Deep Generative models for next-gen heavy-ion collision experiments

Manjunath Omana Kuttan, Kai Zhou, Jan Steinheimer, Horst Stöcker

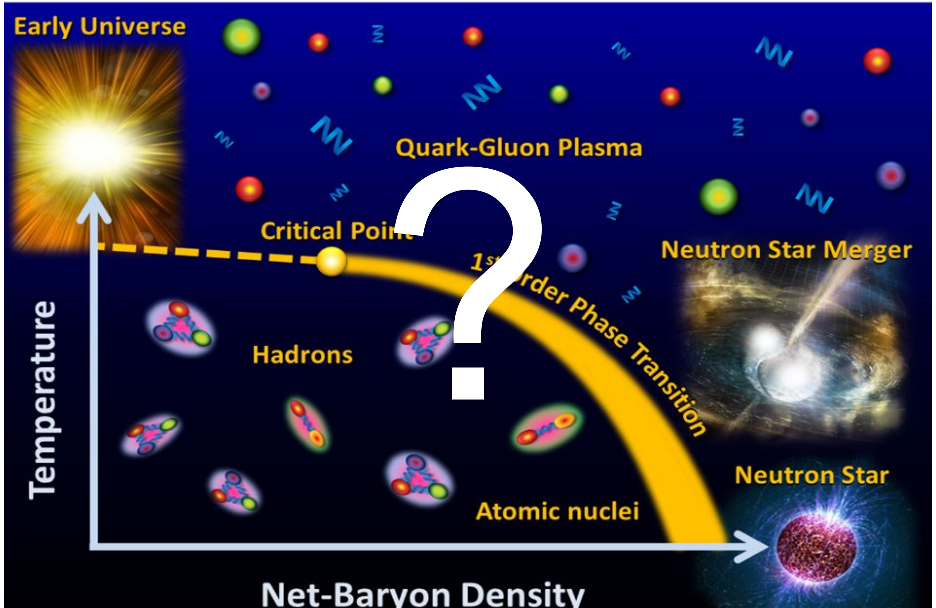
arXiv:2412.10352 , *arXiv:2502.16330*

FIAS Frankfurt Institute for Advanced Studies

KISS

The QCD phase diagram and heavy-ion reactions

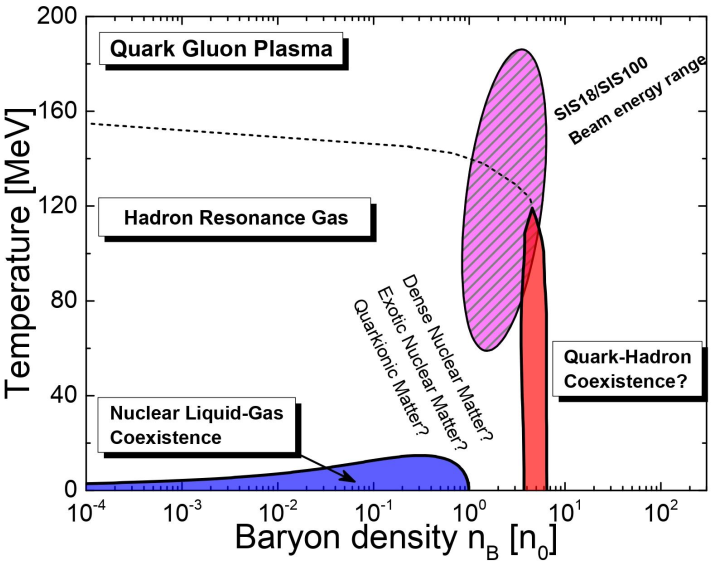
⚠️ *The phase structure of QCD is largely conjectured*



Particles 3.2 (2020): 320-335

The QCD phase diagram and heavy-ion reactions

⚠️ *The phase structure of QCD is largely conjectured*



- Moderate energy collisions -> QCD at high baryon density
  - Chiral and deconfinement transition ?
  - QCD critical point ?
  - Neutron star core/ merger densities?
- First principle calculations are not possible!

Comprehensive model calculations are necessary!

Decoding the phase structure

What models provide:

Physics of interest:
EoS, initial conditions, etc.

→

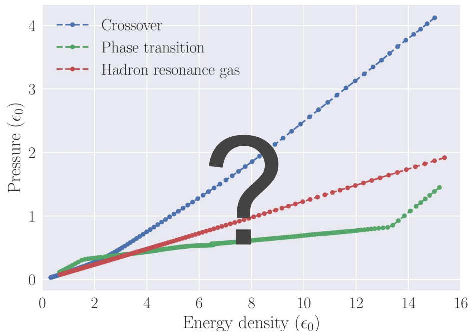
Model simulations

transport/ hydrodynamics/ hybrid etc.

→

expected data /
sensitive observables

The ultimate goal: Inferring the cause from effect!

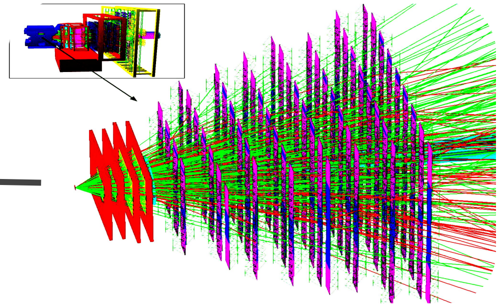


←

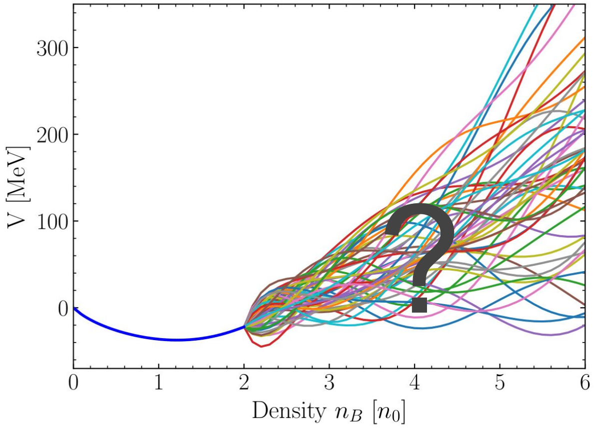
Model simulations

transport/ hydrodynamics/ hybrid etc.

←

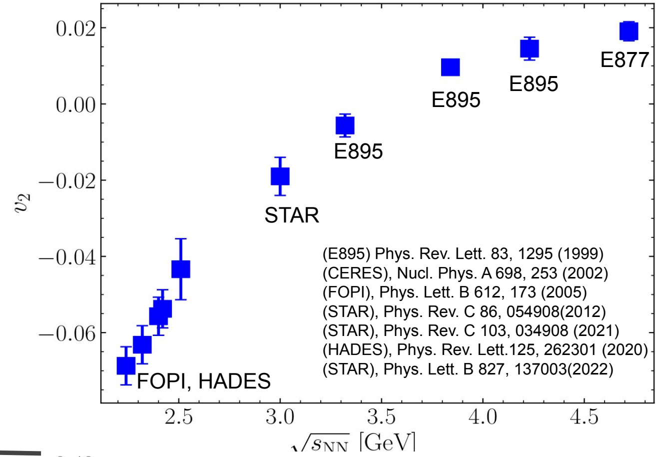


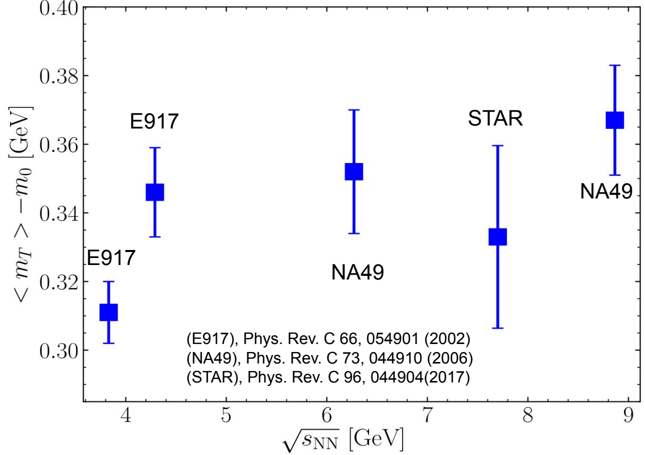
An example: extracting the QCD EoS



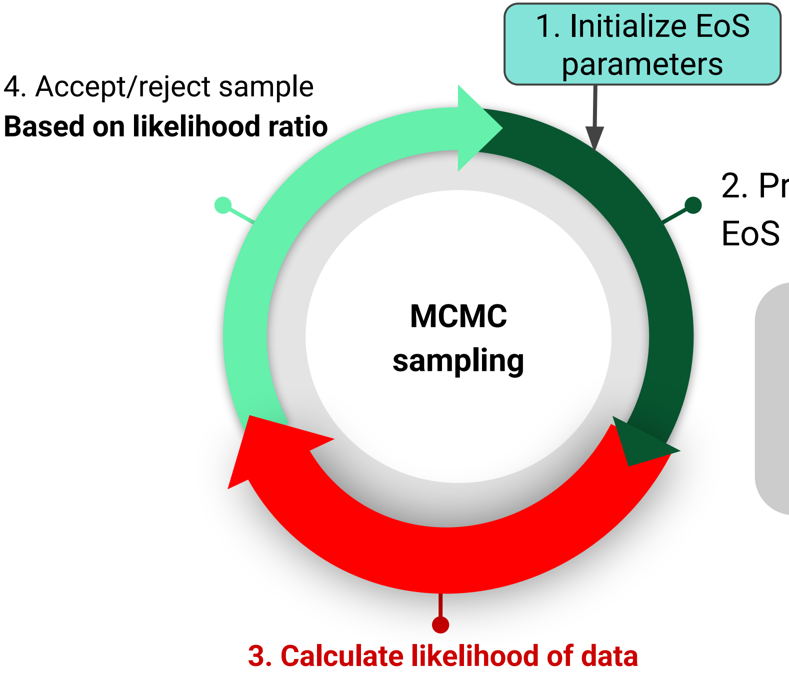
Potentials generated using the polynomial parameterization

UrQMD simulations





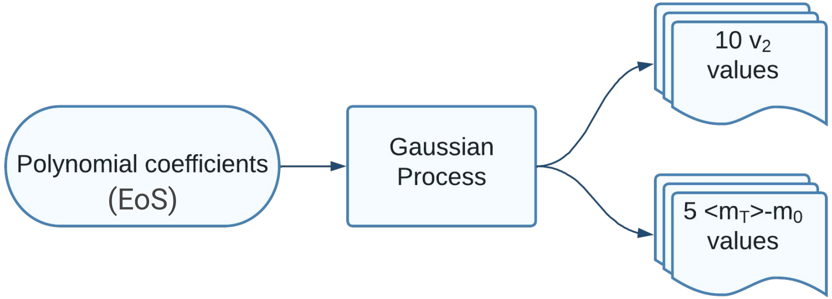
Bayes to the rescue!



