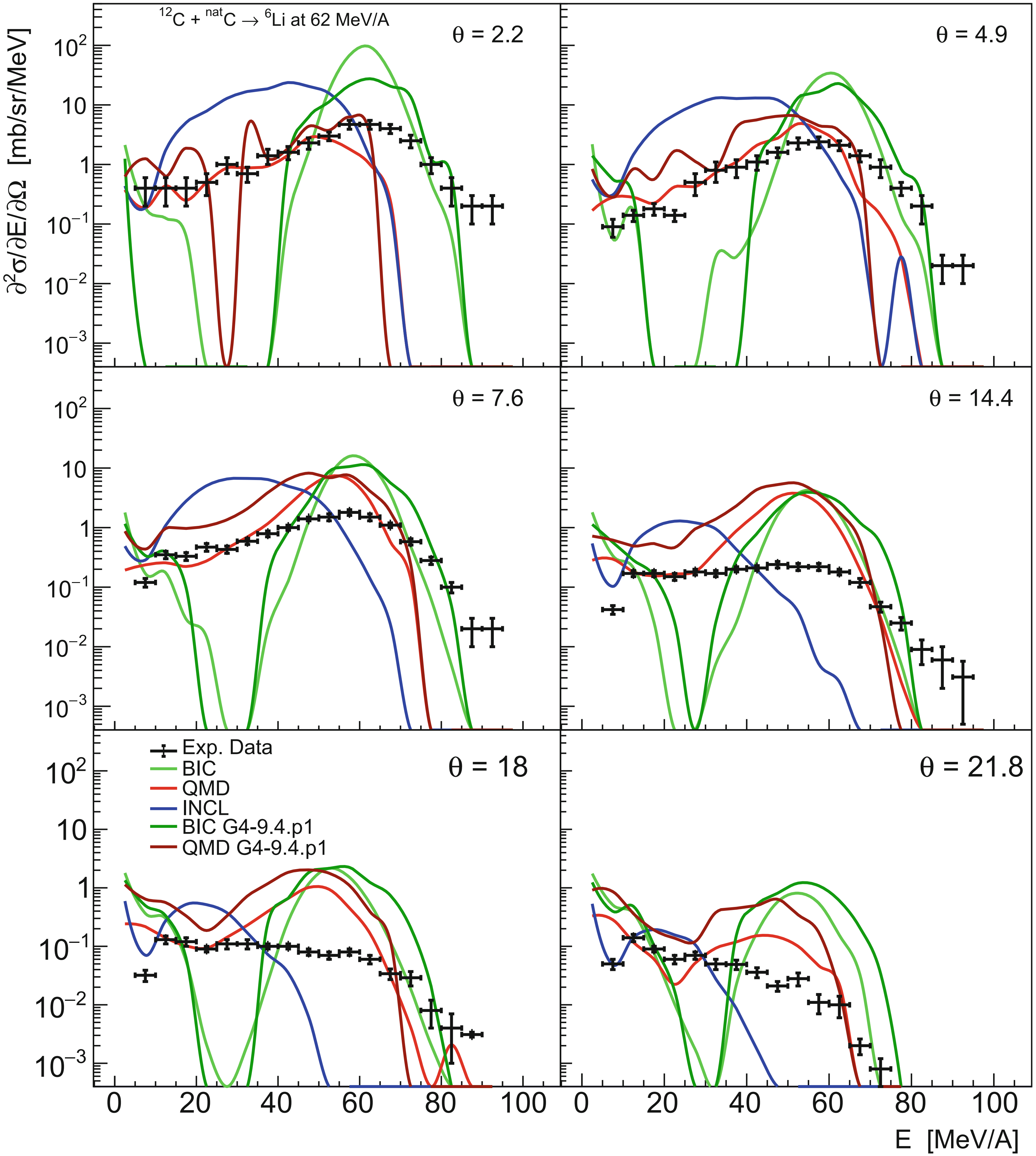
[C. Mancini-Terracciano et al. IFMBE Proceedings Series 68/1 (2018), pp. 675–685 doi: 10.1007/978-981-10-9035- 6_126]

Update of a ^{12}C fragmentation benchmark

Update of the benchmark originally published on De Napoli et al. Phys. Med. Biol., vol. 57, no. 22, pp. 7651–7671, Nov. 2012

- ^{6}Li

[C. Mancini-Terracciano et al. IFMBE Proceedings Series 68/1 (2018), pp. 675–685. doi: 10.1007/978-981-10-9035- 6_126]



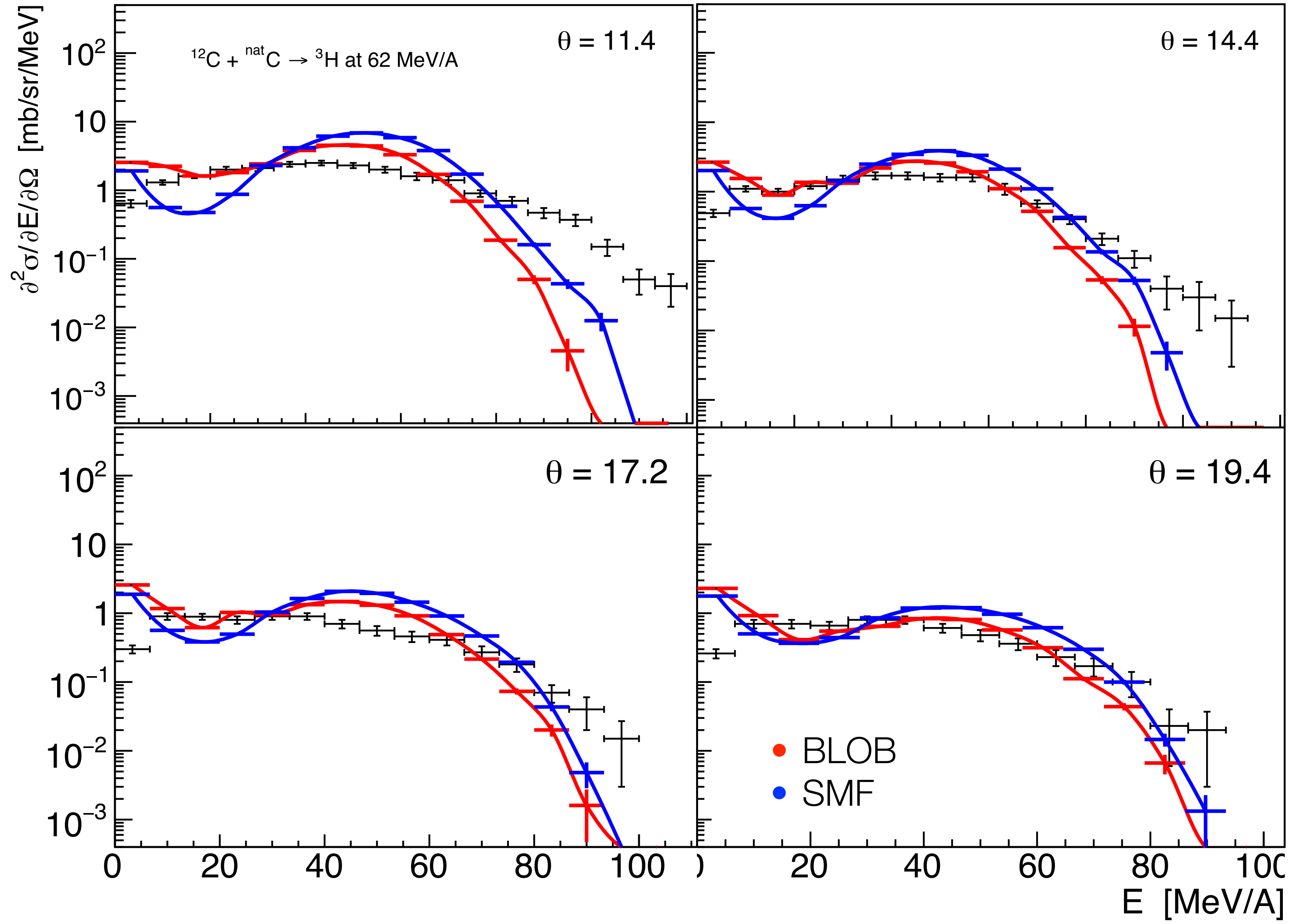
Geant4 interface to SMF and BLOB

- Developed as a G4-model
- Loads the SMF/BLOB output
- Samples the final state
  - Fragments mass and charge
  - Gas particles emitted
- Applies Geant4 de-excitation to excited fragments

Interfacing SMF and BLOB to Geant4

- SMF and BLOB had been interfaced with Geant4 and its de-excitation phase

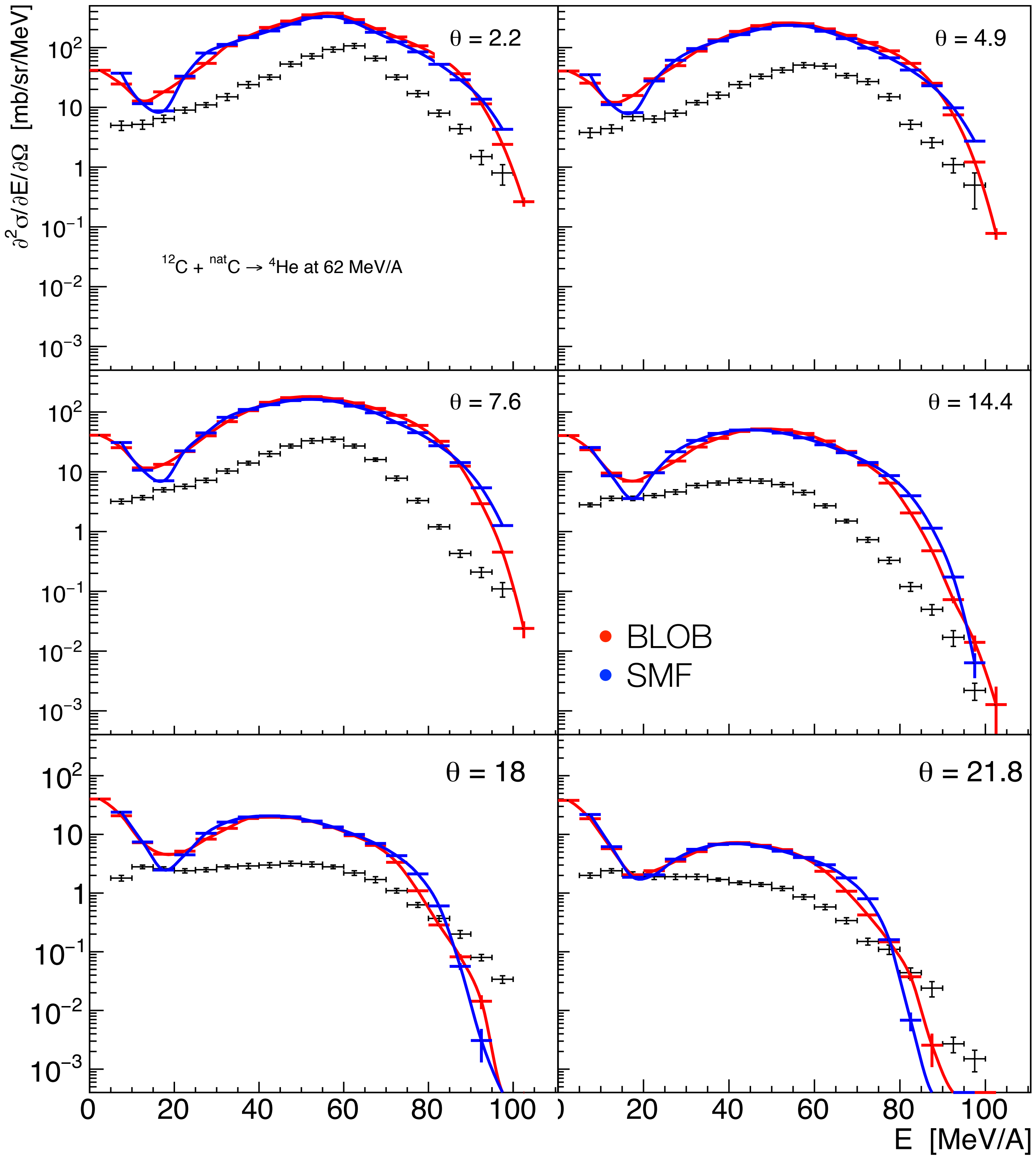
- Similar results between SMF and BLOB



Interfacing SMF and BLOB to Geant4

- SMF and BLOB had been interfaced with Geant4 and its de-excitation phase

- Similar results between SMF and BLOB

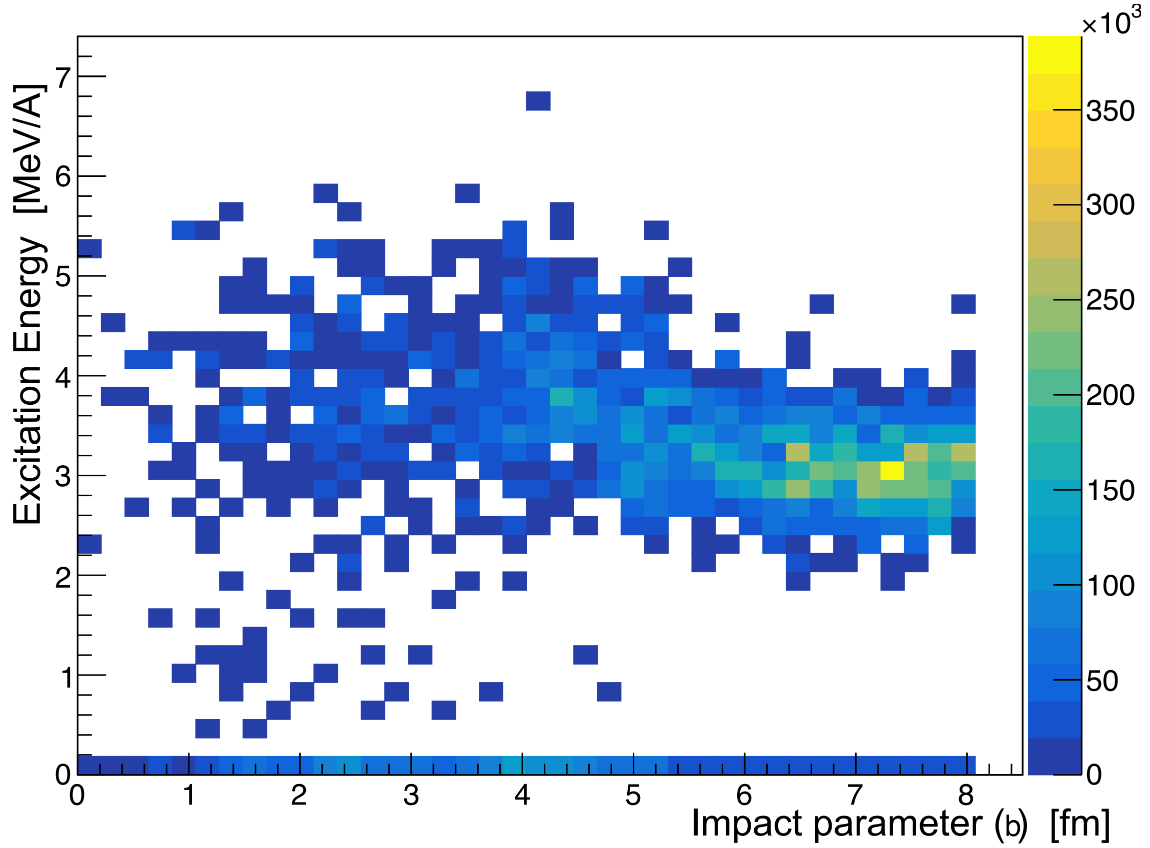


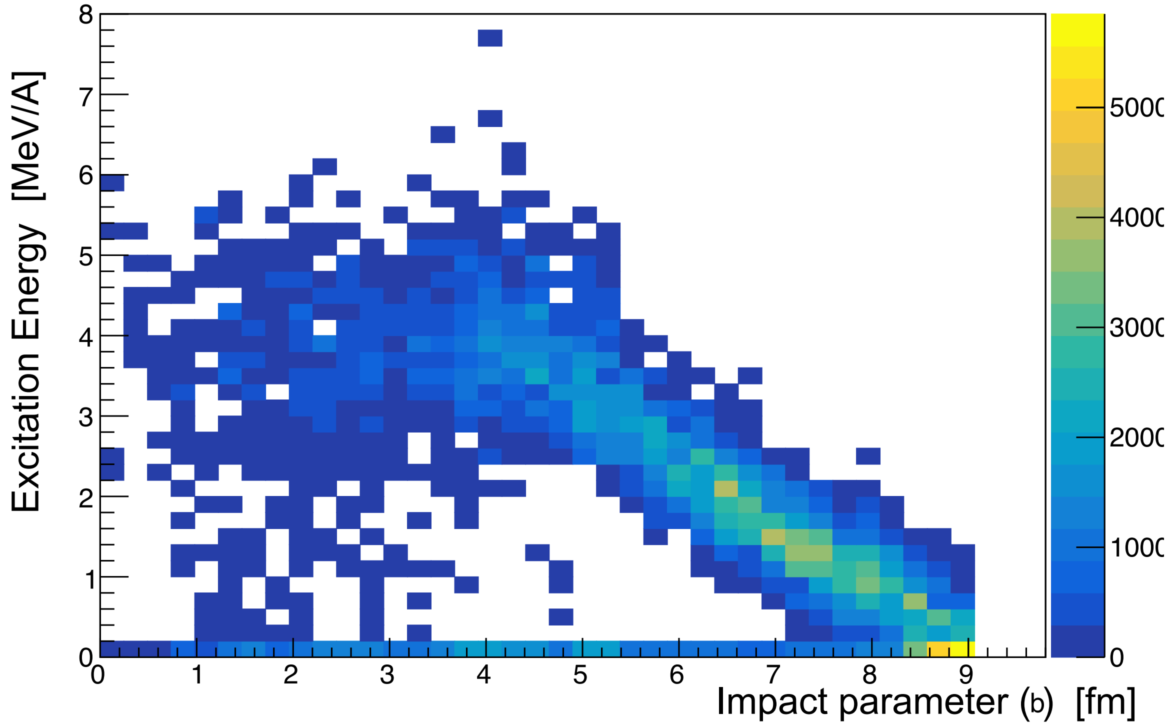
Coalescence

- To insert more than two bodies correlation in an effective way
- Implemented between SMF/BLOB and the de-excitation phase
- Two small fragments are coalesced if $\Delta x<6$ fm
- Applied recursively

Excitation energy correction

- SMF and BLOB tend to overestimate the excitation energy
- Especially for peripherals interaction
- A correction to the excitation energy has been applied
- Linear with *b*
- Up to 3 MeV/A