- **Expensive!**
- UrQMD ~ 80 s/event
- Each EoS~125K events* 80 s/event =**~2700 hrs**

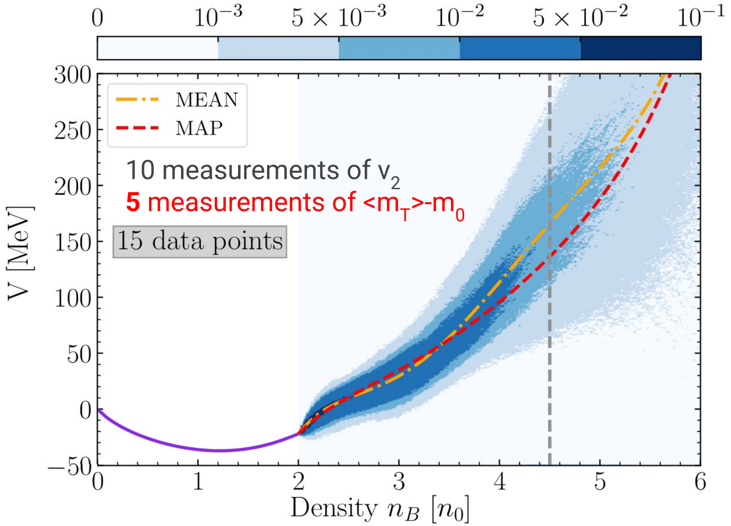
Gaussian Process emulator

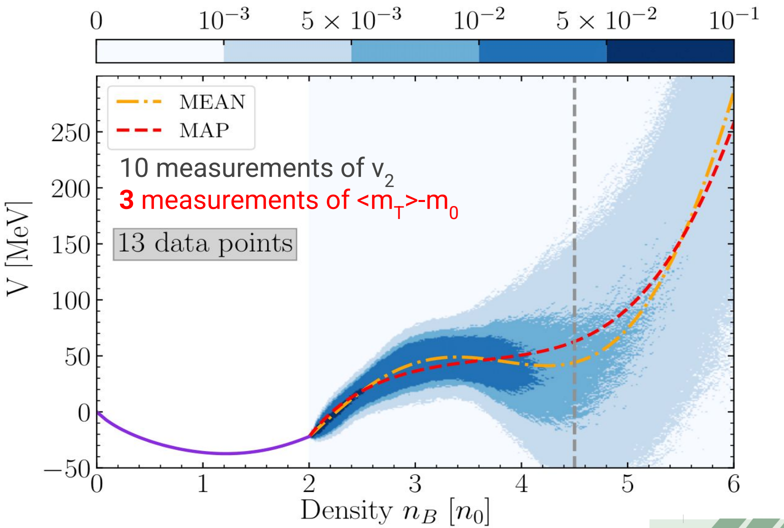
Maps the EoS to UrQMD observables



The need for faster simulations!

Phys Rev Lett 131 (20), 202303





The results depend on choice of observables !

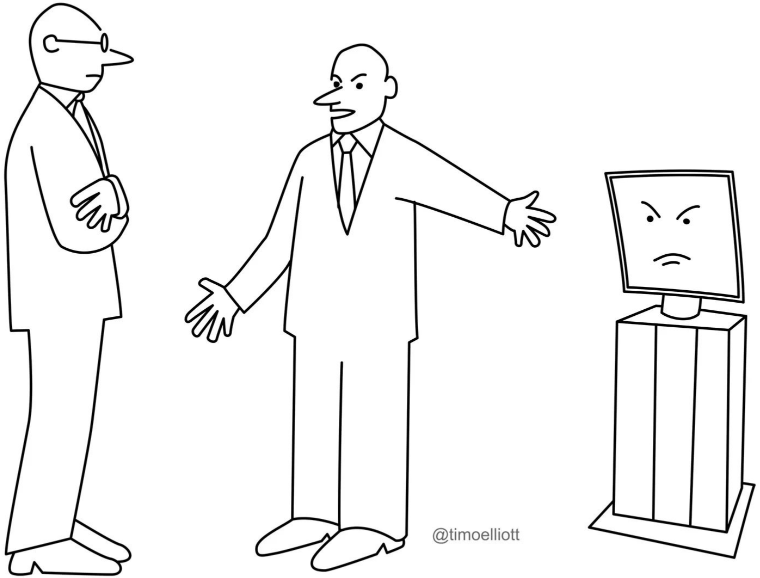
- Comprehensive bayesian inference necessary for unambiguous solution
    - expensive, multi differential observables => GP models not feasible!
- Next-gen experiments will provide immense amount of high precision data
    - Alternate techniques to accelerate model simulations are necessary!

The solution: an AI clone of the physics model

Generate the entire collision output instead of predicting specific observable !

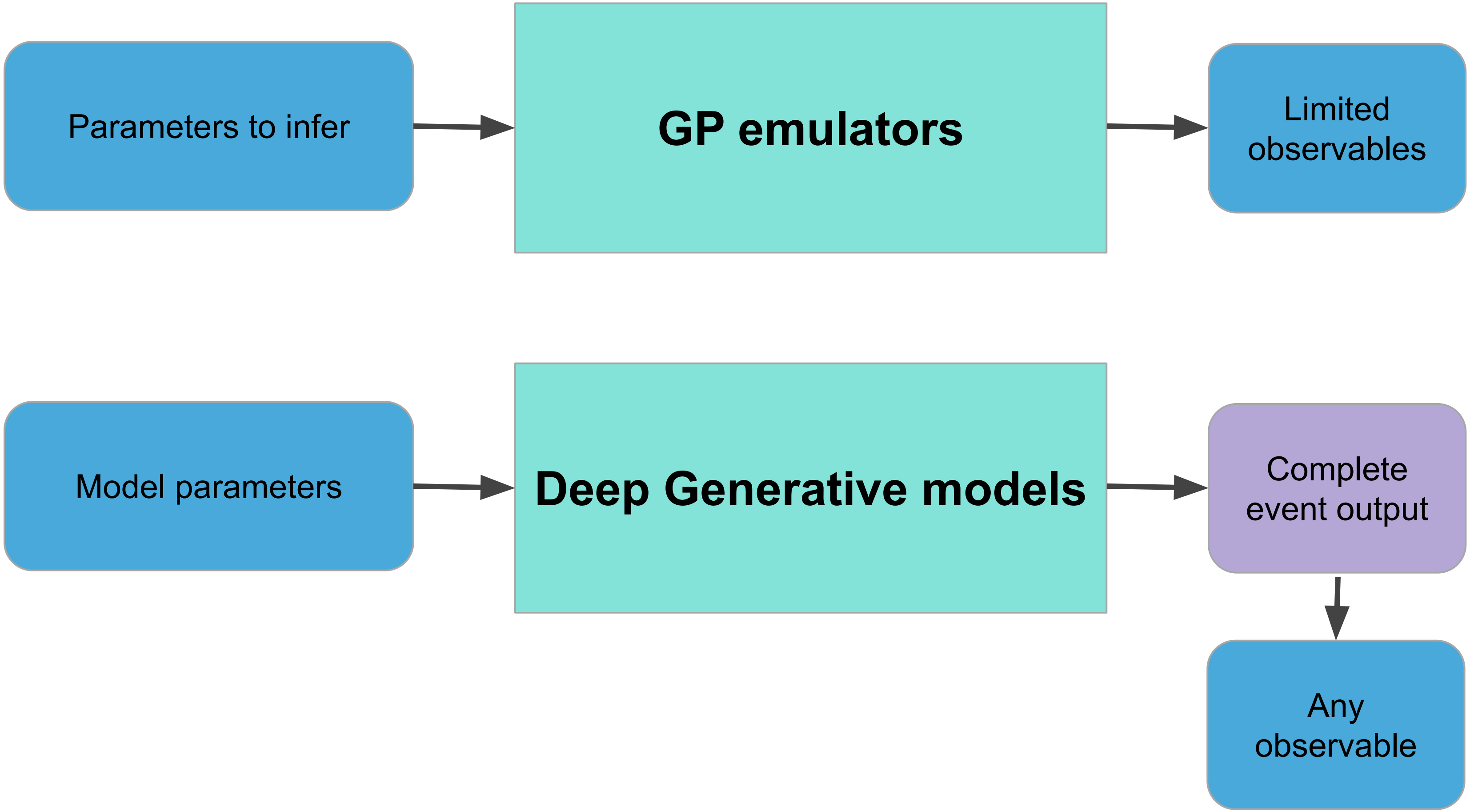
With a fast generative model:

- Any necessary observable can be calculated
  - training new models not necessary
- Infer any physics of interest!
  - Not limited to EoS



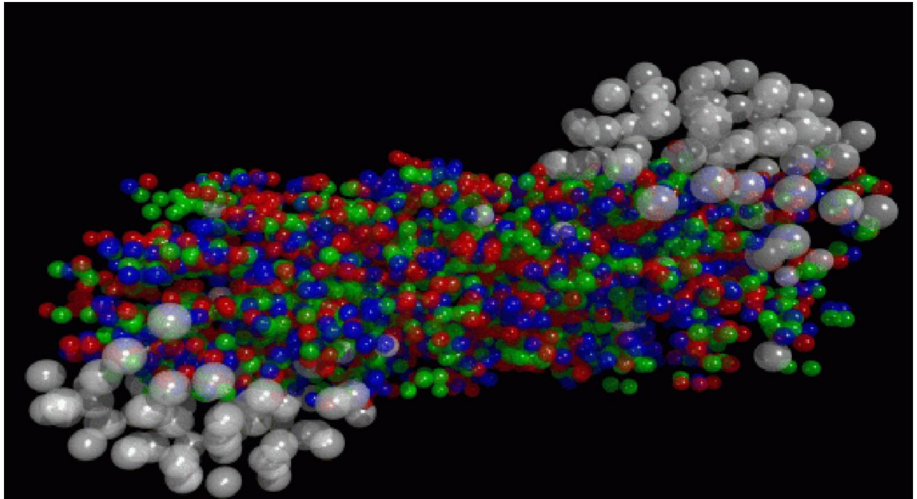
His decisions aren't any better than yours — but they're WAY faster...