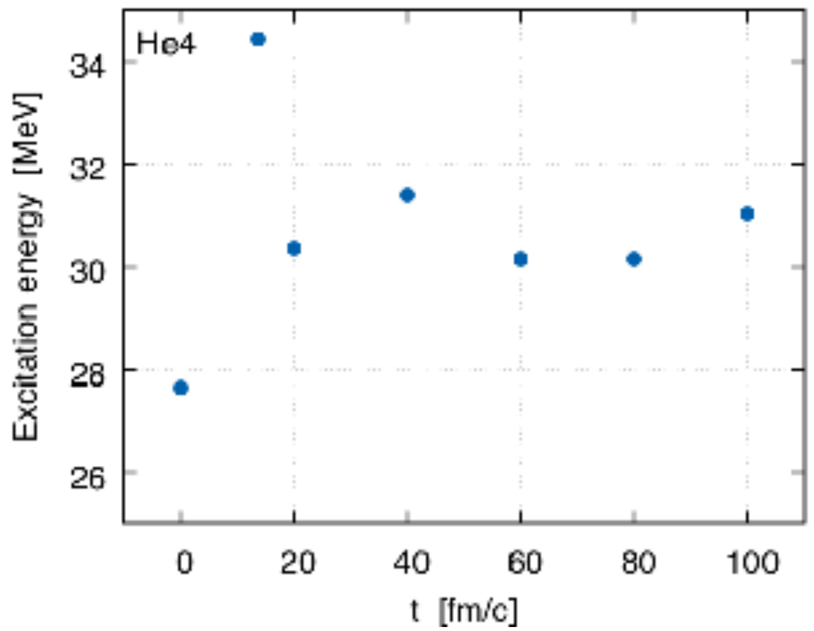
Increasing test particles number

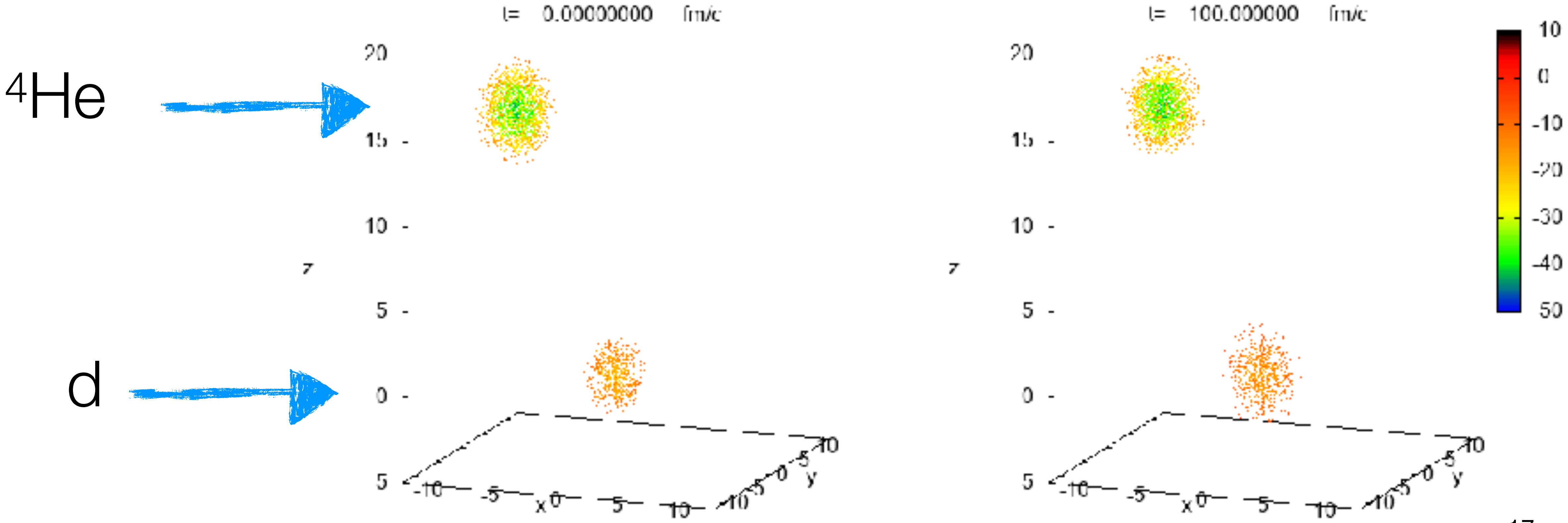
- Increasing the test particles number (from 100 per nucleon to 500 per nucleon) the excitation energy problem is mitigated

- At the moment it is not possible to increase the number of test particle even further (the arrays are not all dynamically allocated)

Correcting for ground state energy

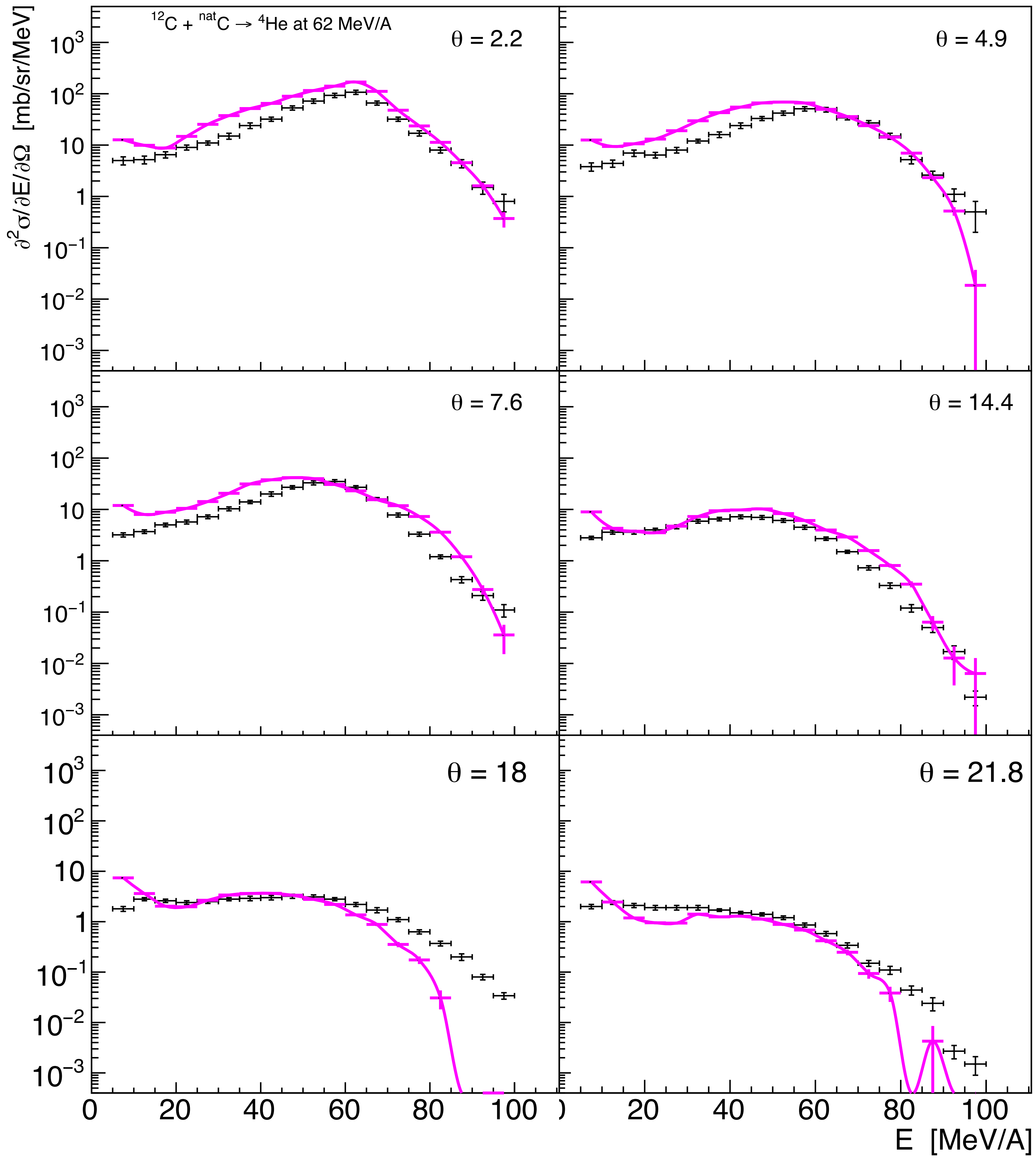
- Light nuclei are not stable in BLOB and SMF
- We calculated the spurious excitation energy of isolated light nuclei
- Subtract such energy to produced fragments





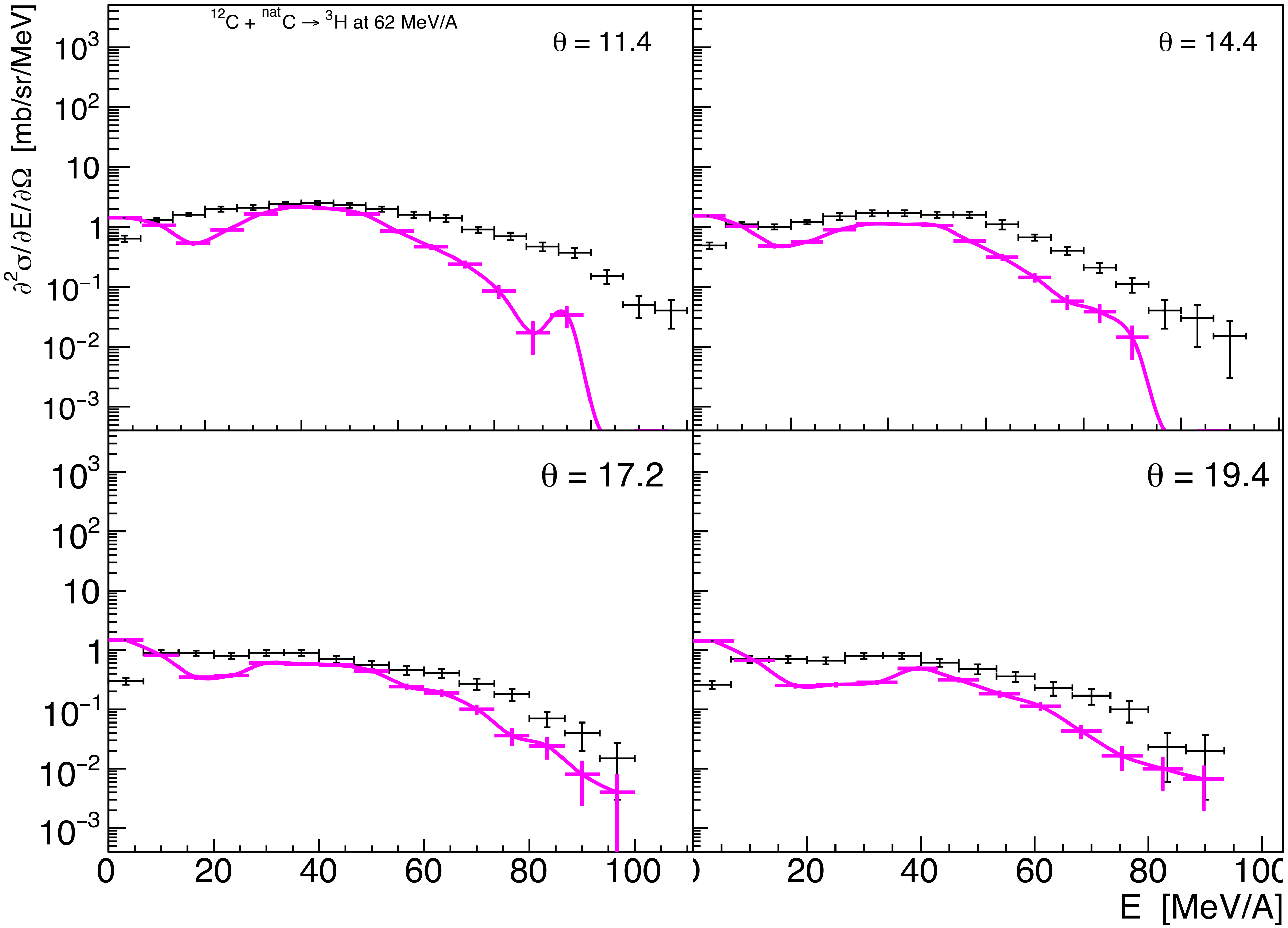
Preliminary results with BLOB and Geant4

- alpha



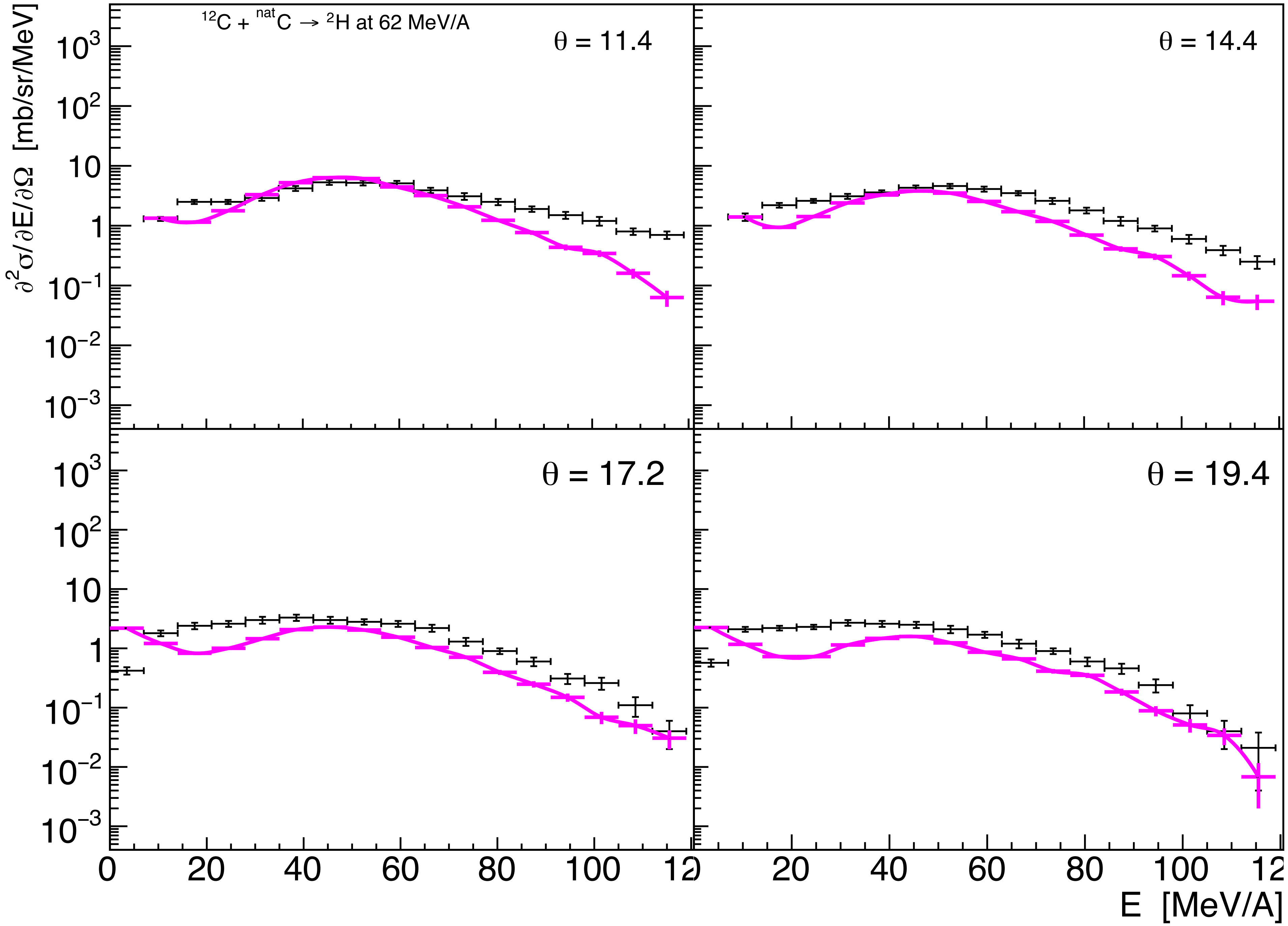
Preliminary results with BLOB and Geant4

- tritium



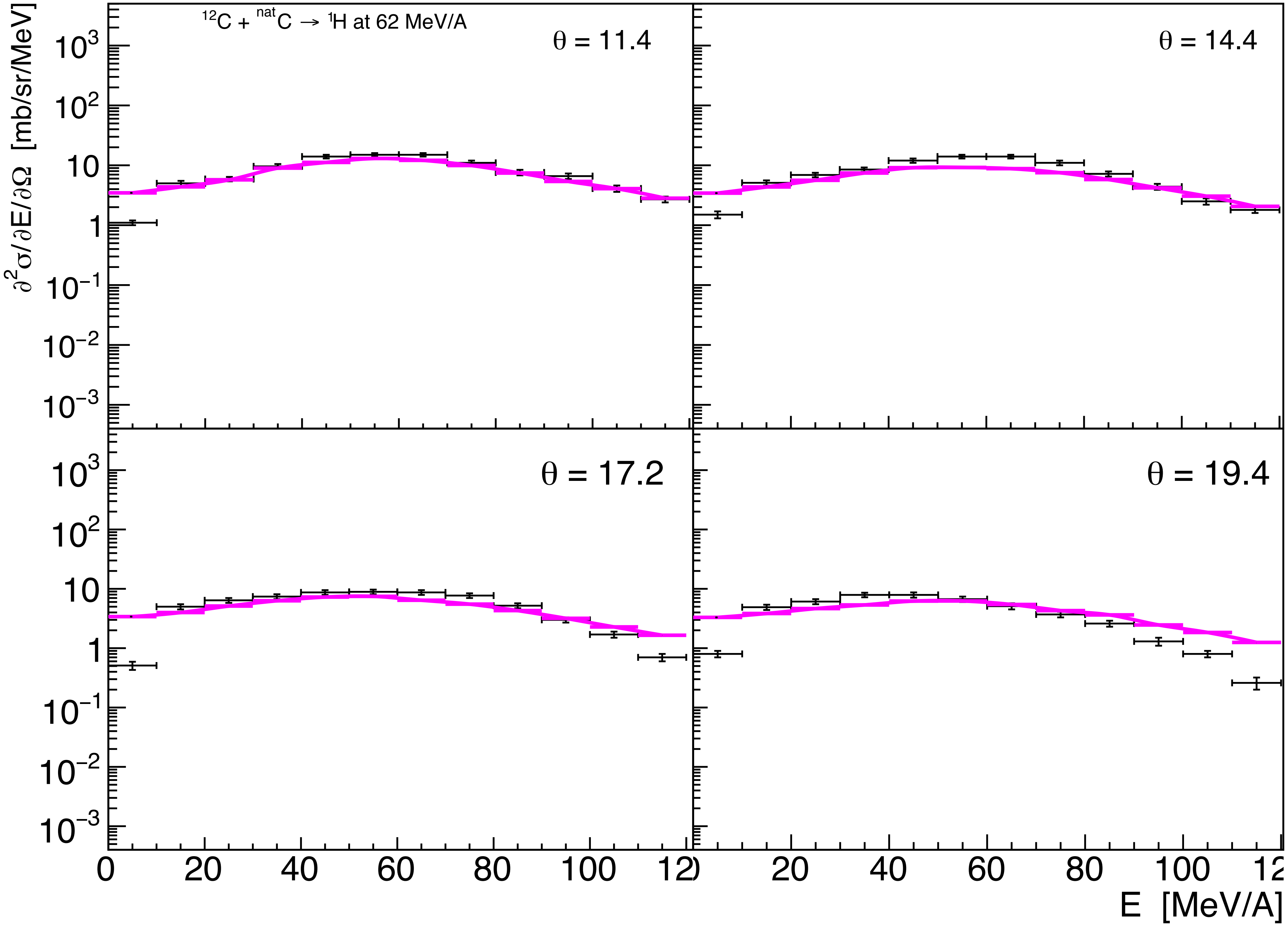
Preliminary results with BLOB and Geant4