An AI clone of your physics model



UrQMD cascade: a microscopic model for collisions

- Event-by-event collision output
- Microscopic non-equilibrium description
- hadrons on classical trajectories
  - stochastic binary scatterings
  - color string formation
  - resonance excitation and decays



https://itp.uni-frankfurt.de/~bleicher/urqmd-cern.gif

- interactions based on scattering cross sections
- default setup effective EoS: Hadron Resonance Gas
- Non-trivial interactions can be added through QMD approach

Can we emulate UrQMD with DL?

A collision event output

- UrQMD outputs a list of final state hadrons along their momentum info
- Pointclouds: ideal representation

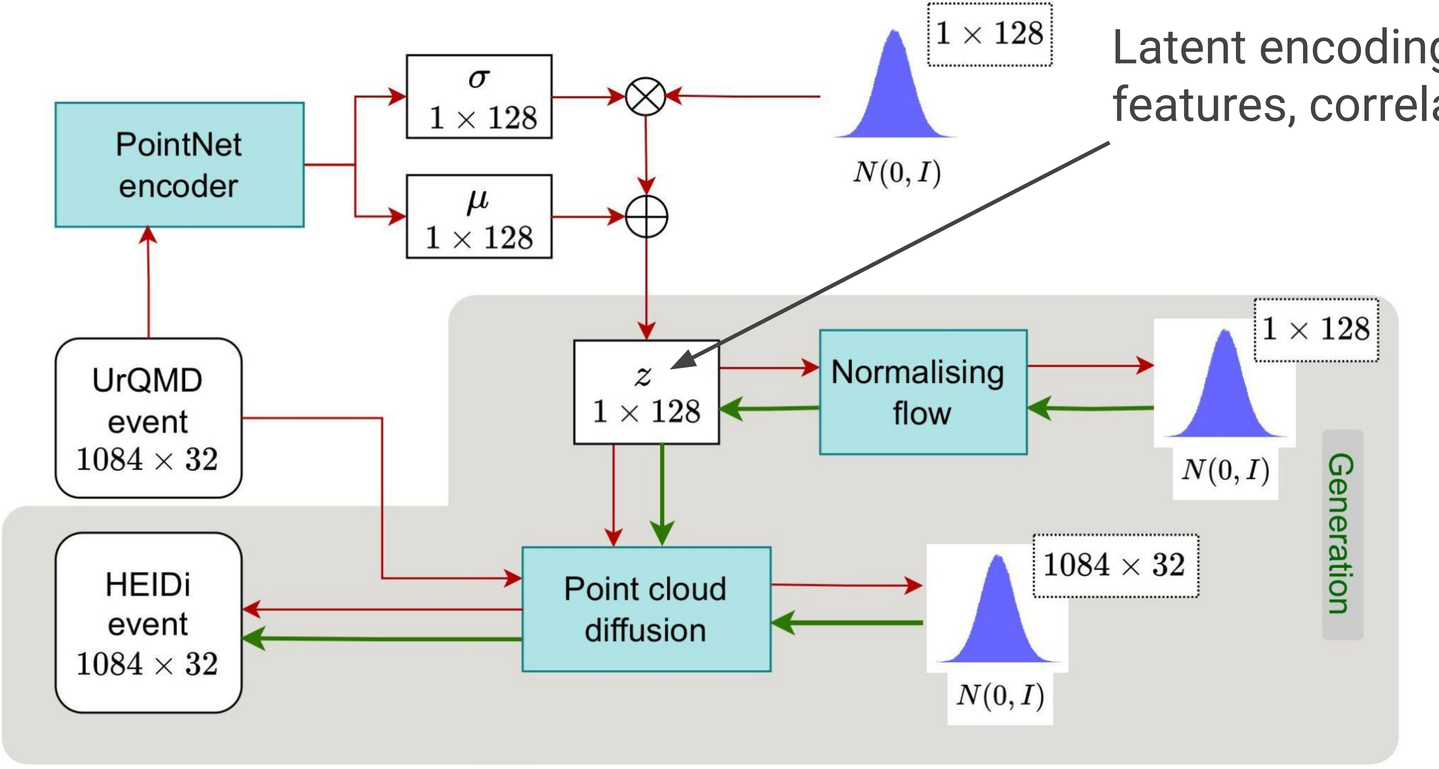
- Consider Au-Au 10 AGeV, impact parameter b=1 fm
  - An event= 1084 X 32
  - Empty rows=0,0,0,0,...
  - p_x, p_y, p_z, One hot encoded PID
  - 26 hadron species, spectator nucleons, empty particles

$$\mathbf{X}^{(0)} = \{\mathbf{x}_i^{(0)}\}_{i=1}^{1084}$$

$$\mathbf{x}_i^{(0)} = \{\mathbf{p}_i^{(0)}, \mathrm{ID}_i^{(0)}\},$$

$$\mathbf{p}_i^{(0)} = (p_{x_i}^{(0)}, p_{y_i}^{(0)}, p_{z_i}^{(0)})$$

HEIDi: a generative model for heavy-ion reactions



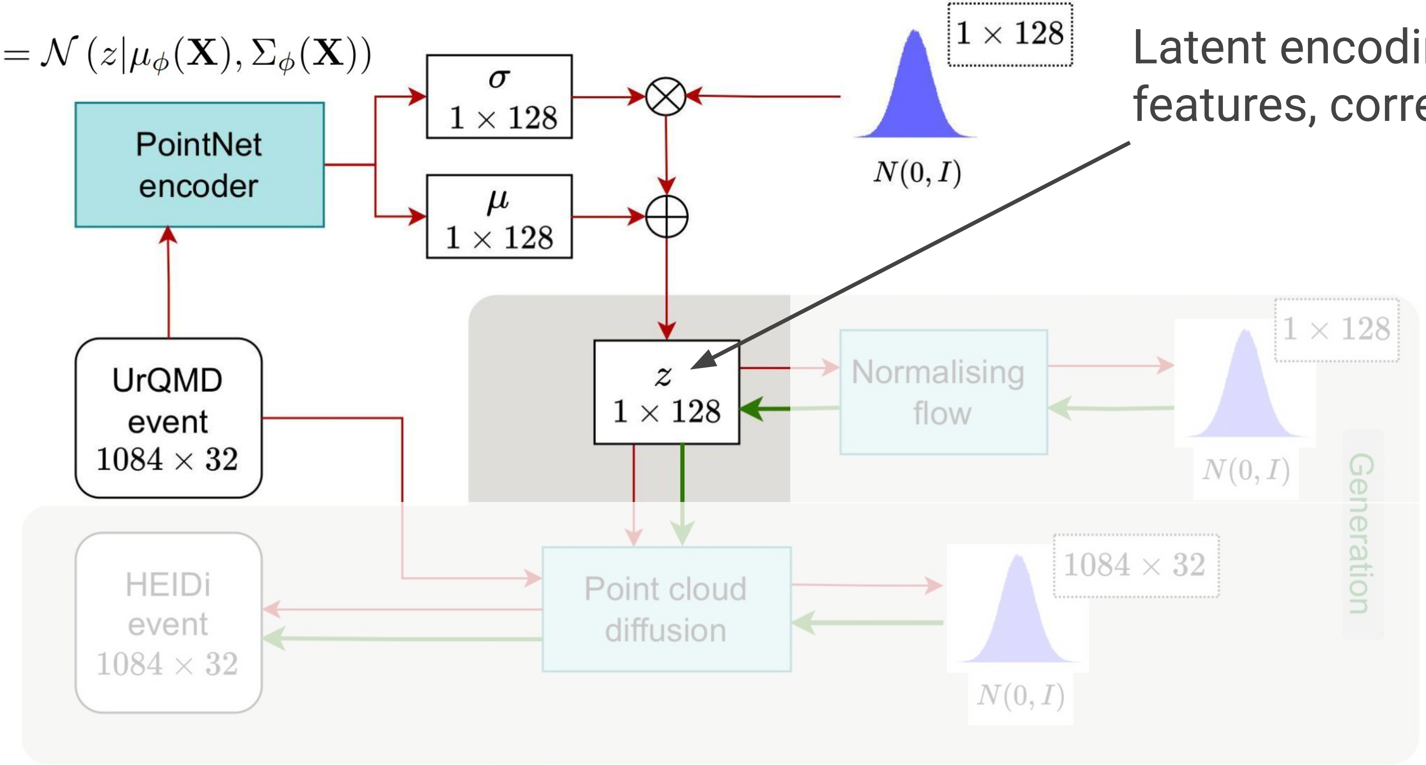
Latent encoding of event features, correlations