- deuterium



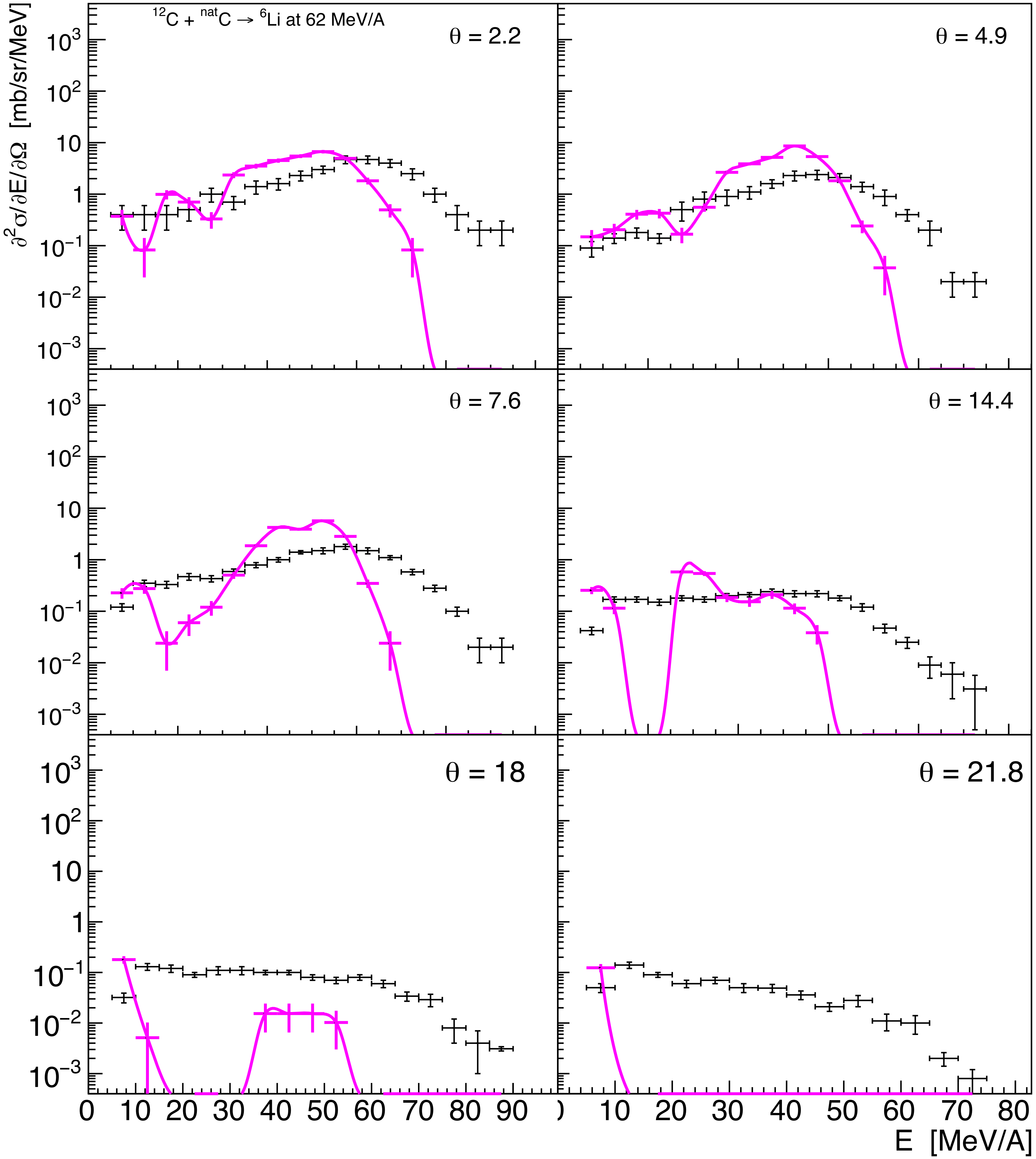
Preliminary results with BLOB and Geant4

- protons



Preliminary results with BLOB and Geant4

- ^{6}Li
- With larger fragments further improvements are needed
- We plan to modify the surface coefficient



Future plans

Short term

- Increase number of test particles to 1000
- Test changing surface coefficient

Long term

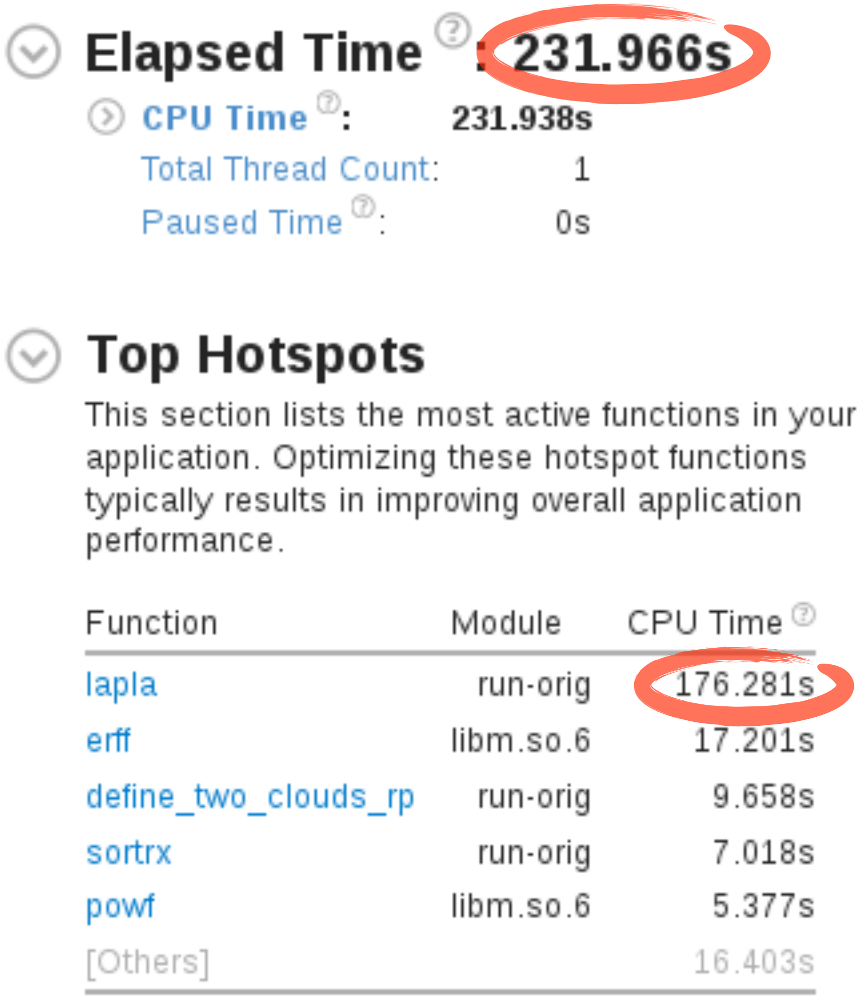
- Port the code on GPU

Not-so-long term

- Add test particle clustering to take into account 3 and 4-body interaction terms
- Benchmark with more data
- Automatise BLOB/SMF running from Geant4

Profiling BLOB

- Accurate profiling done with Intel VTune Amplifier (https://software.intel.com/en-us/vtune)
- More than 75% of the running time is spent in the function that calculates the laplacian of the mean field



Elapsed Time: 231.966s

CPU Time: 231.938s
Total Thread Count: 1
Paused Time: 0s

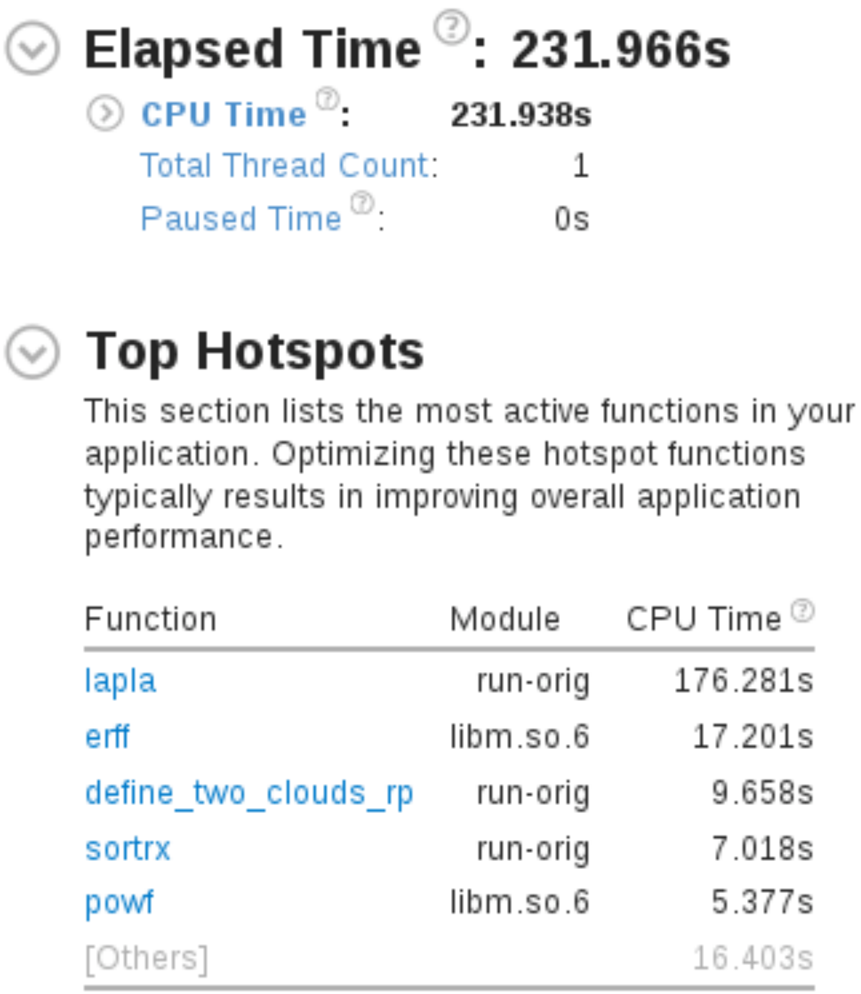
Top Hotspots

This section lists the most active functions in your application. Optimizing these hotspot functions typically results in improving overall application performance.

Function	Module	CPU Time
lapla	run-orig	176.281s
erff	libm.so.6	17.201s
define_two_clouds_rp	run-orig	9.658s
sortrx	run-orig	7.018s
powf	libm.so.6	5.377s
[Others]		16.403s

Code optimisations

- Optimisation of the function “lapla” without changing the code structure
- 68% speed-up in the function
- 52% speed-up overall



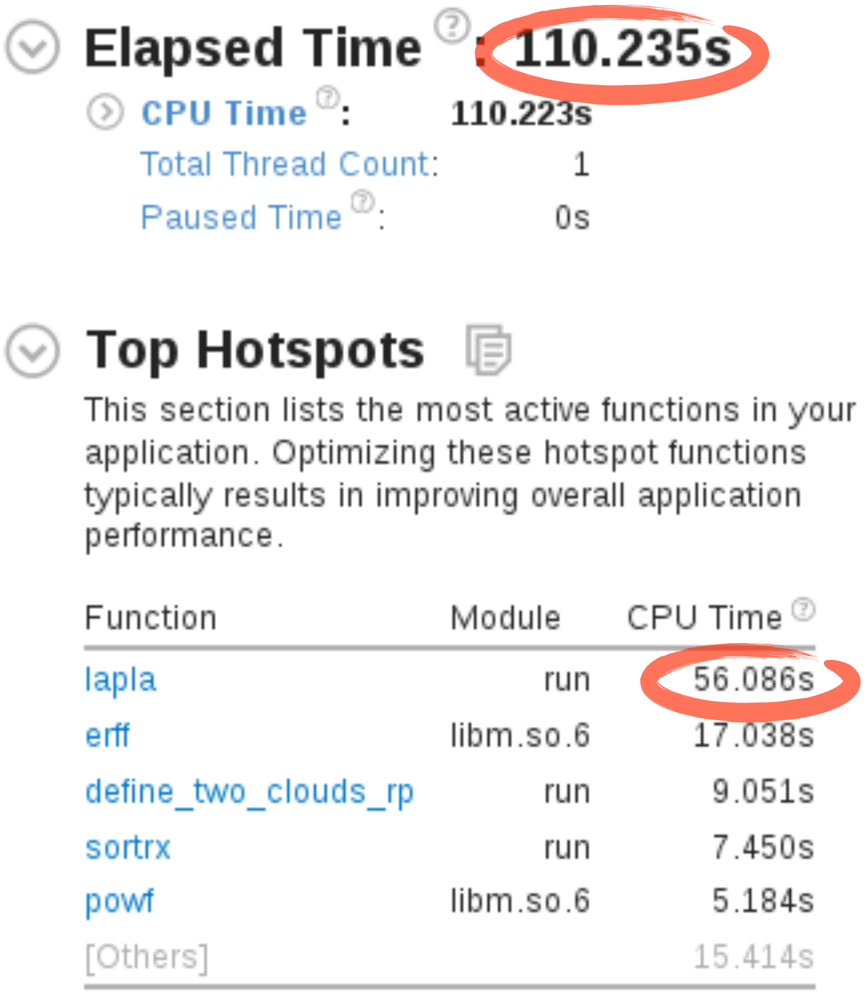
Elapsed Time: 231.966s

CPU Time: 231.938s
Total Thread Count: 1
Paused Time: 0s

Top Hotspots

This section lists the most active functions in your application. Optimizing these hotspot functions typically results in improving overall application performance.

Function	Module	CPU Time
lapla	run-orig	176.281s
erff	libm.so.6	17.201s
define_two_clouds_rp	run-orig	9.658s
sortrx	run-orig	7.018s
powf	libm.so.6	5.377s
[Others]		16.403s



Elapsed Time: 110.235s

CPU Time: 110.223s
Total Thread Count: 1
Paused Time: 0s

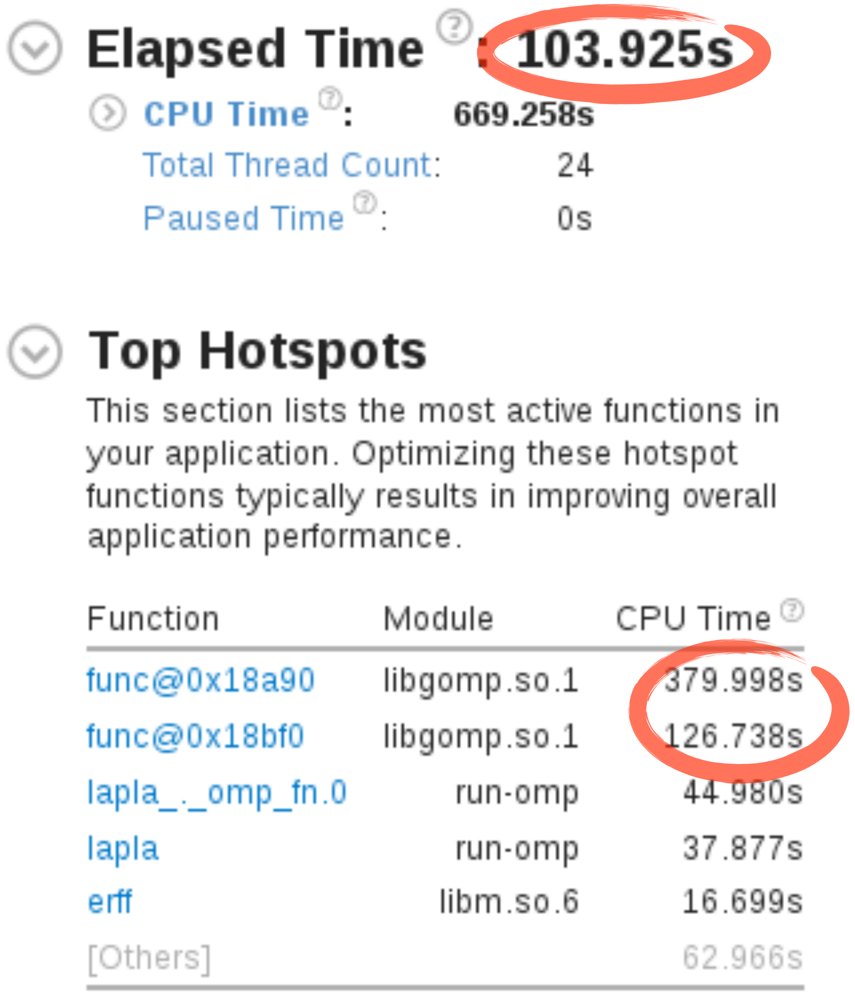
Top Hotspots

This section lists the most active functions in your application. Optimizing these hotspot functions typically results in improving overall application performance.

Function	Module	CPU Time
lapla	run	56.086s
erff	libm.so.6	17.038s
define_two_clouds_rp	run	9.051s
sortrx	run	7.450s
powf	libm.so.6	5.184s
[Others]		15.414s

Using OpenMP

- Distributing the main loop of the “lapla” on 24 cores
- Small gain overall
- A lot of time spent in distributing data



Elapsed Time: 103.925s

CPU Time: 669.258s

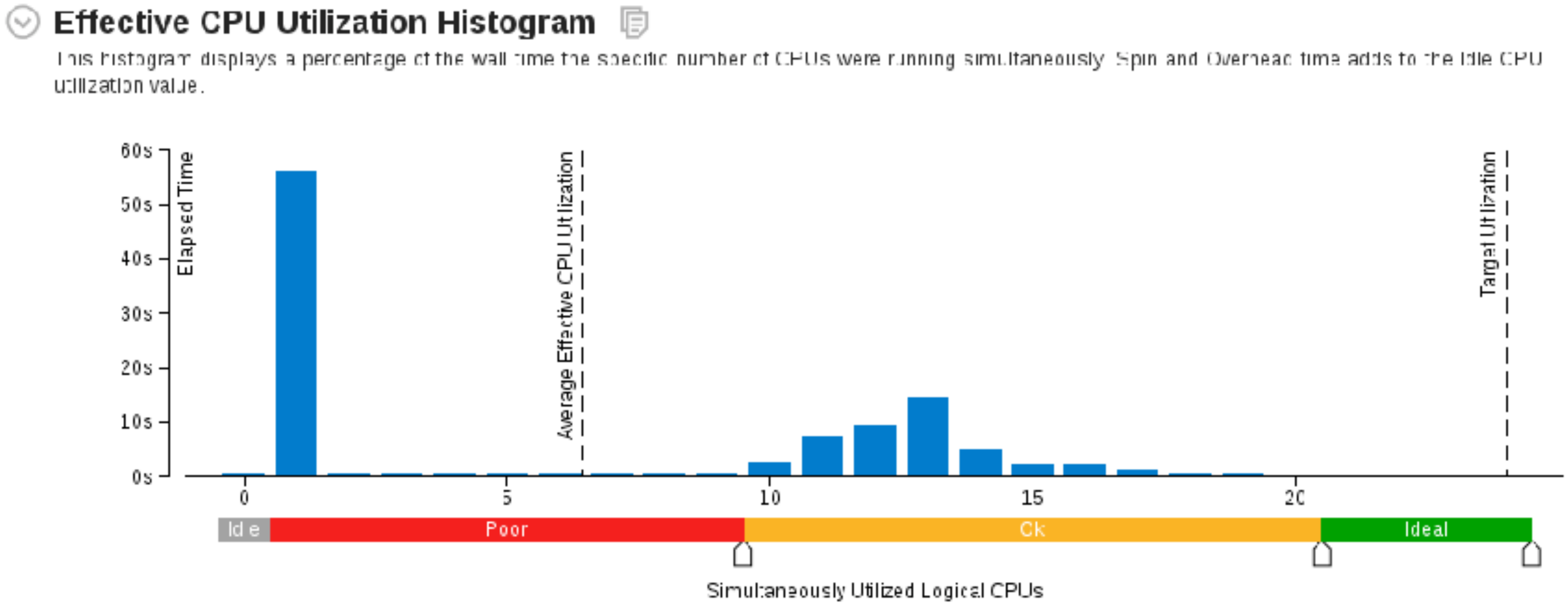
Total Thread Count: 24

Paused Time: 0s

Top Hotspots

This section lists the most active functions in your application. Optimizing these hotspot functions typically results in improving overall application performance.

Function	Module	CPU Time
func@0x18a90	libgomp.so.1	379.998s
func@0x18bf0	libgomp.so.1	126.738s
lapla_._omp_fn.0	run-omp	44.980s
lapla	run-omp	37.877s
erff	libm.so.6	16.699s
[Others]		62.966s



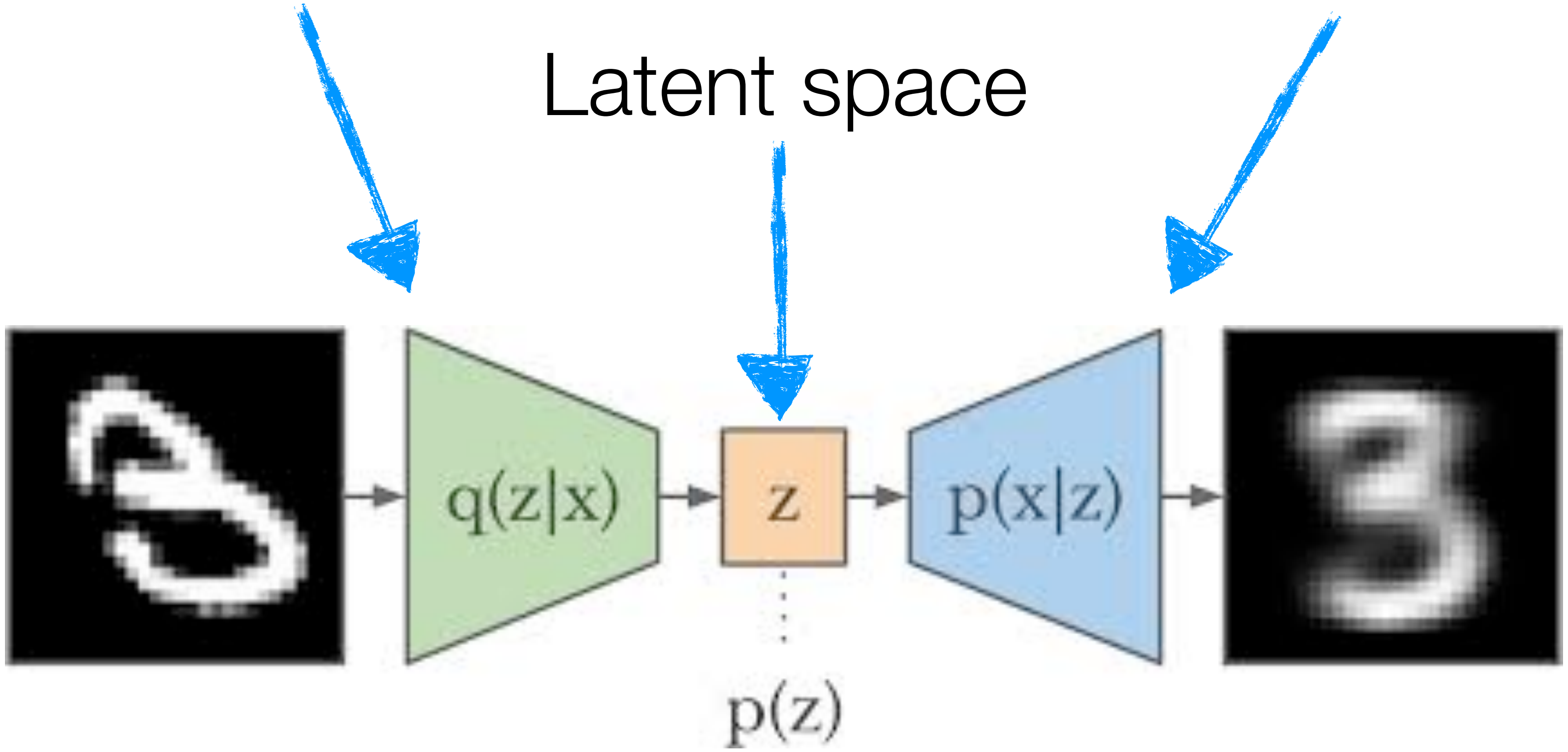
Effective CPU Utilization Histogram

This histogram displays a percentage of the wall time the specific number of CPUs were running simultaneously. Spin and Overhead time adds to the Idle CPU utilization value.

Using Variational Auto Encoder

- Despite the optimisation, BLOB is still too slow
- The idea:
    - Bin the PDF output of BLOB
    - Creating a 3D “image”
    - Train a Variational Auto Encoder (VAE) to reproduce such “images”
    - Condition the VAE to impact parameter

Variational Auto Encoders

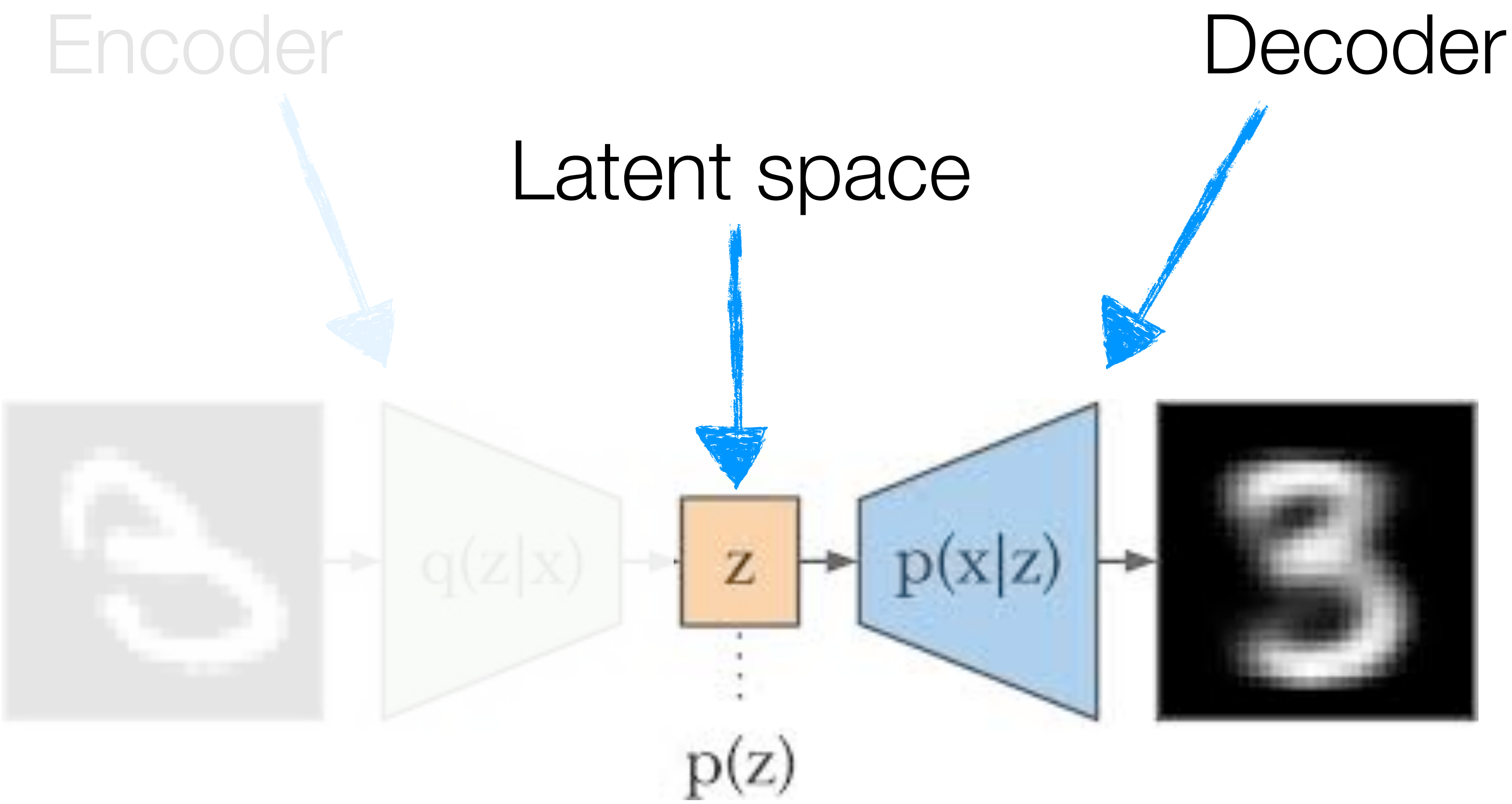

Encoder

Latent space

Decoder

- Train an identity function

Variational Auto Encoders



Encoder

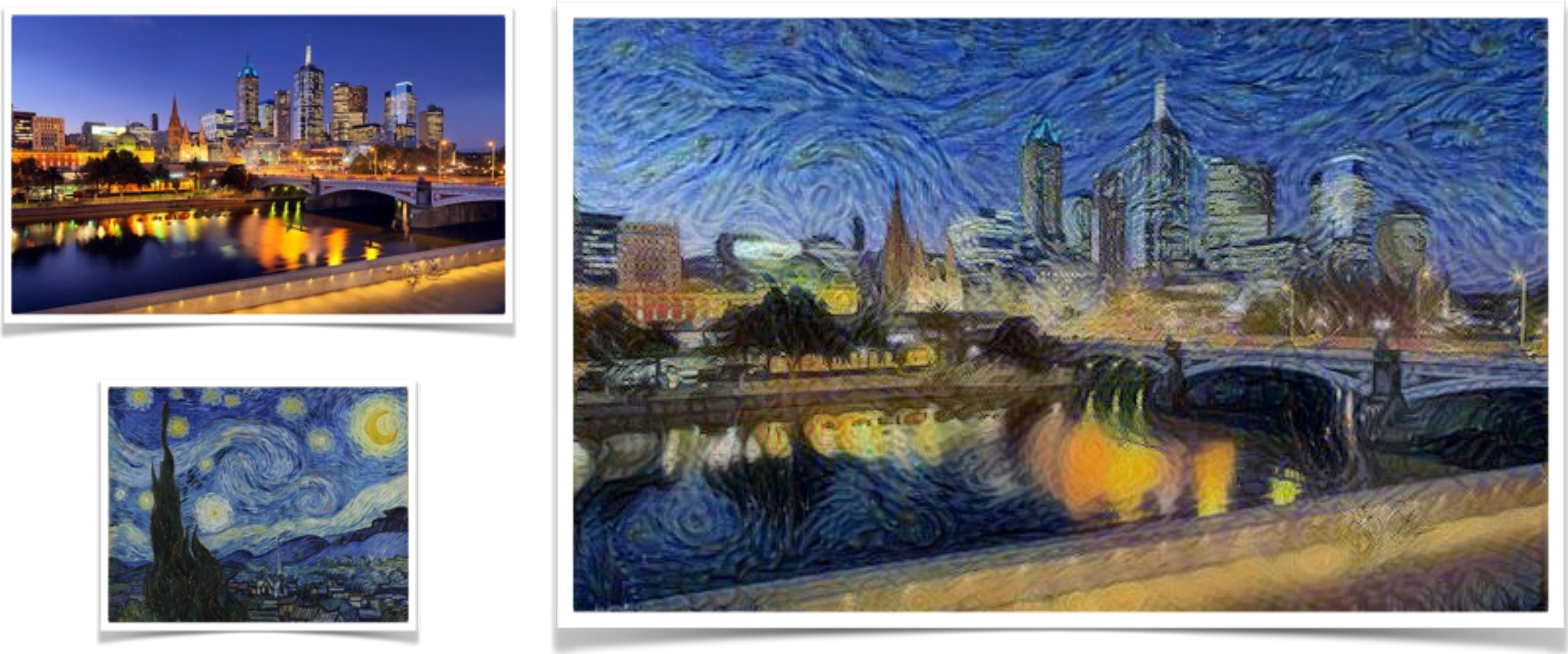
Latent space

Decoder

- Use the decoder to produce artificial images

Variational Auto Encoders

- Painting like Van Gogh…

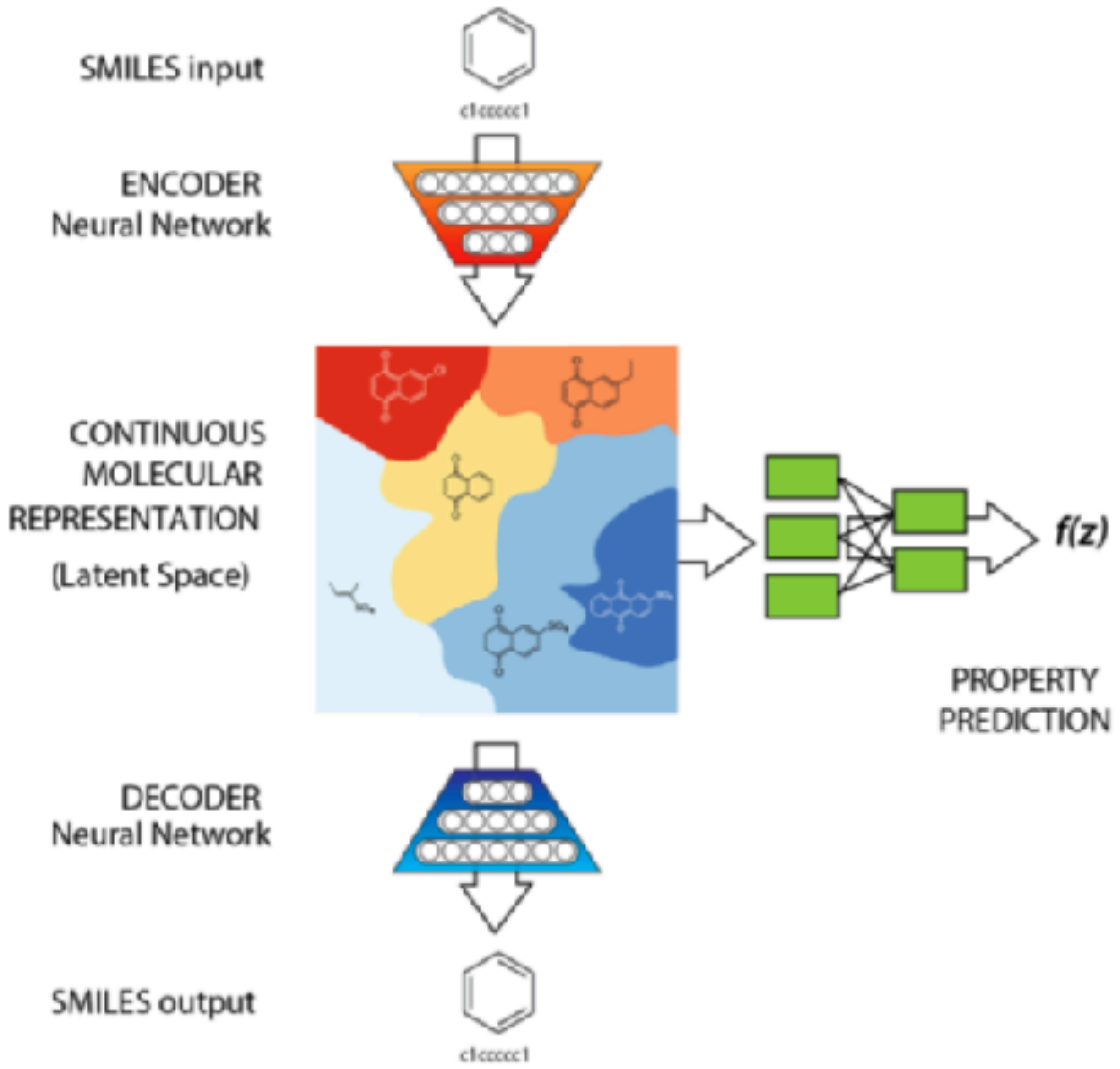


Conditioning to *b*

- Taking inspiration from:

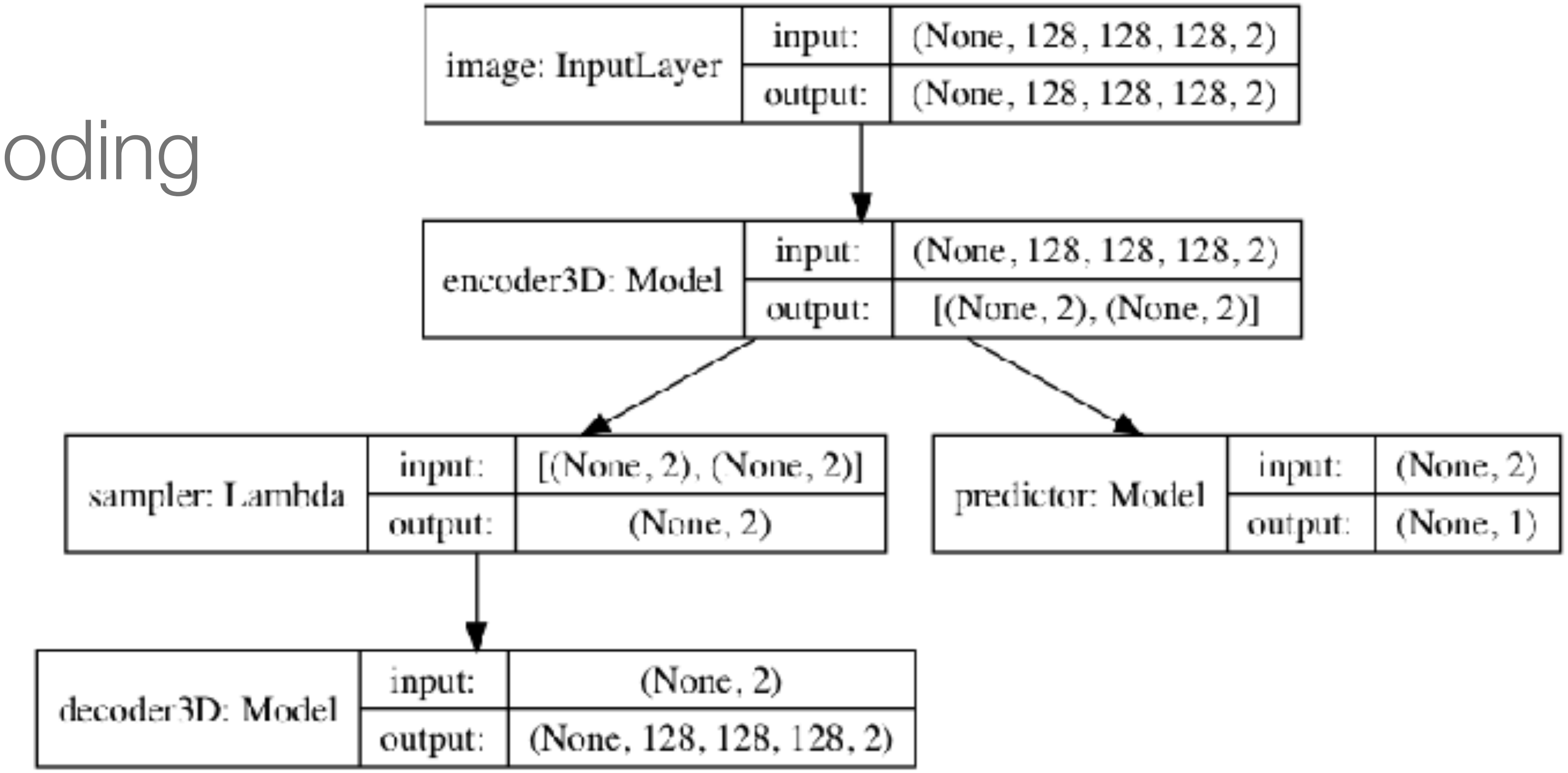
[Automatic chemical design using a data-driven continuous representation of molecules, Gómez-Bombarelli at al. arXiv:1610.02415]