PointNet encoder + Normalizing flow decoder + Pointcloud diffusion

Based on arXiv:2103.01458

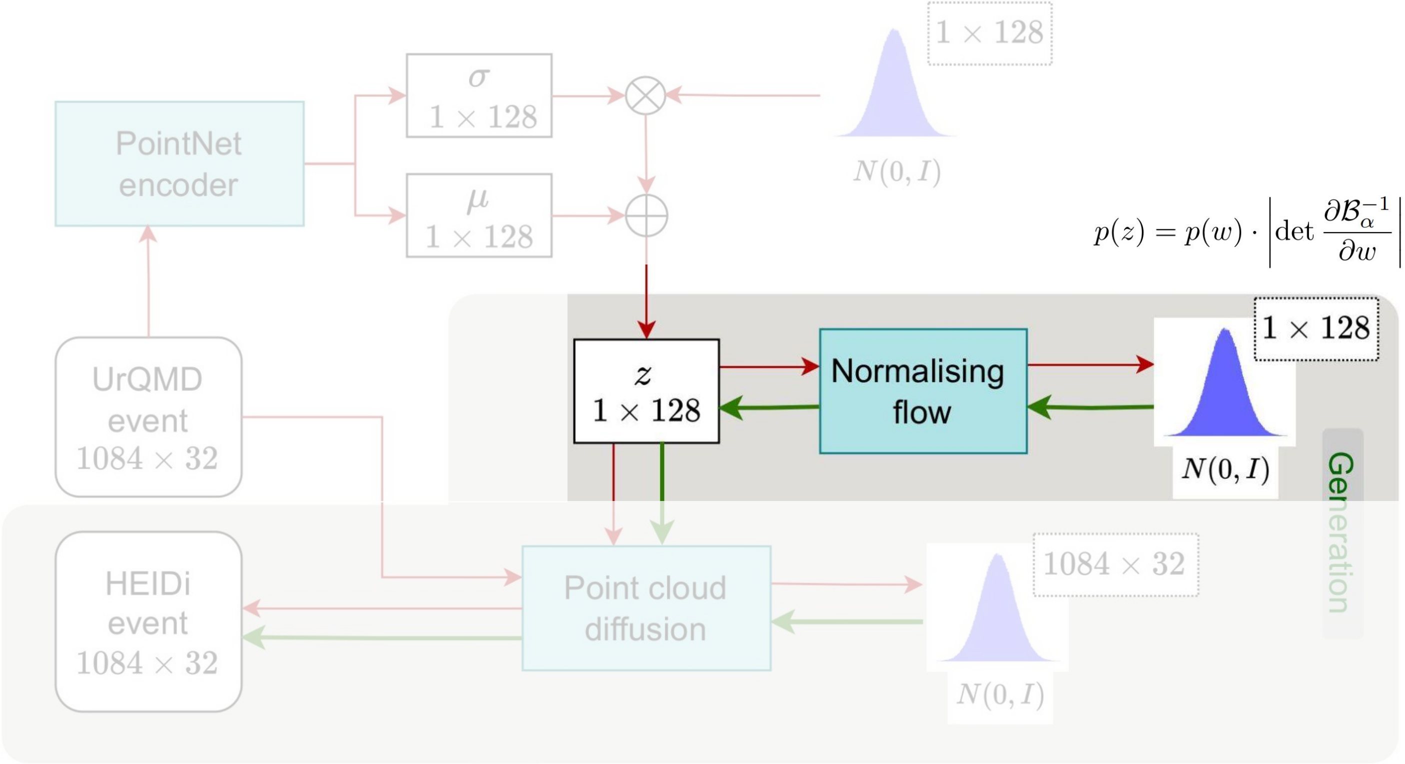
HEIDi: a generative model for heavy-ion reactions

$$\tilde{q}_\phi(z|\mathbf{X}) = \mathcal{N}\left(z|\mu_\phi(\mathbf{X}), \Sigma_\phi(\mathbf{X})\right)$$



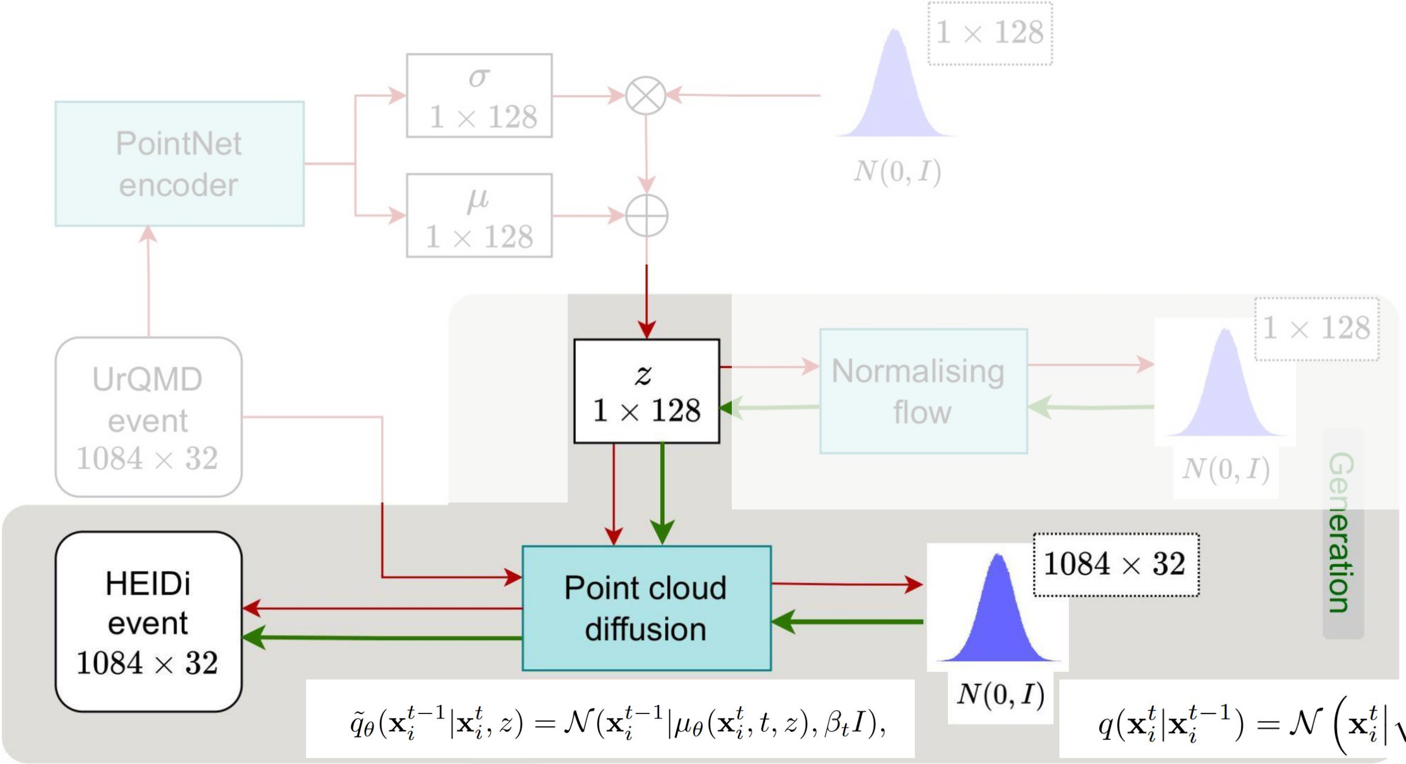
Latent encoding of event features, correlations

HEIDi: a generative model for heavy-ion reactions



$$p(z) = p(w) \cdot \left| \det \frac{\partial \mathcal{B}_\alpha^{-1}}{\partial w} \right|^{-1}$$

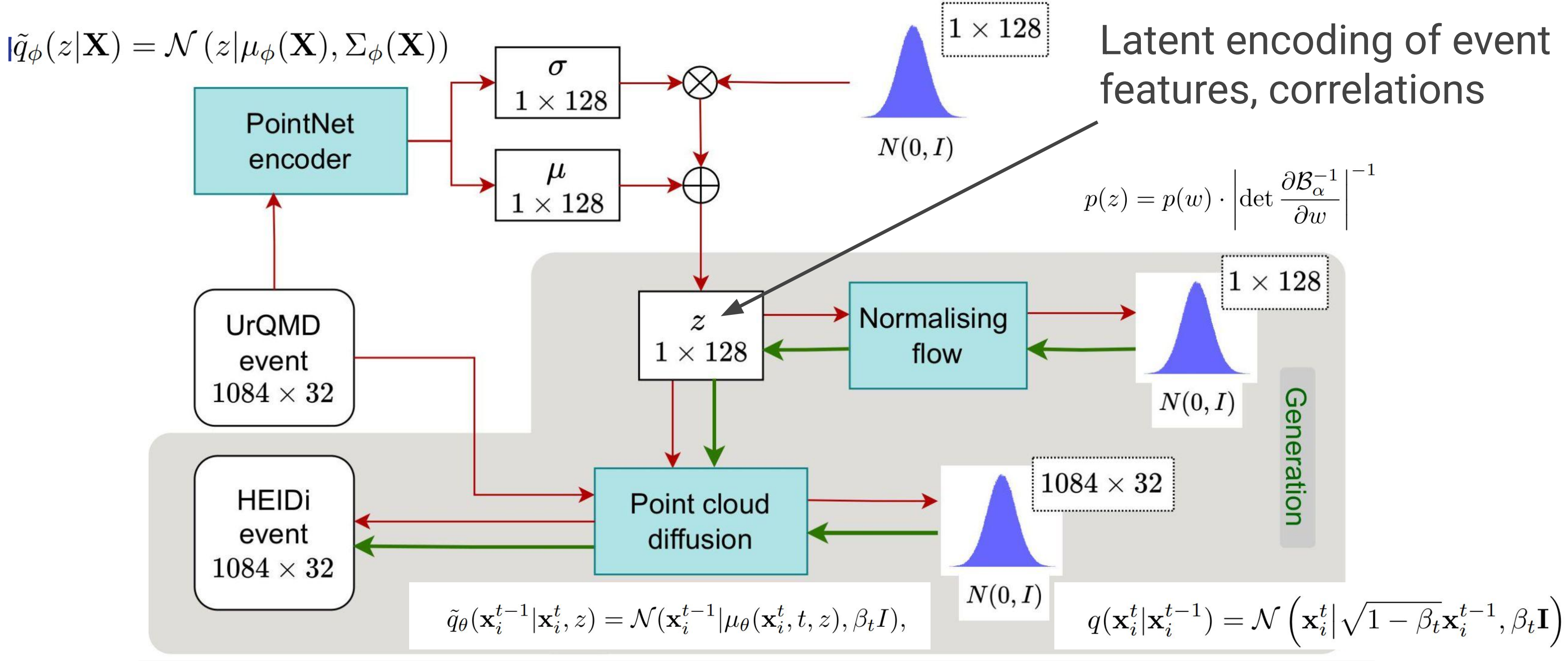
HEIDi: a generative model for heavy-ion reactions



$$\tilde{q}_\theta(\mathbf{x}_i^{t-1}|\mathbf{x}_i^t, z) = \mathcal{N}(\mathbf{x}_i^{t-1}|\mu_\theta(\mathbf{x}_i^t, t, z), \beta_t I),$$

$$q(\mathbf{x}_i^t|\mathbf{x}_i^{t-1}) = \mathcal{N}\left(\mathbf{x}_i^t\middle|\sqrt{1-\beta_t}\mathbf{x}_i^{t-1}, \beta_t \mathbf{I}\right)$$

HEIDi: a generative model for heavy-ion reactions



Latent encoding of event features, correlations

Learning the collision output

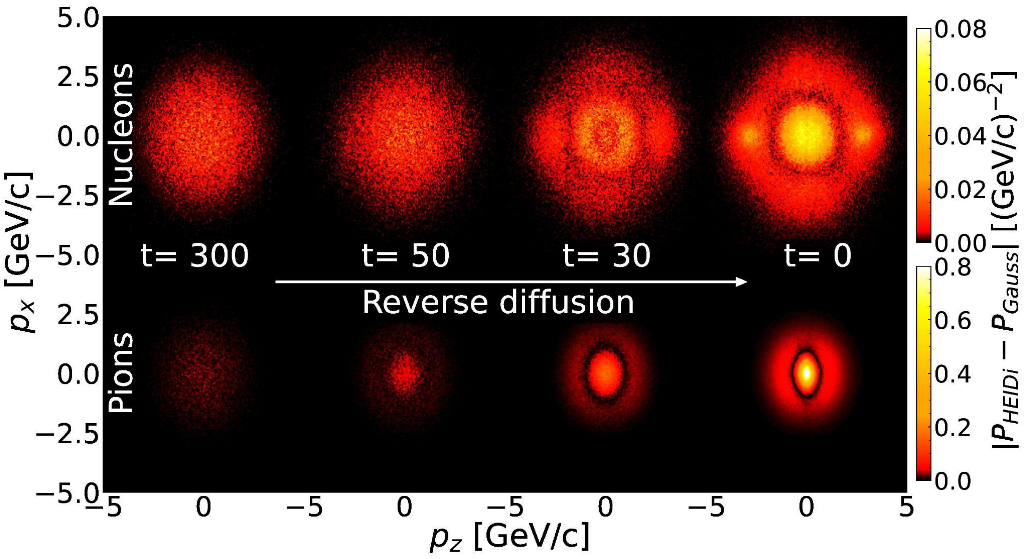
$$\mathbf{x}_i^{(0:T)} = \{\mathbf{x}_i^{(0)}, \mathbf{x}_i^{(1)}, \ldots, \mathbf{x}_i^{(T)}\}$$

The probability of states 0,1,...T, given z:

$$\tilde{q}_\theta(\mathbf{x}_i^{(0:T)}|z) = \tilde{q}(\mathbf{x}_i^{(T)}) \prod_{t=1}^{T} \tilde{q}_\theta(\mathbf{x}_i^{(t-1)}|\mathbf{x}_i^{(t)}, z).$$

The reverse diffusion:

$$\tilde{q}_\theta(\mathbf{x}_i^{(t-1)}|\mathbf{x}_i^{(t)}, z) = \mathcal{N}(\mathbf{x}_i^{(t-1)}|\mu_\theta(\mathbf{x}_i^{(t)}, t, z), \beta_t I),$$