- VAE for generating new chemical compounds with properties that are of interest for drug discovery
- To organize latent space w.r.t chemical properties they jointly trained the VAE with a predictor
- It predicts these properties from latent space representations



Conditional VAE

- Convolutional 3D encoding
- Conditioned latent space
- Symmetric decoding



Training dataset

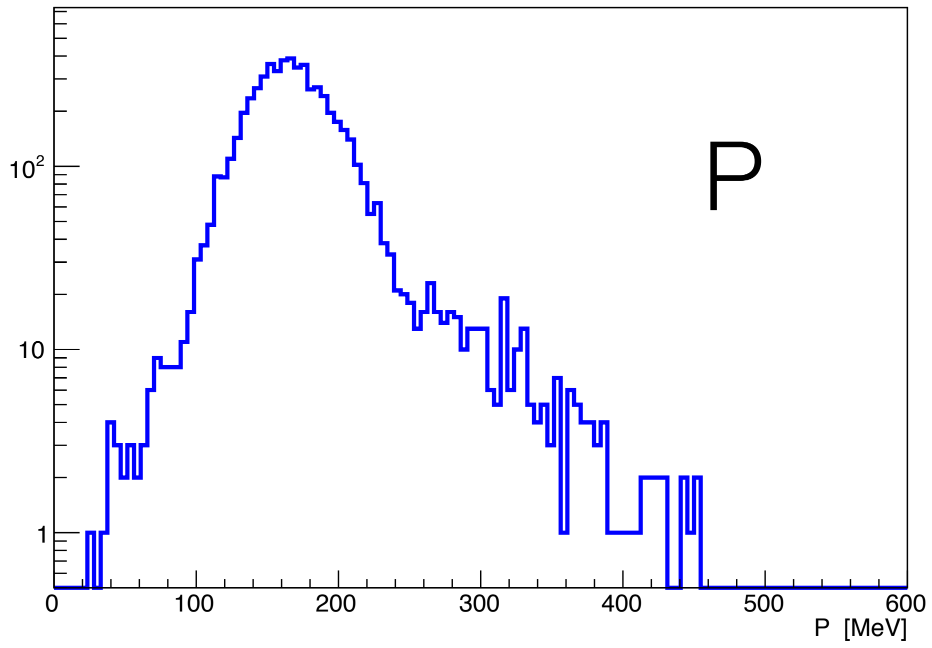
- The BLOB final state is a list with the position in the phase space of fragments and gas particles
- Fragments: A and Z (real), P, Q and Excitation energy
- Gas particles: Z, P and Q. Each represent a 1/500 probability of having a nucleon in that position of phase space
- 1 000 events
- Generated with linear impact parameter
- 90% for training and 10% of them for test

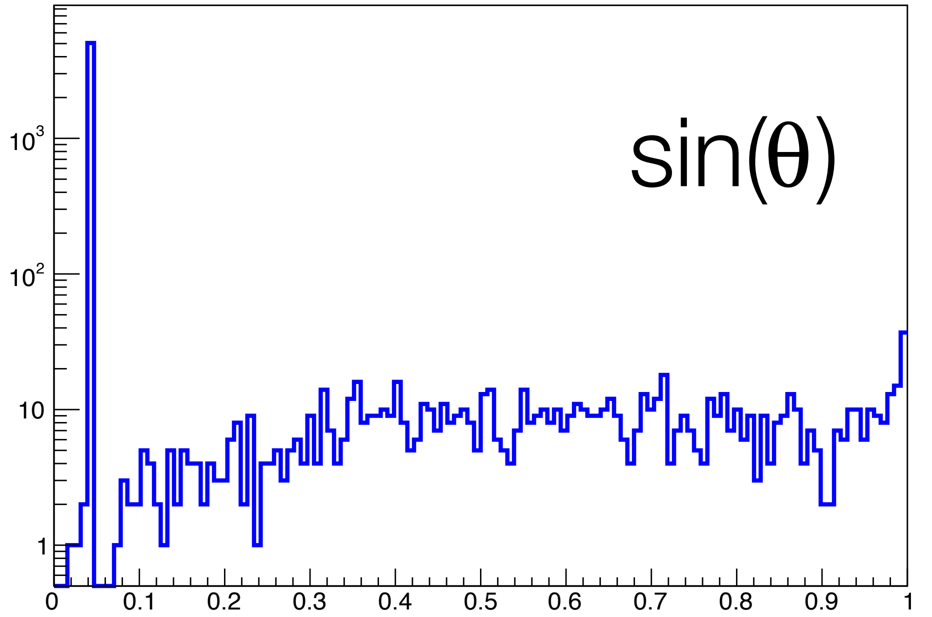
Reducing dimensionality

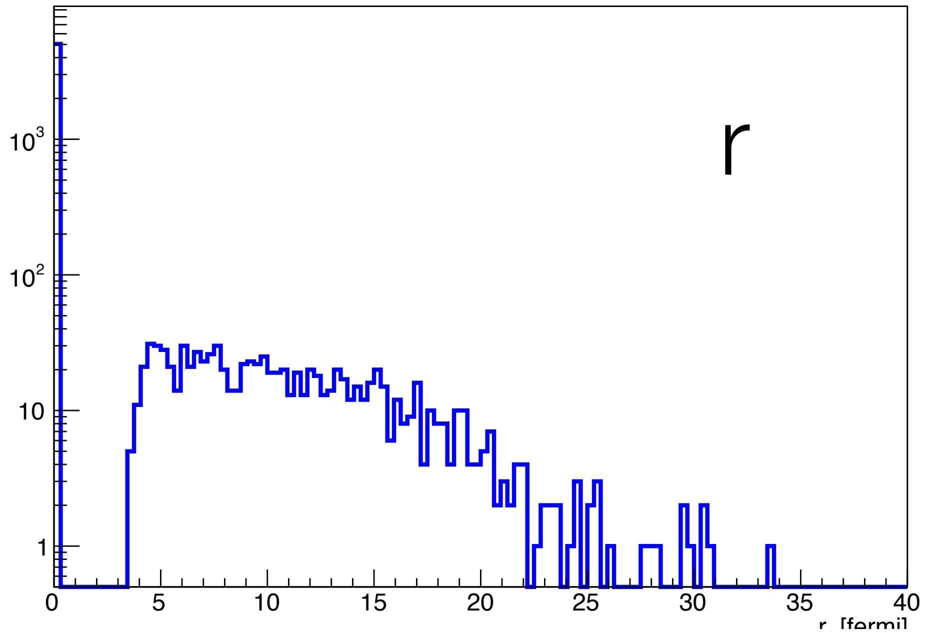
- Only events with 2 fragments are considered

- We divided the test particles in two samples:
    - Projectile like (red)
    - Target like (blue)

- sin(θ) instead of θ to:
    - have same sign
    - enhance small angles

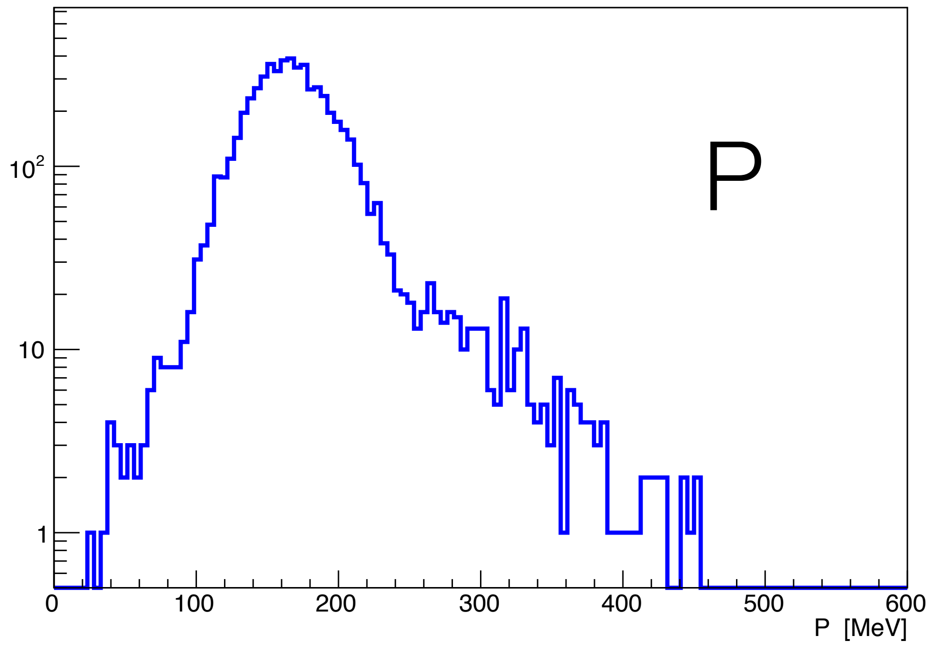


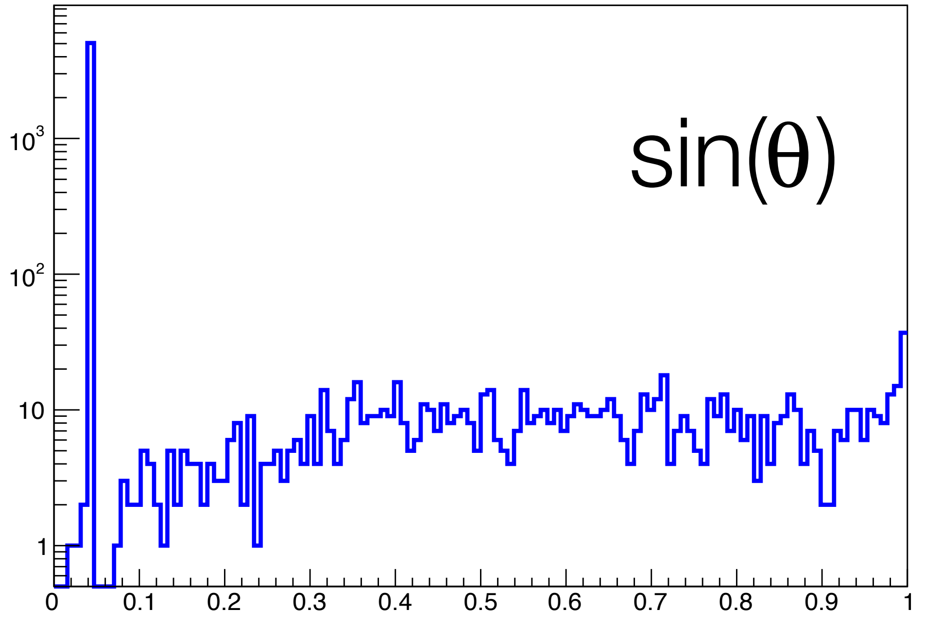


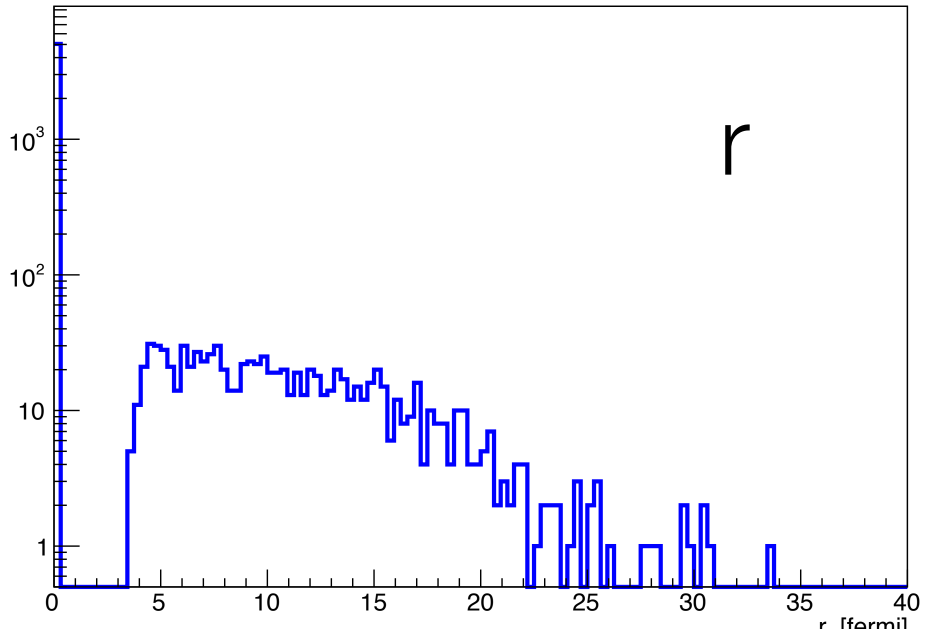


Reducing dimensionality

- To reduce the dimensionality and use the keras 3D kernels
- We consider only:
  - The modulus of the momentum
  - its angle with the collision axis
  - The distance of each test particle with the fragment center

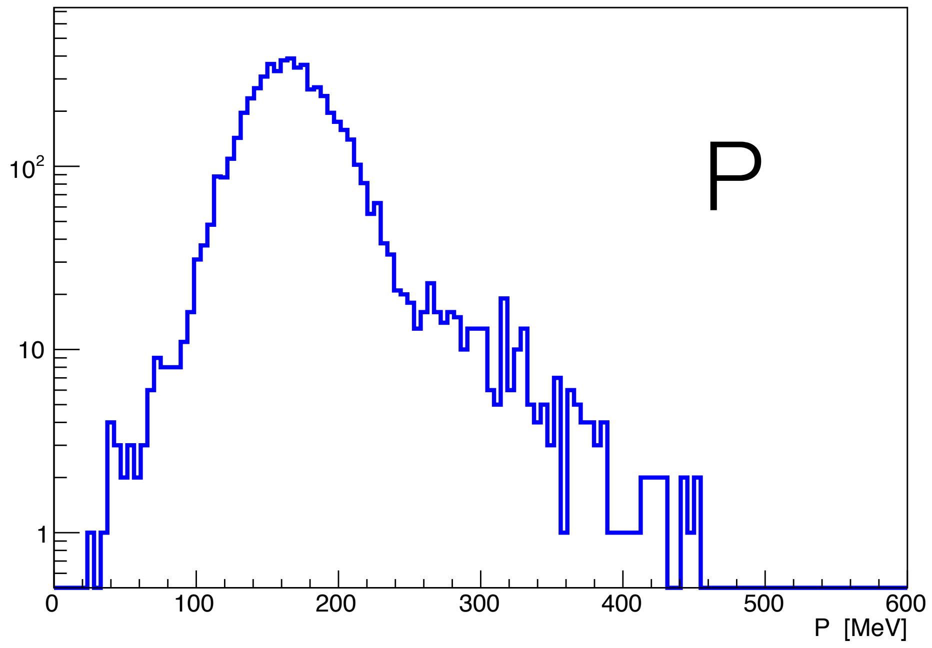


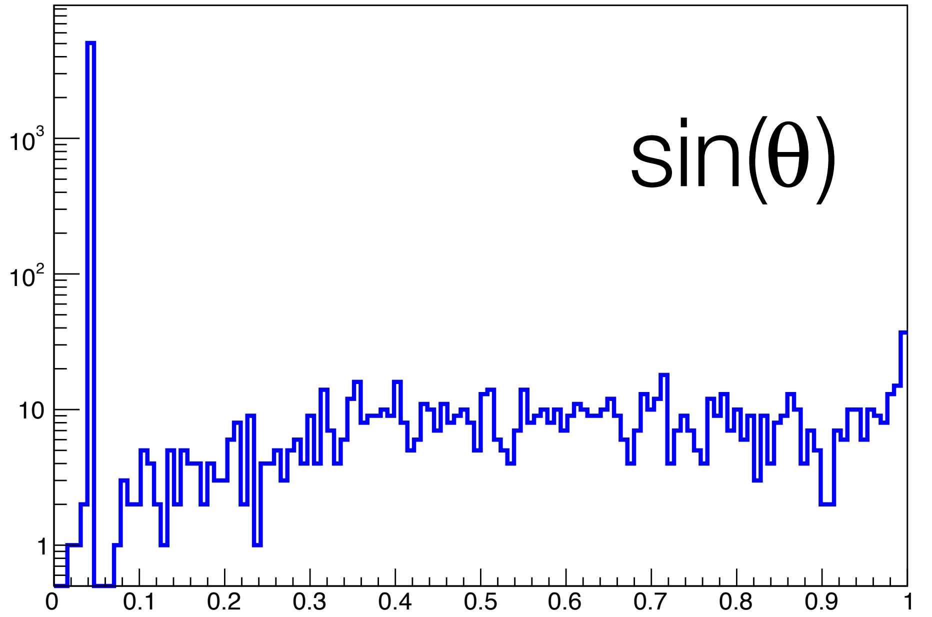


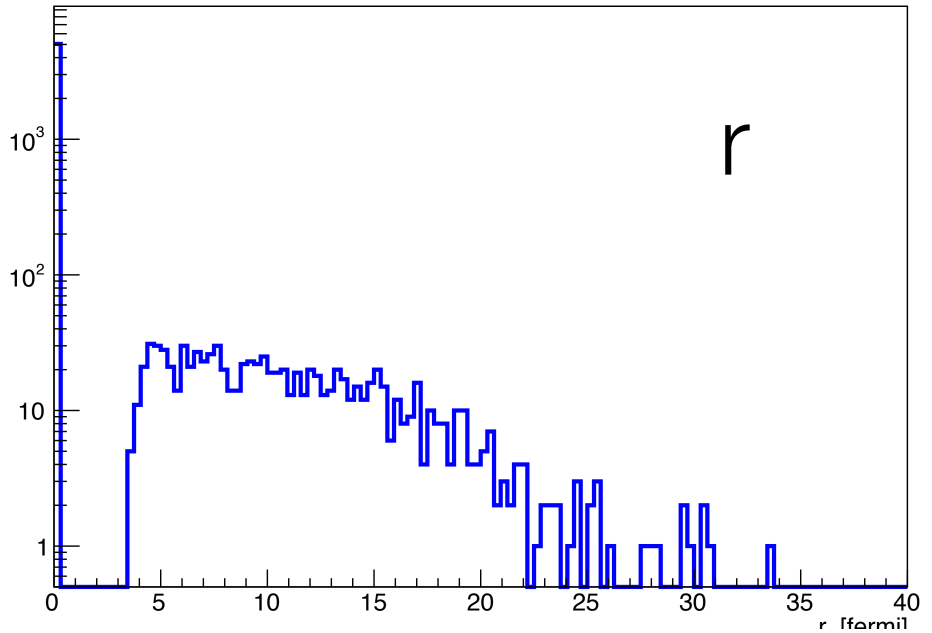


Reducing dimensionality

- Fragments are represented by 500*A particles
- P is sampled with gaussian distribution:
    - mean = Pfrag
    - sigma = Excitation energy
- All with the same θ
- r = 0

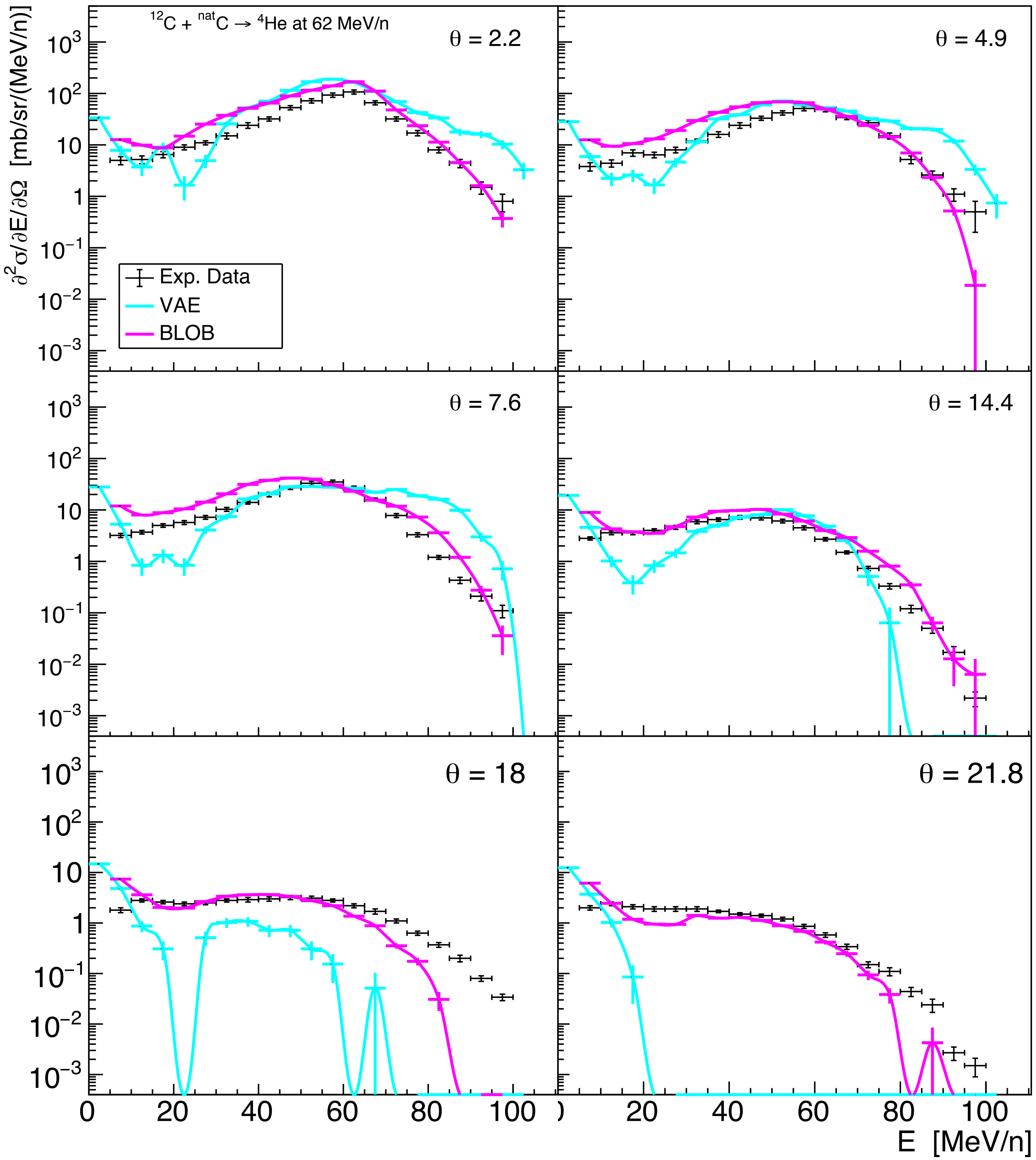






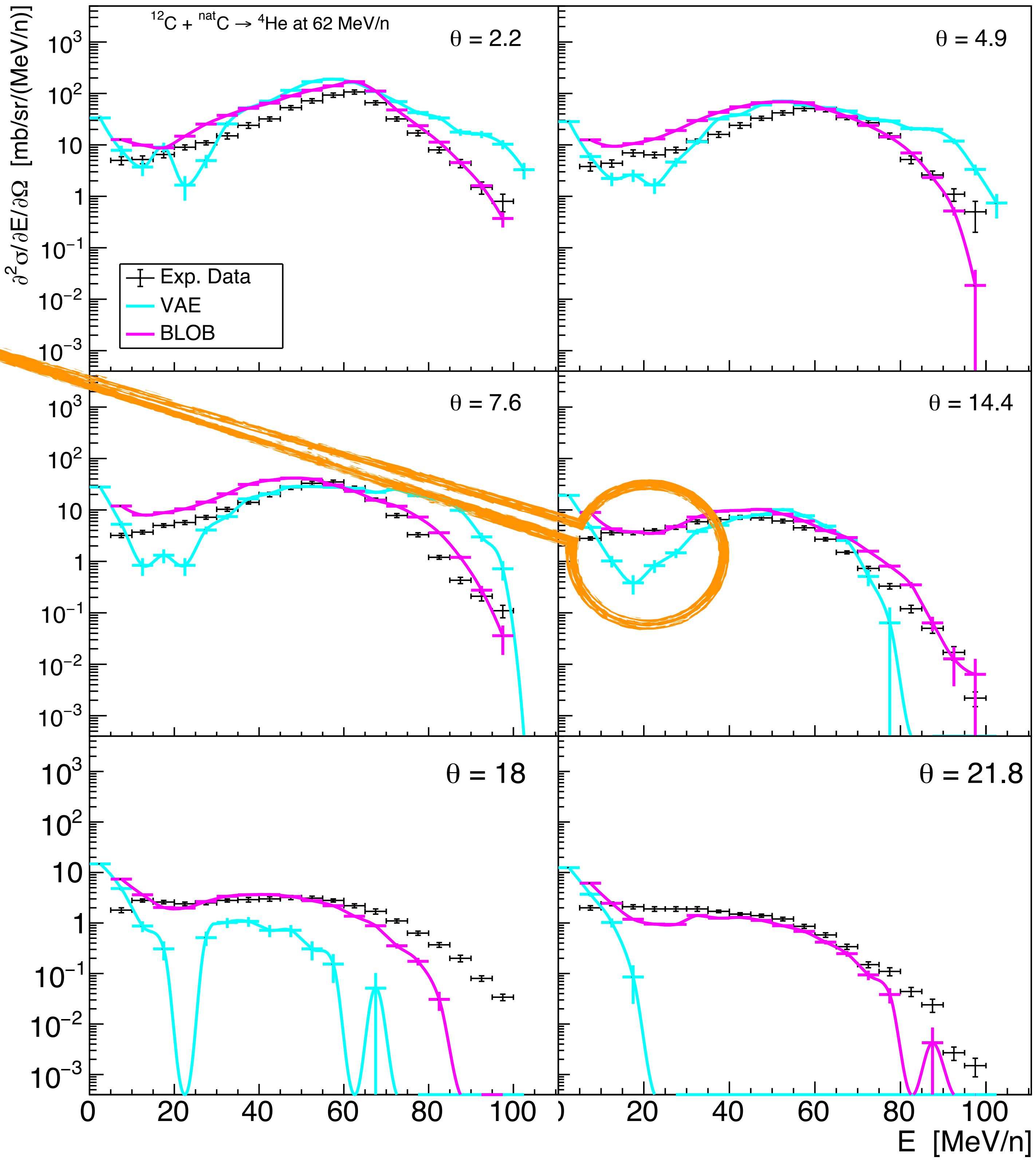
Testing reconstruction

- Fragments are identified selecting r<1fermi
- Momentum = average
- Excitation energy = variance
- θ = average



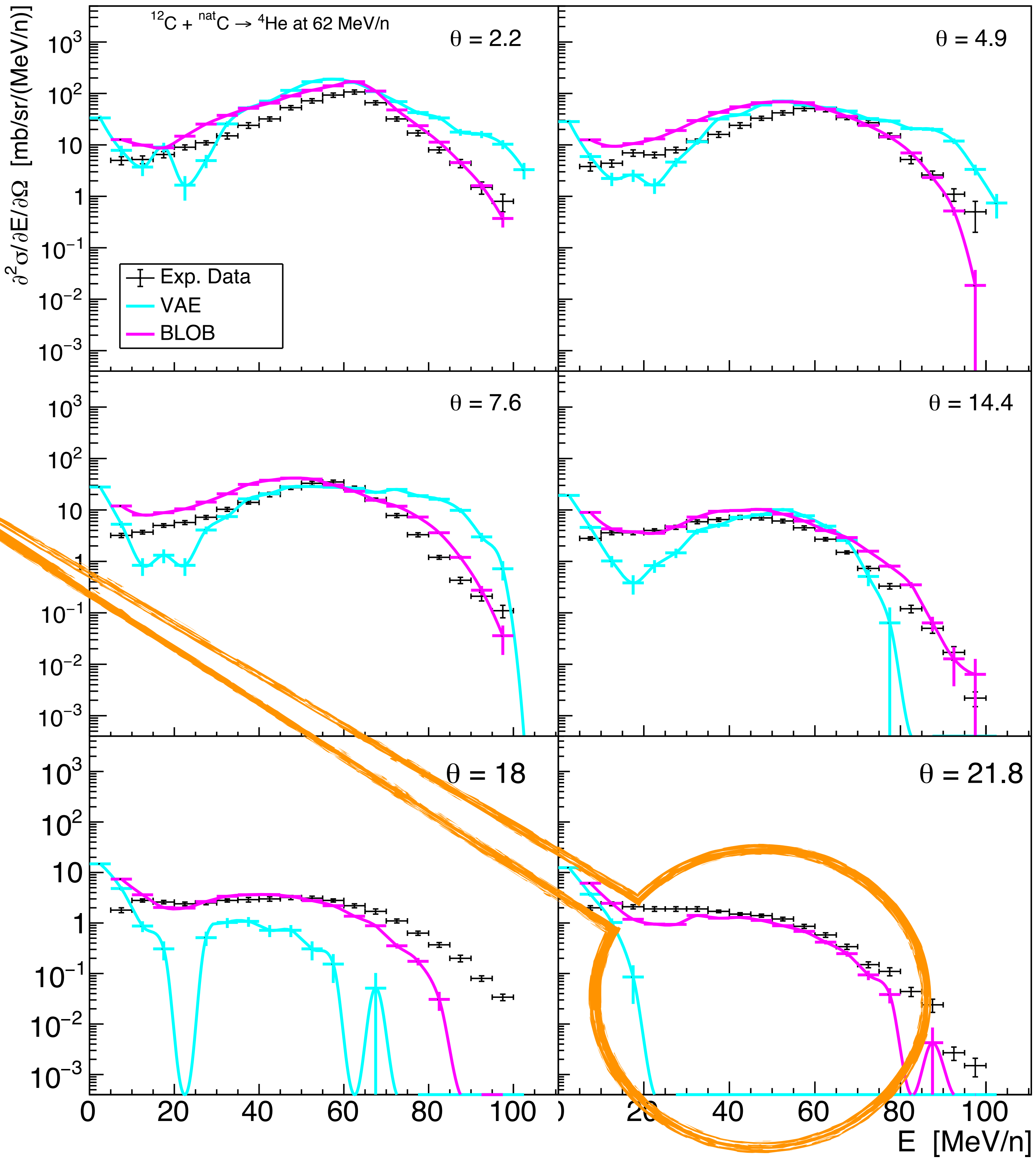
Testing reconstruction

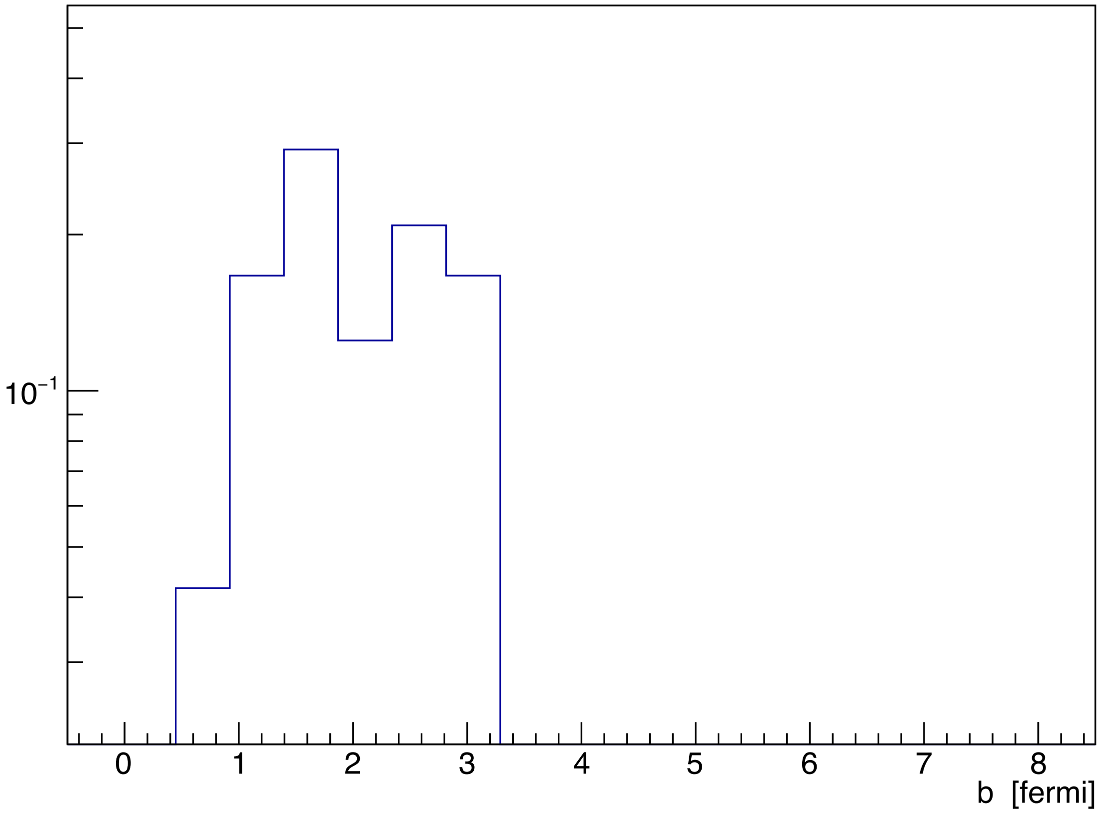
- Lack of particles at mid rapidity
- Underestimation of neck events
- Because of coalescence not active



Testing reconstruction

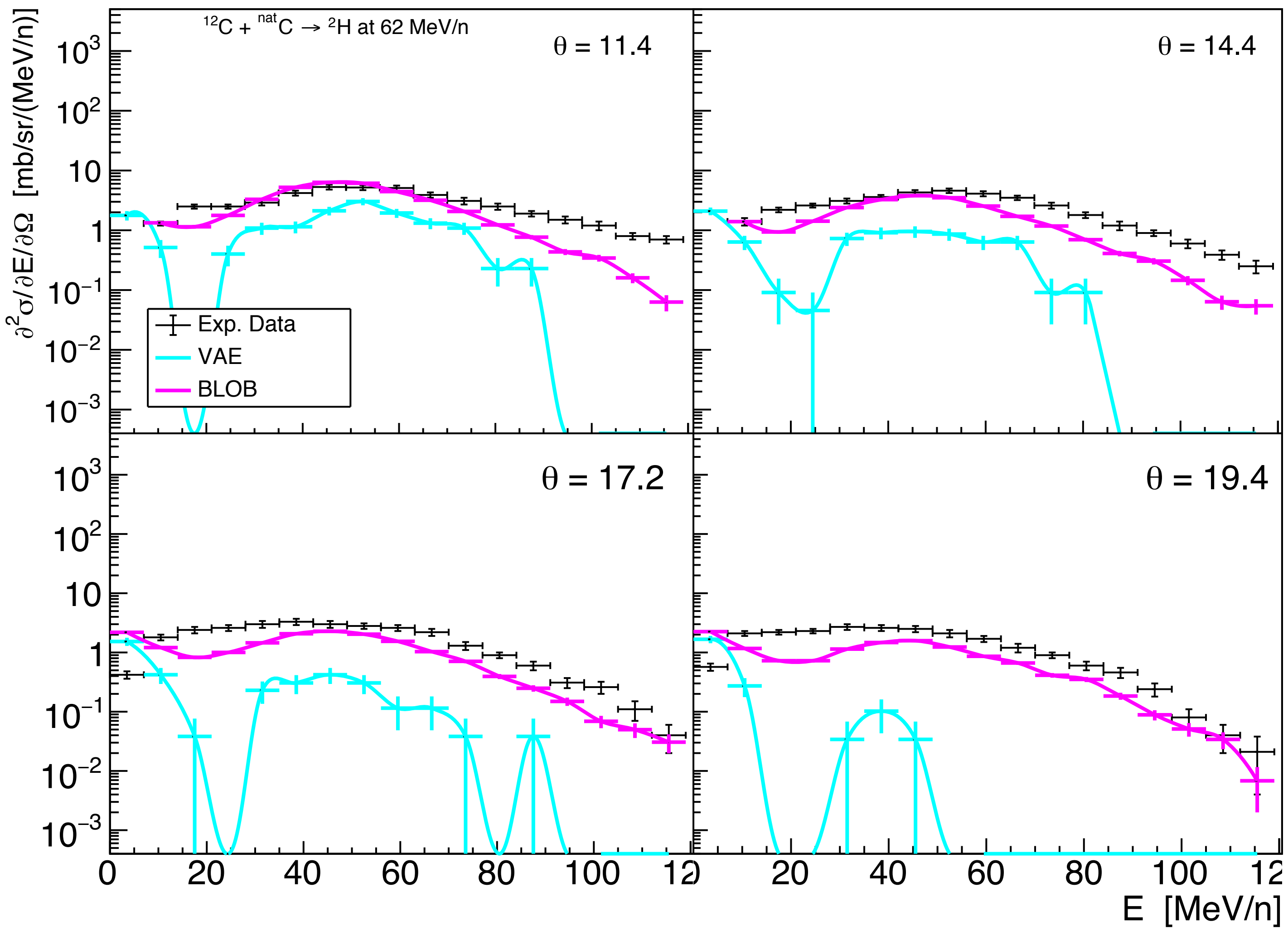
- Lack of particles at large angle
- Because of the request of 2 fragments