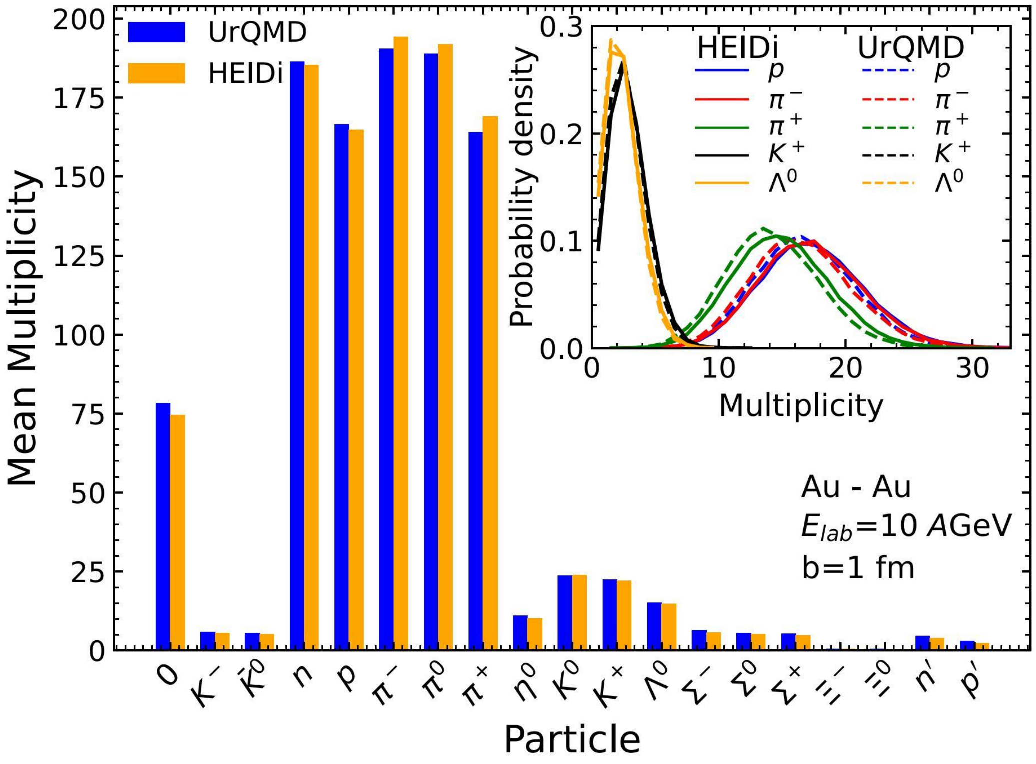


HEIDi is trained on 18000 UrQMD cascade events

Results: Mean Multiplicity of various hadrons

- At mid rapidity, for selected hadrons

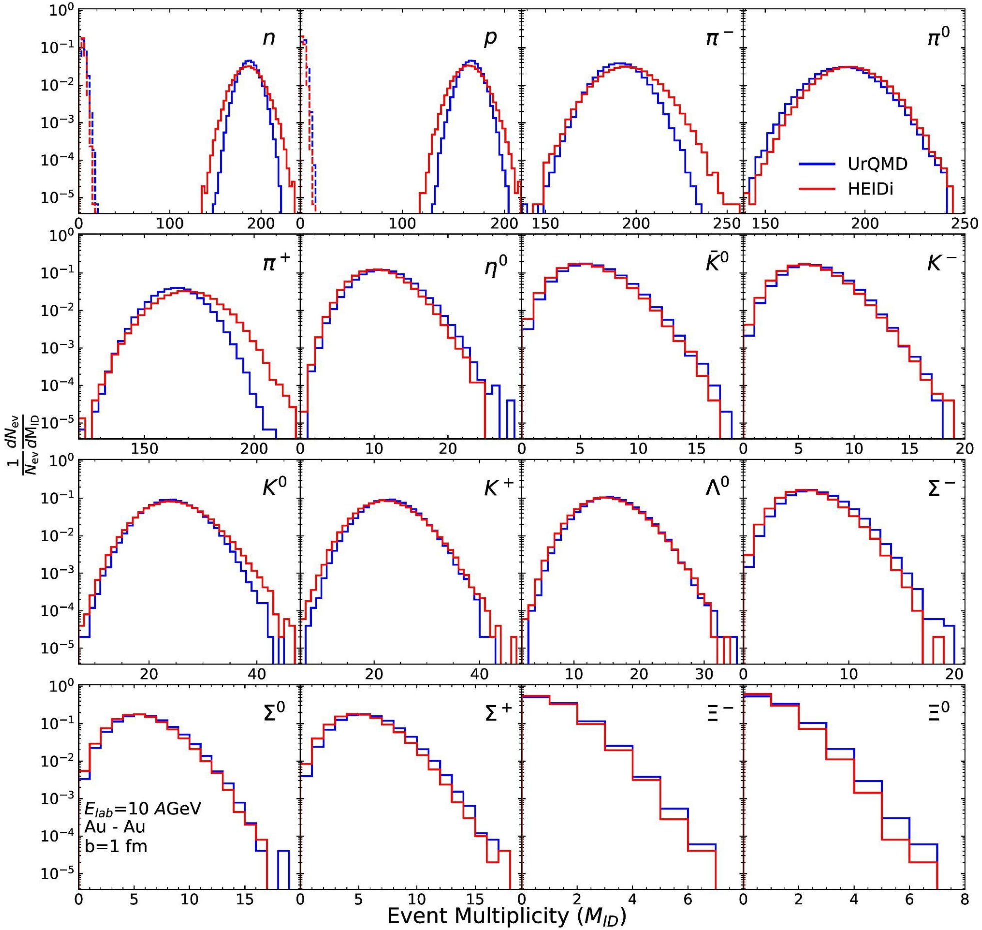
Results from 50000 events



- Very good agreement with UrQMD

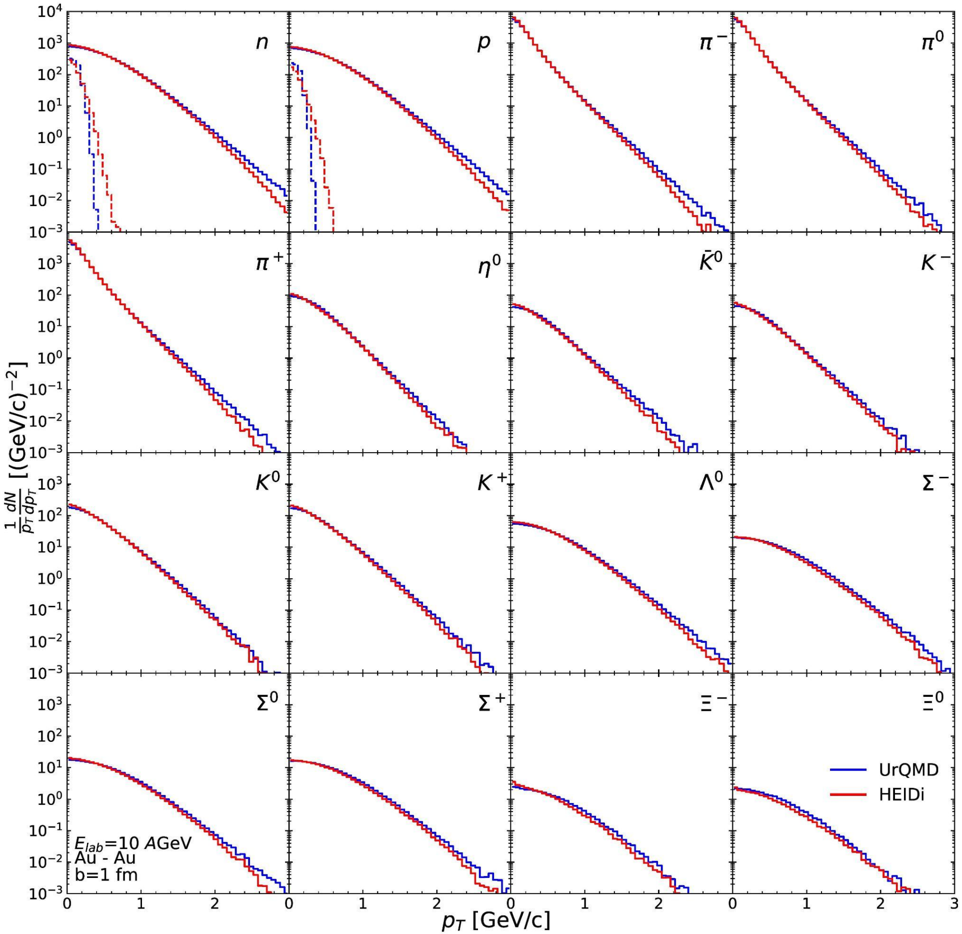
- Captures the relative difference differences in the multiplicities of various hadrons in an event

Results: Event Multiplicity of different hadrons



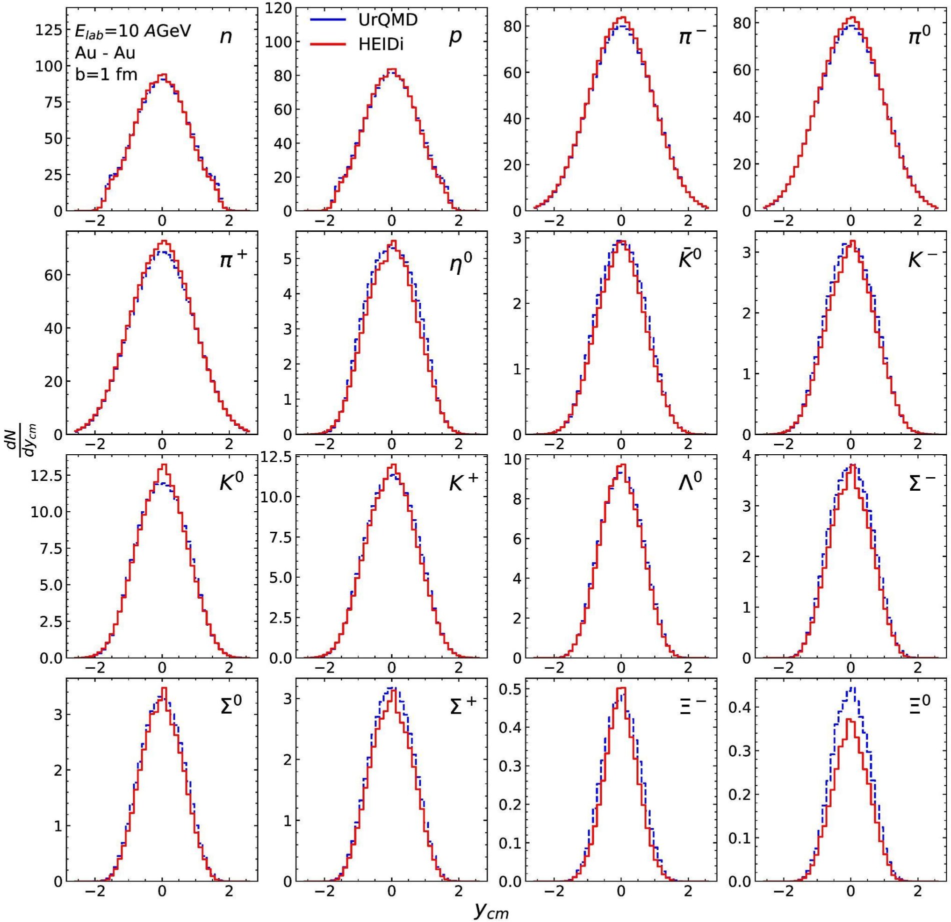
- Good agreement with UrQMD
- Learns the drastic difference in multiplicity of spectators and participants
- Deviations/ offset at the tails for certain species
  - Due to limited training size?

Results: Transverse momentum distributions



- Learns the probabilities of different hadrons across 5 orders of magnitude
  - from just 18000 events
  - not trivial !
- Very good agreement with only small deviations at the tails

Results: Rapidity distribution of hadrons

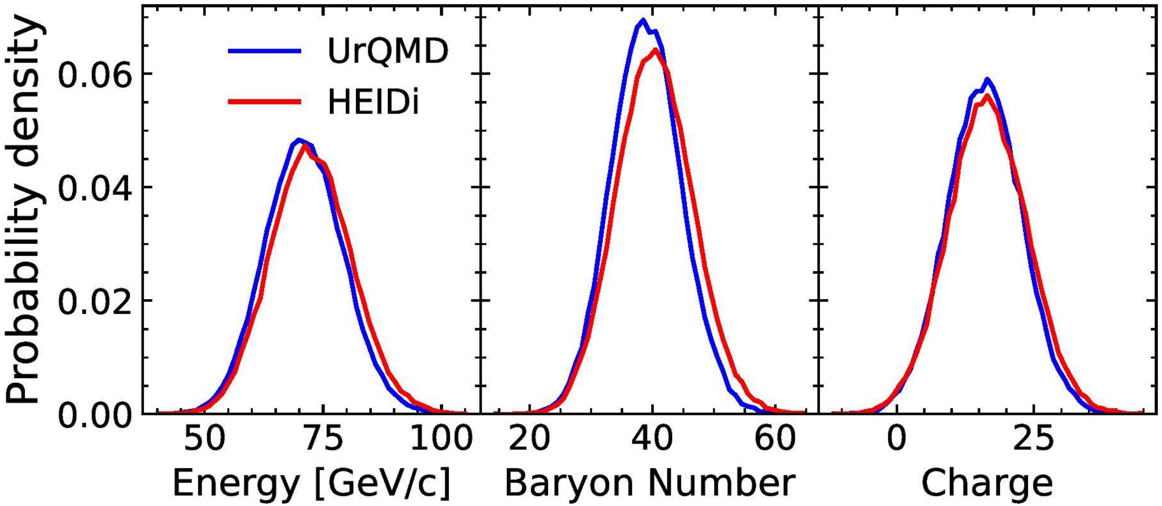


$$y = \frac{1}{2}\ln\left(\frac{E+p_z}{E-p_z}\right)$$

- Rapidities are well reproduced
  - Small deviations at mid rapidities
- Model overestimates very low momentum particles
  - Due to limited sample?
  - More diverse training data needed?

Results: Global collision properties

- HEIDi also learns different global event features
- The total energy, total baryon number and total charge of hadrons at midrapidity show good agreement with UrQMD distributions



Results: Speed of event generation

- URQMD cascade: one of the fastest model
  - ~ 3 Sec/ event
- HEIDi on NVIDIA A-100 GPU:
  - ~30 ms /event
- HEIDi can be easily adapted for other more expensive complex physics models
  - UrQMD with potential ~3 min/event
  - URQMD hybrid (with hydro intermediate stage) **~1 hour/ event**

- UrQMD hybrid: Speedup of at least 5 orders of magnitude can be expected !