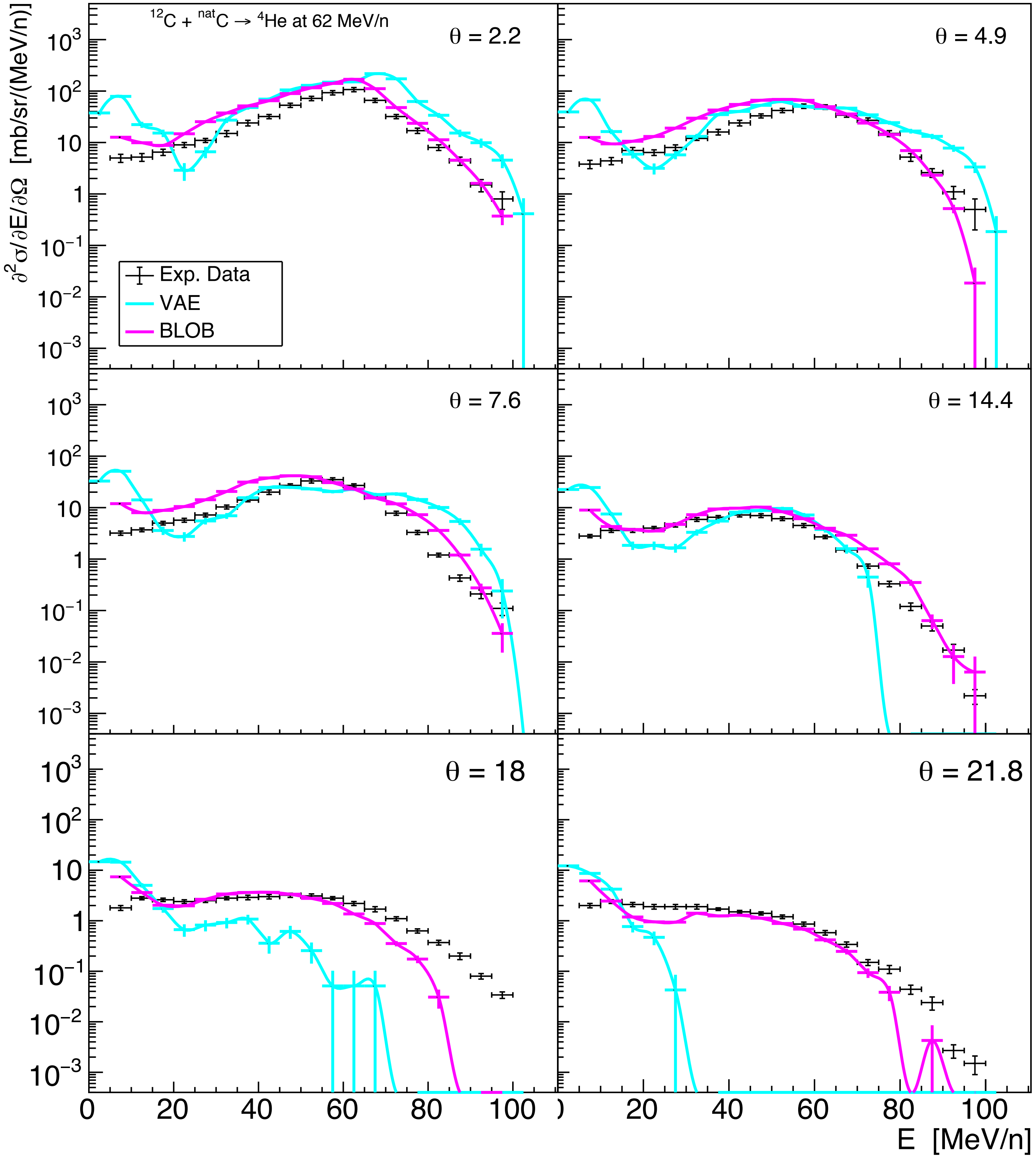
Testing reconstruction

- Same on deuterium
- Next step will be add a 3rd channel for neck particles
- And clustering algorithm



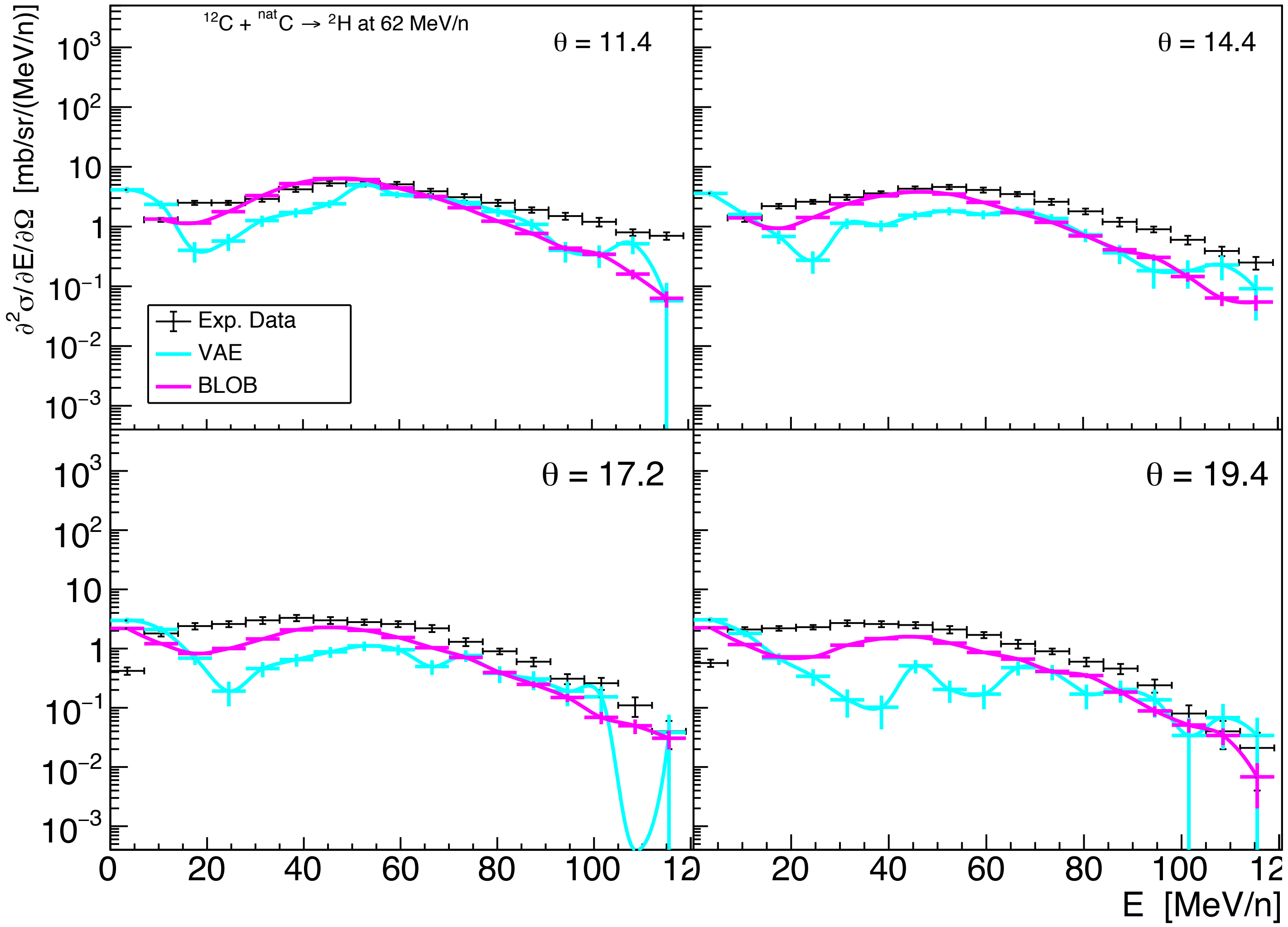
With coalescence

- Using the coalescence
- The mid rapidity lack is mitigated



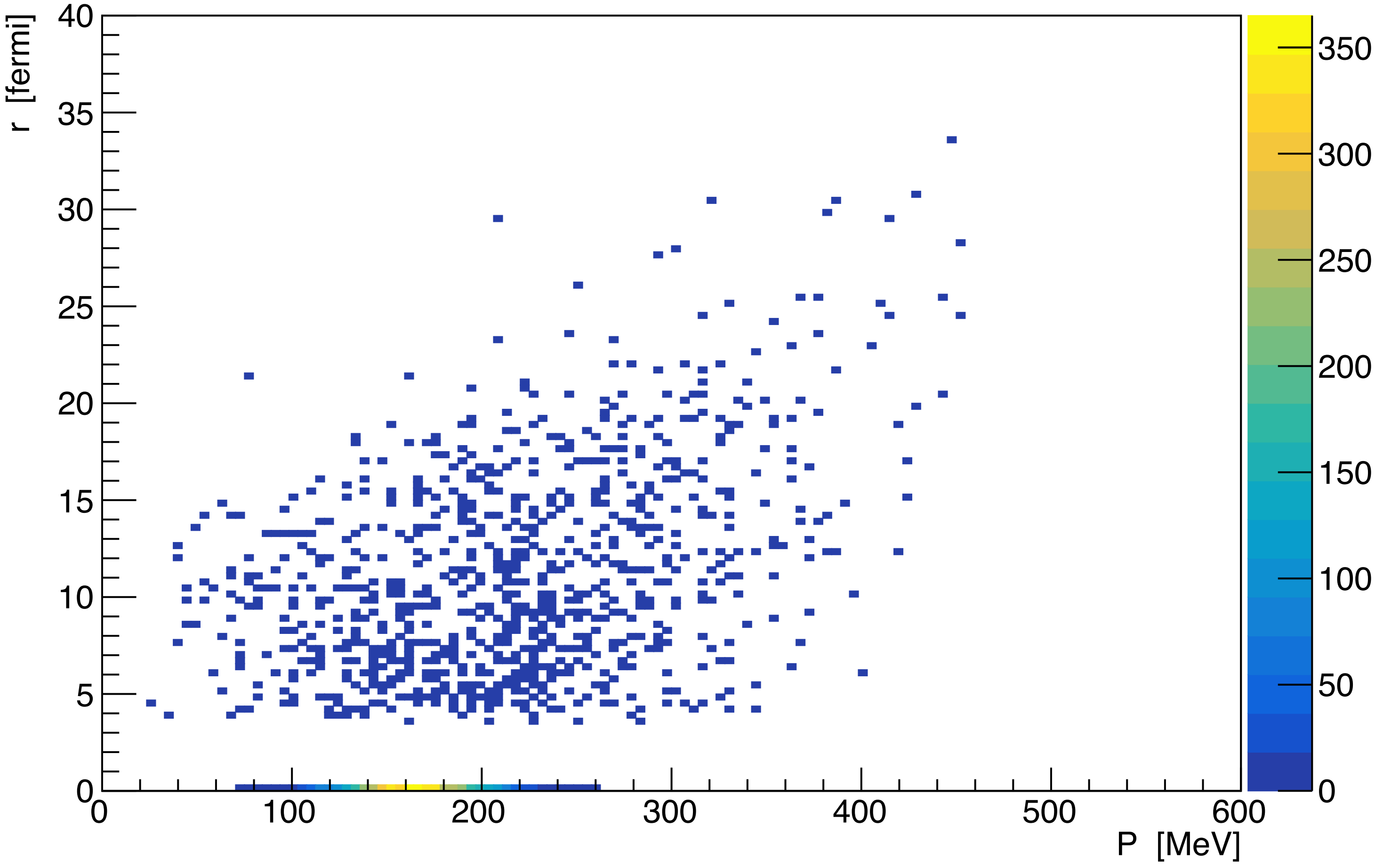
With coalescence

- Also for deuterium



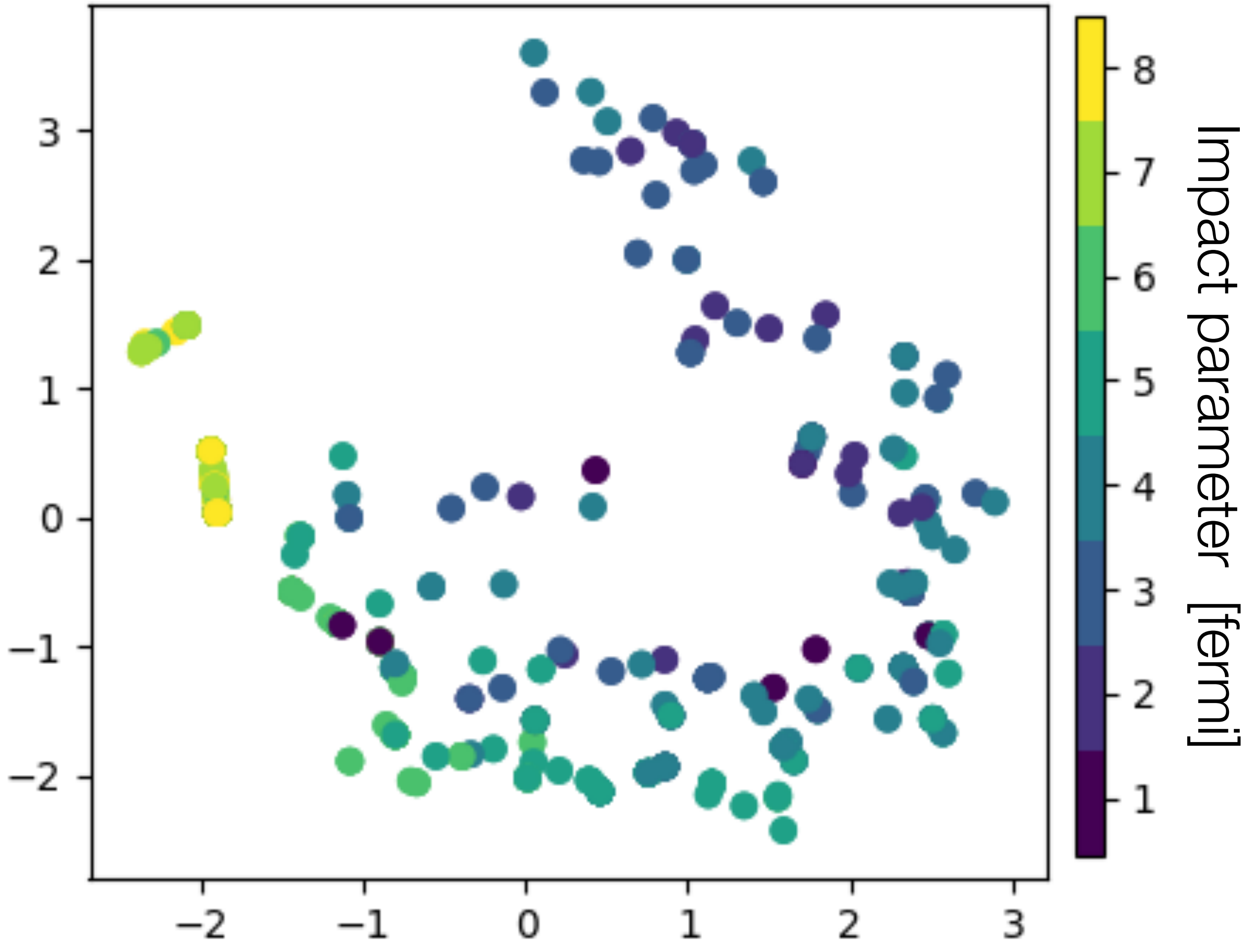
Challenges

- Sparse
- Large input (128^3 numbers)
- Small dataset (for the moment)
- Impact parameter distribution non uniform (for the moment)



Latent space

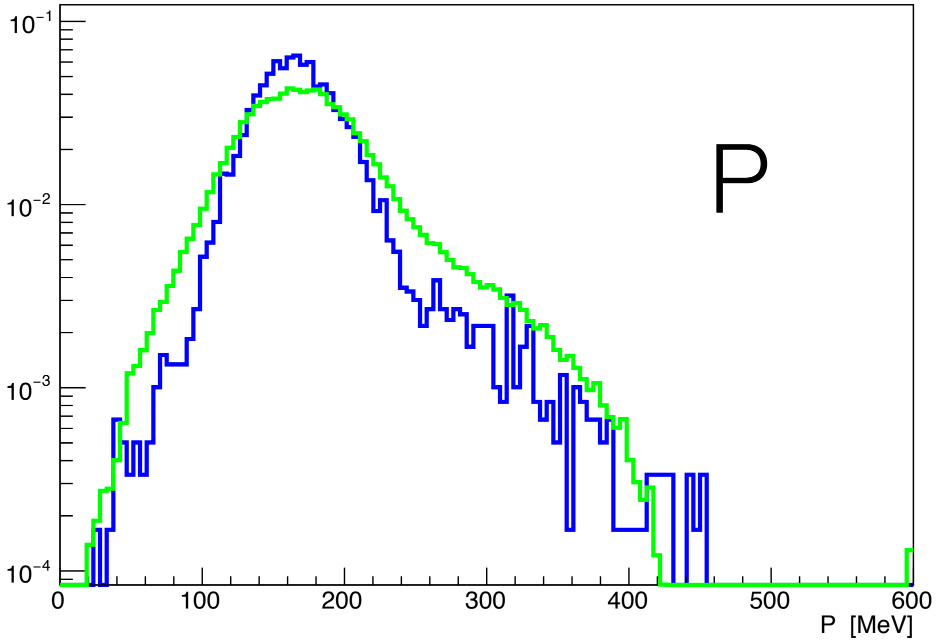
- 40 epochs of training
- Events with similar impact parameters are close in latent space
- Especially the events with very large impact parameters
- *b* is linearly distributed (a uniform distribution is needed)