Outlook

- HEIDi is a point cloud diffusion model for ultra-fast e-b-e collision output generation
  - 26 hadrons species
  - Complete event output
  - Generates particles , not spectra or aggregate information
- Accurately learns various properties of different hadron species and global collision features
- 100x speedup: up to 10000x possible for hydro-codes

Next step:

- Conditional generation in HEIDi
  - collision energies, collision systems, centrality and EoS
  - Enables comprehensive bayesian inference
  - Multi differential observables can be used for inference

Outlook

Advantages of HEIDi based models

- Not limited to EoS, but any physics can be studied
- Direct inference of physics of interest from experimental data
  - Gradient based optimisation techniques
- Can be adapted for any theoretical model, detector simulations
- Useful for real time experimental data analysis, quality check, trigger on interesting physics etc.

Variants of HEIDi under development

- HEIDi for ultra fast cosmic shower simulations - bachelor thesis: Lina Jeritslev
- HEIDi for UrQMD with potentials

Backup slides

Physics inference from data: the computational nightmare

What a simulation model does:

Physics of interest → **Models** transport/ hydrodynamics/ hybrid etc. → expected data

What's necessary for physics inference:

Underlying physics ← **Models** transport/ hydrodynamics/ hybrid etc. ← measured data

Physics inference: Inverse problem

- Multi-param fits, bayesian inference
- e.g. EoS, Phase transitions

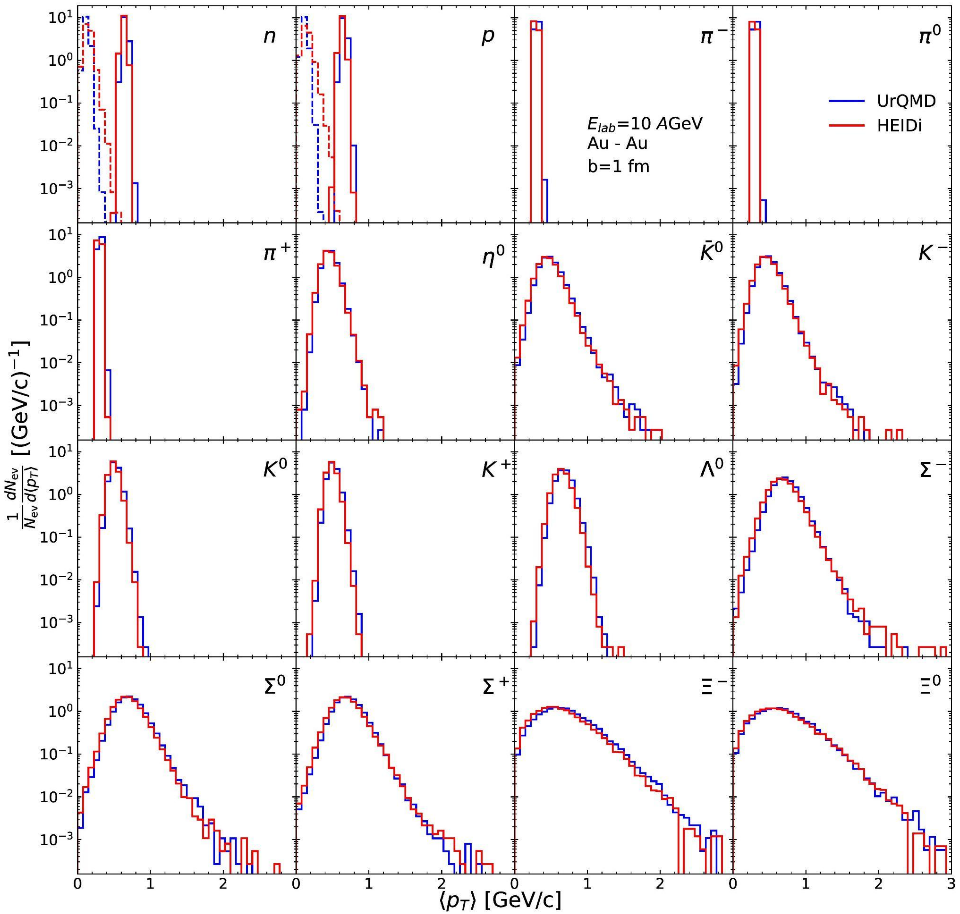
Expensive model - data comparisons are necessary !

The network

$$\mathcal{L} = \mathbb{E}_q \left[\sum_{t=2}^{T} \sum_{i=1}^{N} D_{KL}\left(q(\mathbf{x}_i^{t-1} | \mathbf{x}_i^t, \mathbf{x}_i^0) \middle\| \tilde{q}_\theta(\mathbf{x}_i^{t-1} | \mathbf{x}_i^t, z) \right) - \sum_{i=1}^{N} \log \tilde{q}_\theta \left(\mathbf{x}_i^0 | \mathbf{x}_i^1, z \right) + D_{KL}\left(\tilde{q}_\phi(z | \mathbf{X}^0) \middle\| p(z) \right) \right].$$

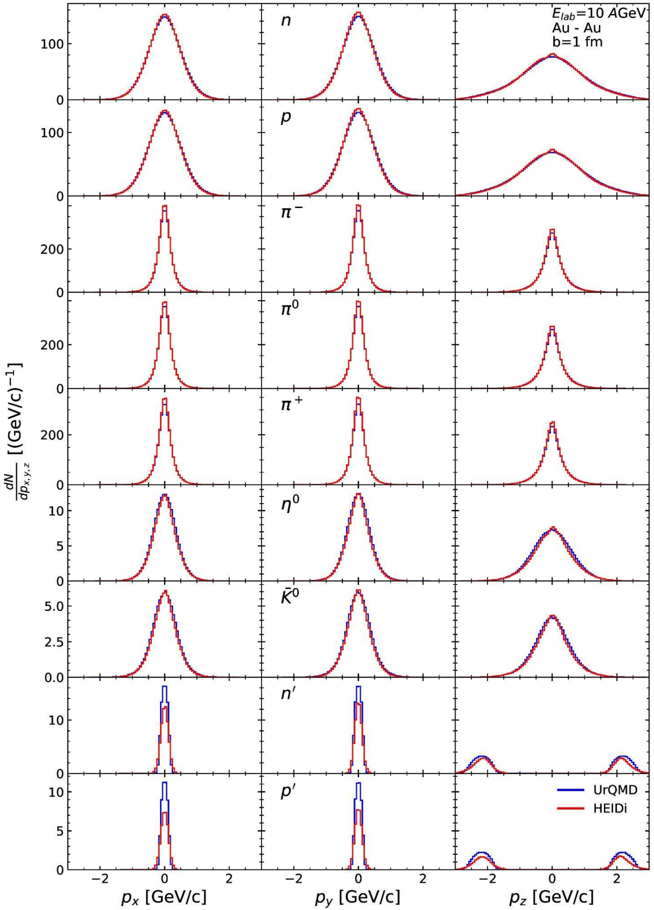
FlowVAE(\n",
" (encoder): **4 X** Conv1d, l**3 x** inear layers for mean, sigma
" (conv1): Conv1d(32, 128, kernel_size=(1,), stride=(1,))\n",
" (conv2): Conv1d(128, 128, kernel_size=(1,), stride=(1,))\n",
" (conv3): Conv1d(128, 256, kernel_size=(1,), stride=(1,))\n",
" (conv4): Conv1d(256, 512, kernel_size=(1,), stride=(1,))\n", +batchnormalisation layers
" (fc1_m): Linear(in_features=512, out_features=256, bias=True)\n",
" (fc2_m): Linear(in_features=256, out_features=128, bias=True)\n",
" (fc3_m): Linear(in_features=128, out_features=128, bias=True)\n",
" (fc1_v): Linear(in_features=512, out_features=256, bias=True)\n",
" (fc2_v): Linear(in_features=256, out_features=128, bias=True)\n",
" (fc3_v): Linear(in_features=128, out_features=128, bias=True)\n",
" (flow): SequentialFlow(\n",
" (chain): ModuleList(\n",
" (0-13): **14 x** CouplingLayer(\n",
" (net_s_t): Sequential(\n",
" (0): Linear(in_features=64, out_features=256, bias=True)\n",
" (1): ReLU(inplace=True)\n",
" (2): Linear(in_features=256, out_features=256, bias=True)\n",
" (3): ReLU(inplace=True)\n",
" (4): Linear(in_features=256, out_features=128, bias=True)\n",
")
(diffusion): **5 x** COncatsquash layers
" (0): ConcatSquashLinear(\n", [h`+1 = CS(h`, t, z) = (W1h` + b1) σ(W2c + b2) + W3c]
" (_layer): Linear(in_features=32, out_features=128, bias=True)\n",
" (_hyper_bias): Linear(in_features=131, out_features=128, bias=False)\n",
" (_hyper_gate): Linear(in_features=131, out_features=128, bias=True)\n",
")\n",

Results

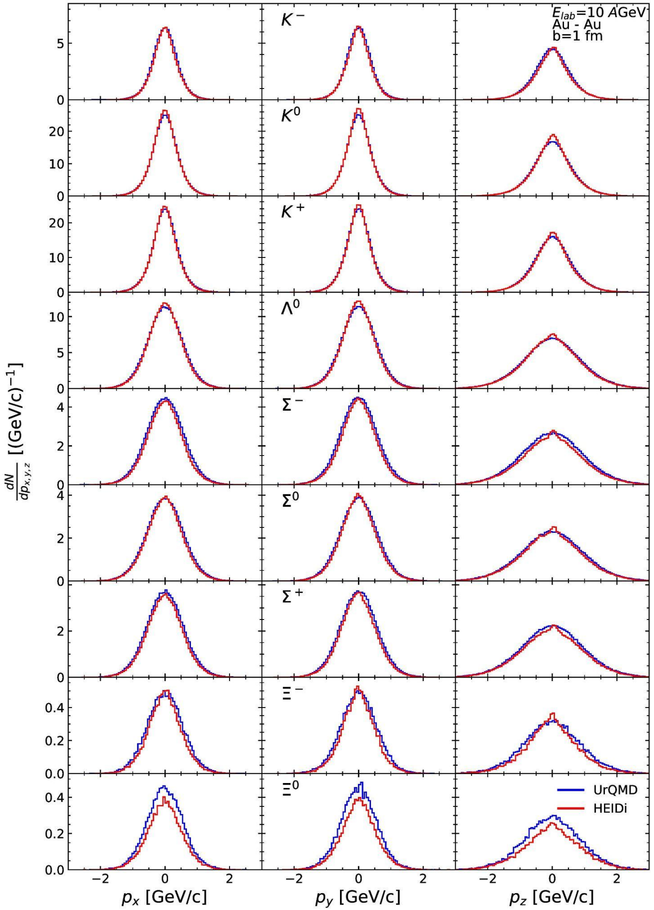


n
p
π⁻
π⁰
UrQMD
HEIDi
E_lab=10 AGeV
Au - Au
b=1 fm
π⁺
η⁰
K̄⁰
K⁻
K⁰
K⁺
Λ⁰
Σ⁻
Σ⁰
Σ⁺
Ξ⁻
Ξ⁰
1/N_ev dN_ev/d⟨p_T⟩ [(GeV/c)⁻¹]
⟨p_T⟩ [GeV/c]

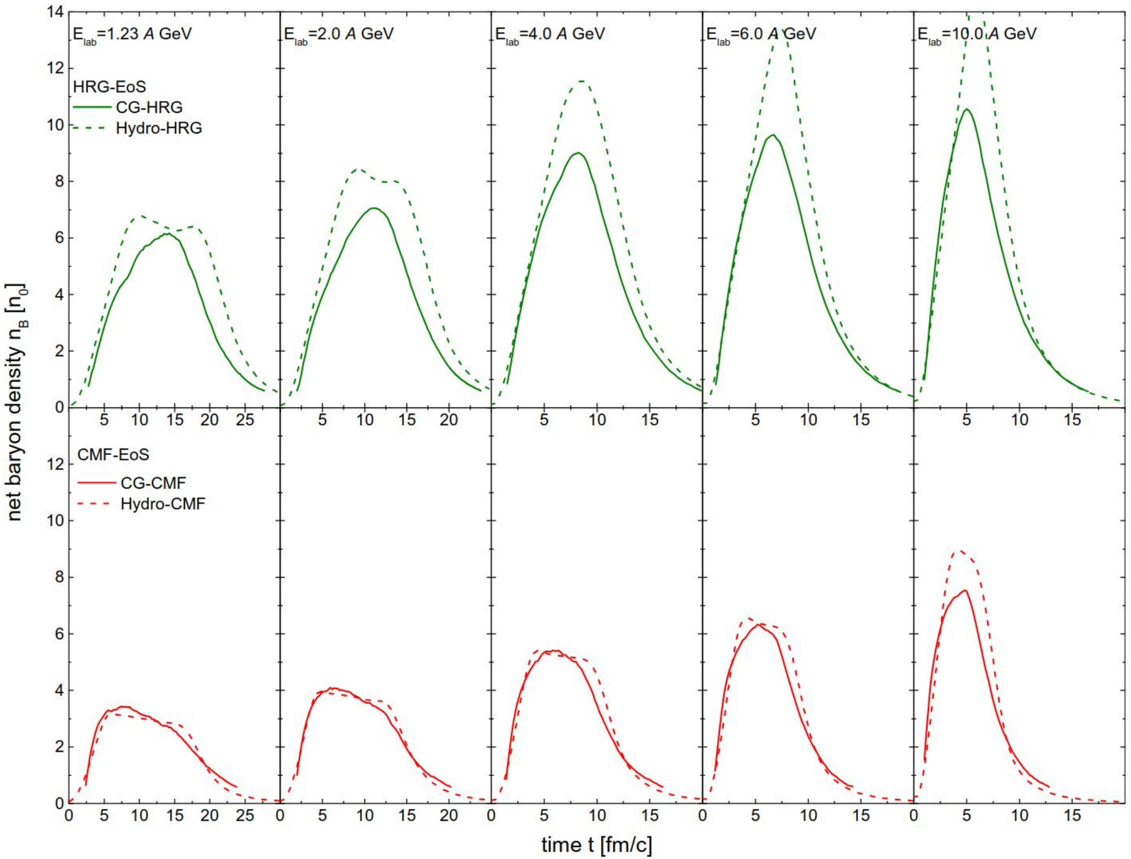
Results



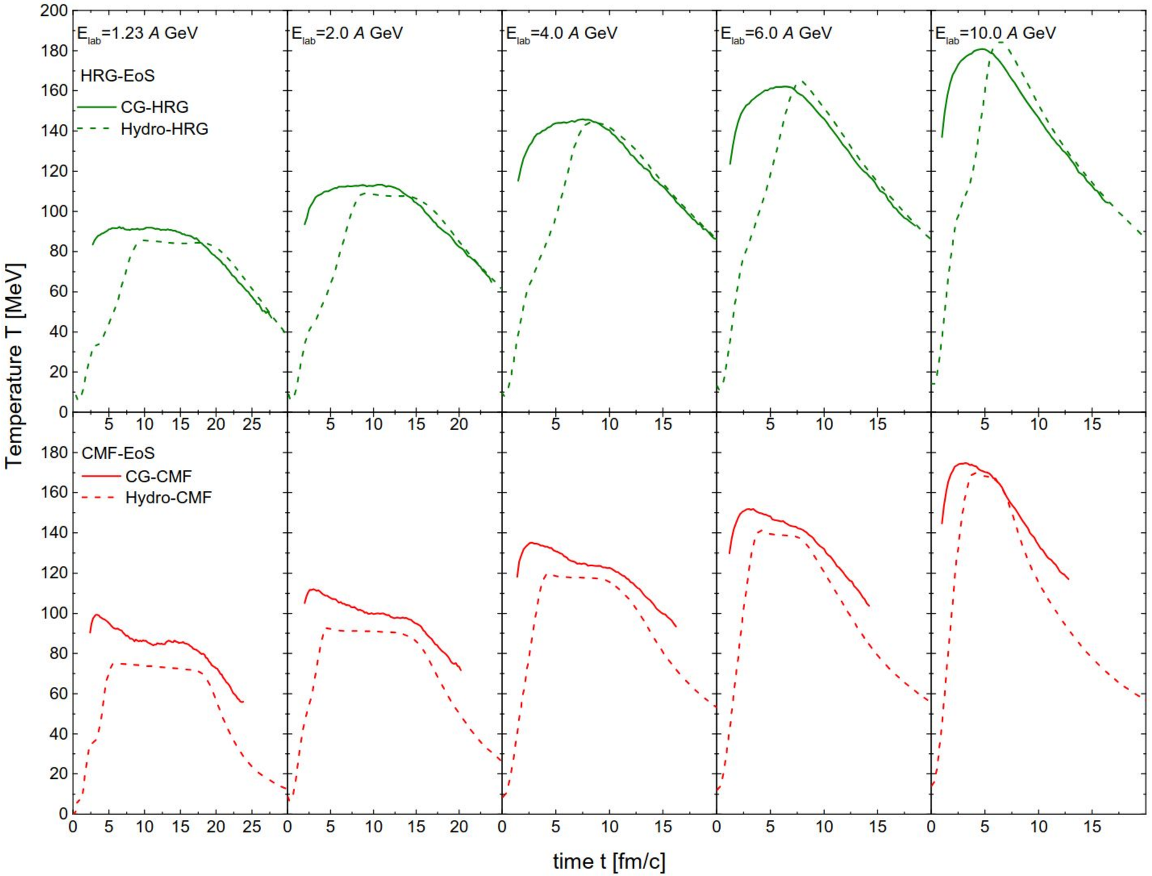
E_{lab}=10 AGeV
Au - Au
b=1 fm
n
p
π^-
π^0
π^+
η^0
$\bar{K}^0$
n'
p'
UrQMD
HEIDi
$\frac{dN}{dp_{x,y,z}}$ [(GeV/c)$^{-1}$]
p_x [GeV/c]
p_y [GeV/c]
p_z [GeV/c]



E_{lab}=10 AGeV
Au - Au
b=1 fm
K^-
K^0
K^+
Λ^0
Σ^-
Σ^0
Σ^+
Ξ^-
Ξ^0
UrQMD
HEIDi
$\frac{dN}{dp_{x,y,z}}$ [(GeV/c)$^{-1}$]
p_x [GeV/c]
p_y [GeV/c]
p_z [GeV/c]



E_{lab}=1.23 *A* GeV
E_{lab}=2.0 *A* GeV
E_{lab}=4.0 *A* GeV
E_{lab}=6.0 *A* GeV
E_{lab}=10.0 *A* GeV
HRG-EoS
CG-HRG
Hydro-HRG
CMF-EoS
CG-CMF
Hydro-CMF
net baryon density n_B [n_0]
time t [fm/c]



E_{lab}=1.23 A GeV
E_{lab}=2.0 A GeV
E_{lab}=4.0 A GeV
E_{lab}=6.0 A GeV
E_{lab}=10.0 A GeV
HRG-EoS
CG-HRG
Hydro-HRG
CMF-EoS
CG-CMF
Hydro-CMF
Temperature T [MeV]
time t [fm/c]

$$\mathcal{L} = \mathbb{E}_q\Bigg[\sum_{t=2}^{T}\sum_{i=1}^{N} D_{KL}\Big(q(\mathbf{x}_i^{t-1}|\mathbf{x}_i^t, \mathbf{x}_i^0)\Big\|\tilde{q}_\theta(\mathbf{x}_i^{t-1}|\mathbf{x}_i^t, z)\Big)$$
$$- \sum_{i=1}^{N} \log \tilde{q}_\theta\Big(\mathbf{x}_i^0|\mathbf{x}_i^1, z\Big) + D_{KL}\Big(\tilde{q}_\phi(z|\mathbf{X}^0)\Big\|p(z)\Big)\Bigg]$$