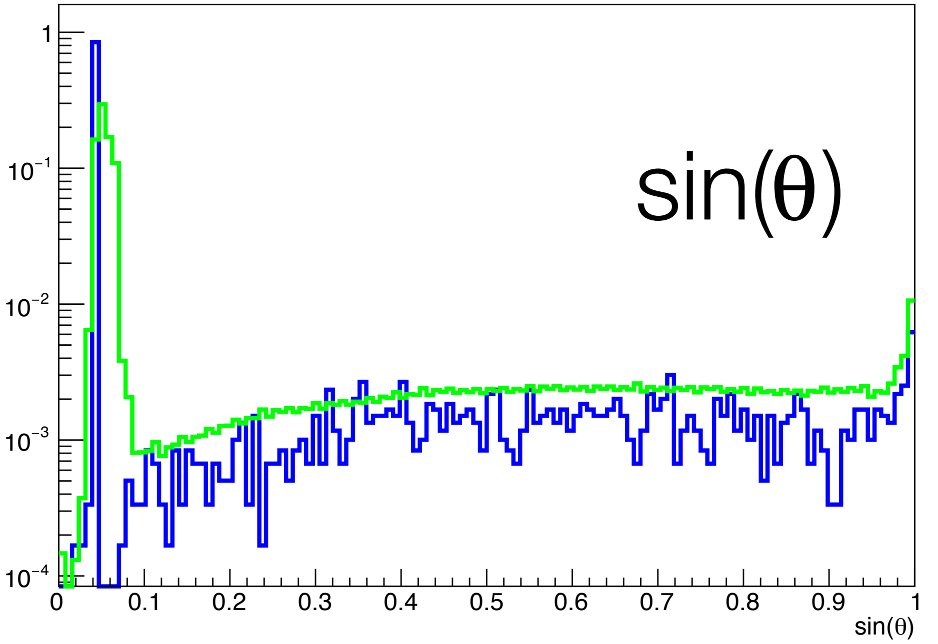


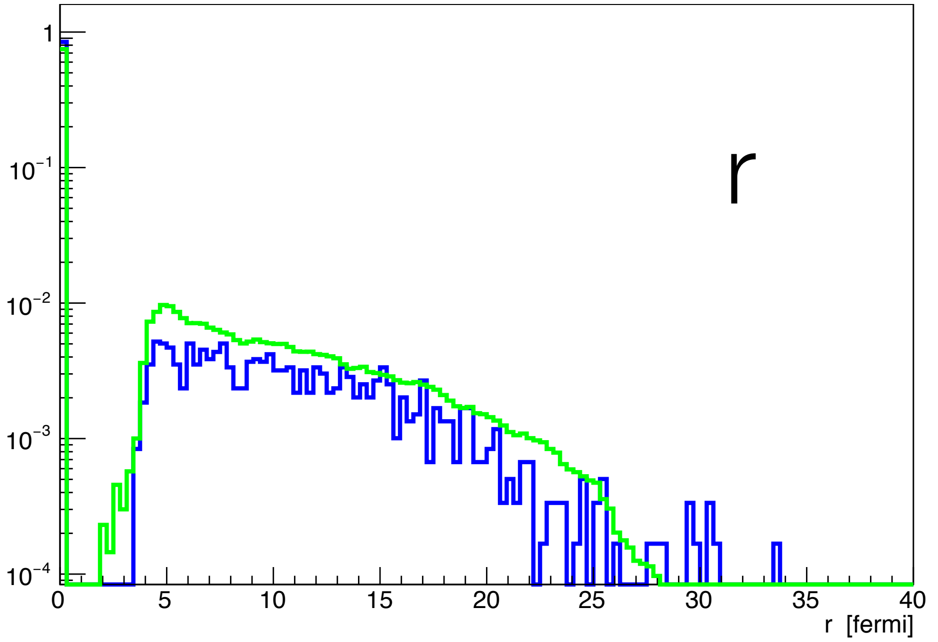
VAE output

Output distributions

- The generated distributions (green) looks similar to the input (blue)
- The generated event has been generated sampling two gaussian in latent space with:
  - means = position of the input
  - sigmas = 0.1







Summary

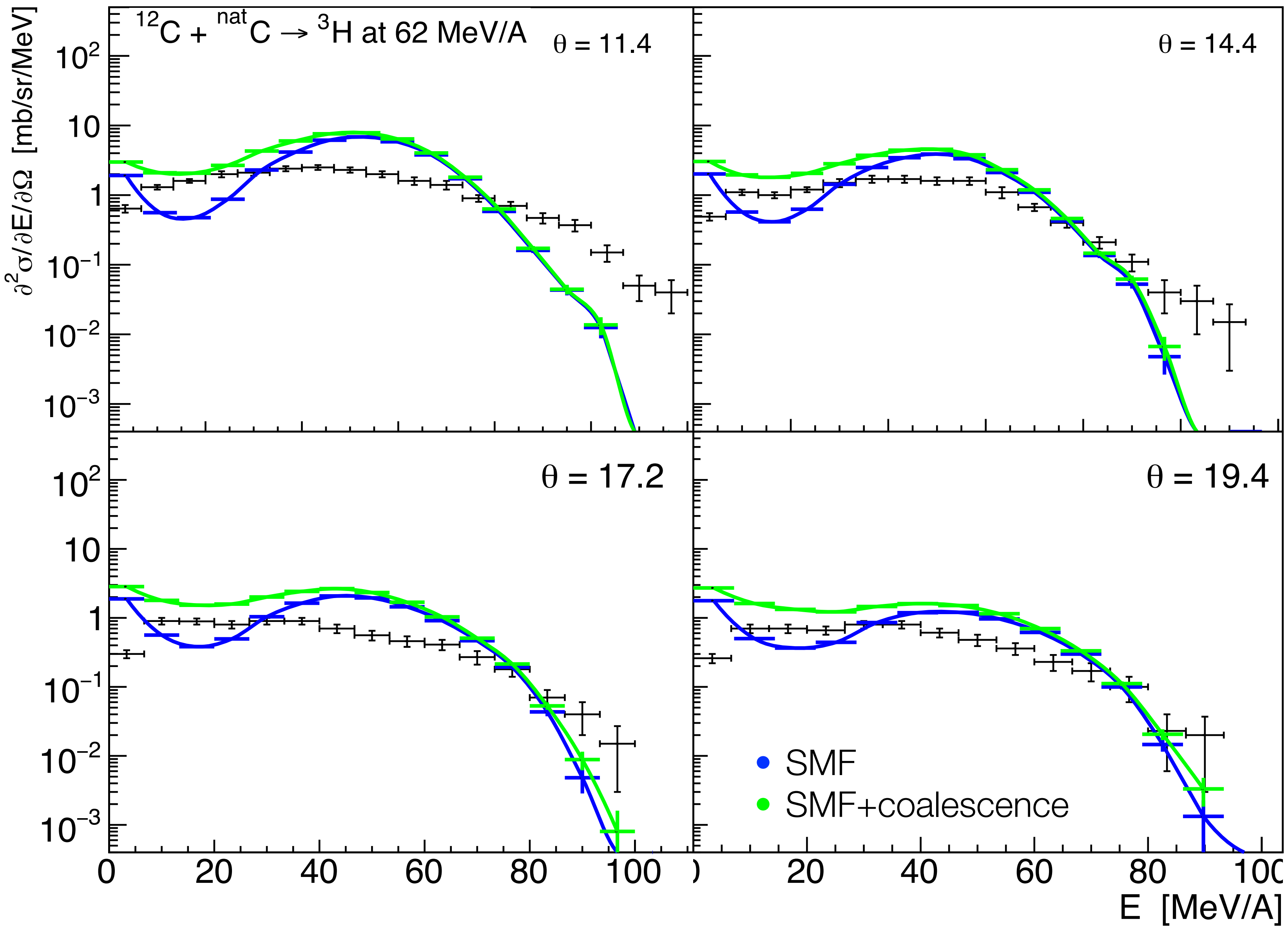
- An interface between BLOB/SMF and Geant4 has been developed
    - Samples the final state (fragments and nucleons emitted)
    - Include a coalescence
    - Corrects excitation energy for large *b*
    - Corrects the excitation energy of stable fragments
- The agreement with doubly differential data is good for light fragments
- Further optimisations are needed for larger fragments

carlo.mancini.terracciano@roma1.infn.it

Backup

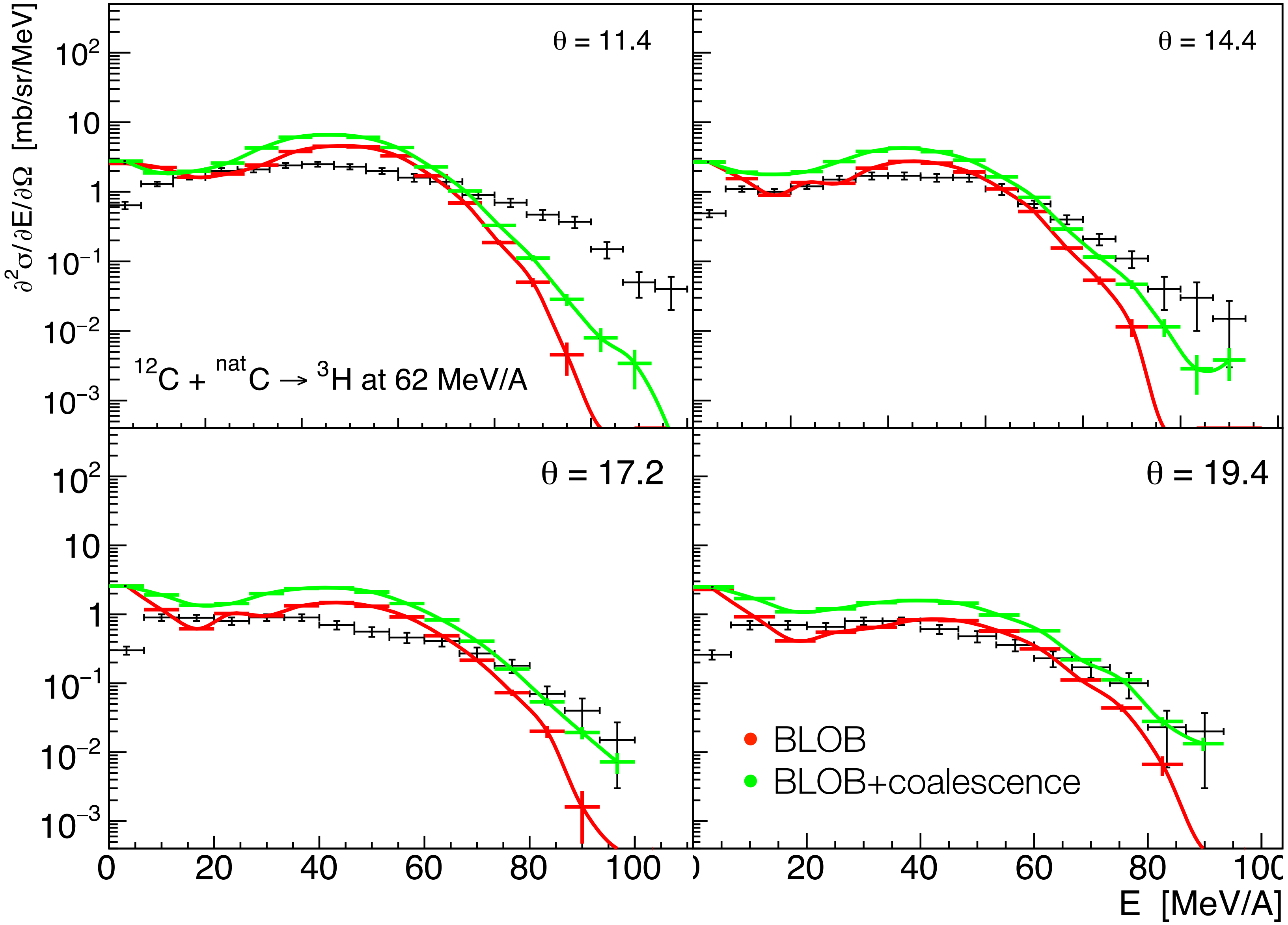
Coalescence

- Mitigates the gap between the fragments produced by projectile-like and target-like



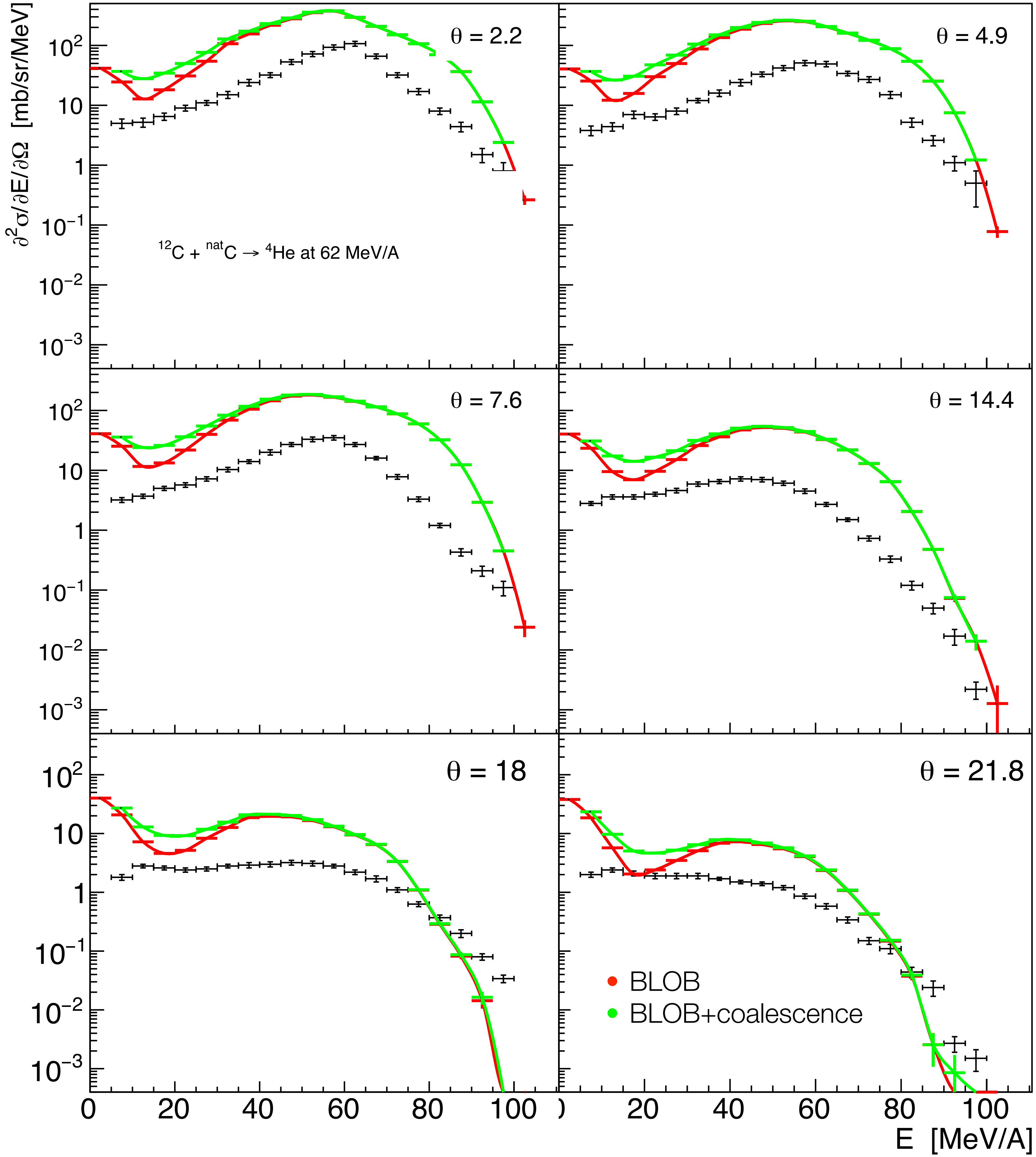
Coalescence

- Similar results with BLOB
- Produces high energy tritium



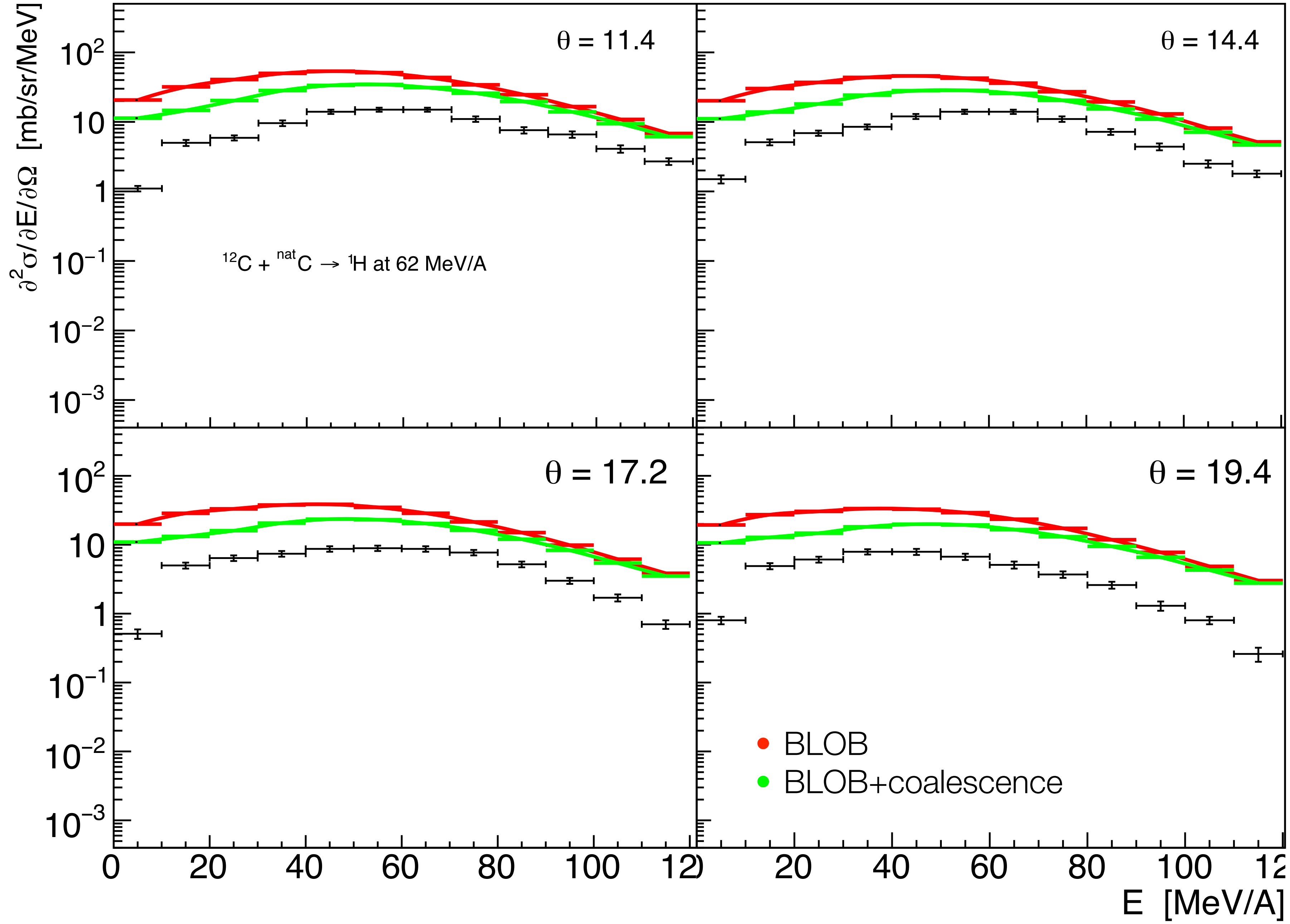
Coalescence

- Mitigates the gap between projectile and target fragments



Coalescence

- Reduces the excess of proton



Why GPUs

- BLOB and SMF explore the time evolution of the density distribution with test particles
- 100 to 1000 test particles per nucleon

- Many test particles
- All the same

- At each step the mean field potential is calculated
- The test particle can interact only with elastic scattering

- Only one possible interaction

Low thread divergency