What do you want Muses to do for Heavy Ions?

Panel discussion

Moderator: Claudia Ratti

Panelists: Dekrayat Almaalol, Micheal Kahangirwe, Gregoire Pihan, Hitansh Shah, Volodymyr Vovchenko, Yumu Yang

NSF

muses

BQS EoS Module – Theory (Hitansh)

- We calculate pressure through Taylor expansion of lattice susceptibilities across $\mu_{B,Q,S}$
  - $\frac{P(T,\mu_B,\mu_Q,\mu_S)}{T^4} = \Sigma_{\{i,j,k\}} \frac{1}{i!\,j!\,k!} \chi^{BQS}_{ijk}\big|_{\mu_i=0} \left(\frac{\mu_B}{T}\right)^i \left(\frac{\mu_Q}{T}\right)^j \left(\frac{\mu_S}{T}\right)^k$
- Here, the lattice susceptibilities are parametrized as:
- $\chi_2^B(T) = e_1^{-\frac{h}{t'}-\frac{h_2}{t'}} \cdot f_3(1+\tanh(f_4 t' + f_5))$ where $t' = \frac{T}{200}\,MeV$
- $\chi^{BQS}_{ijk}(T) = \frac{\Sigma_{n=0}^{9} \frac{a_n^{ijk}}{t^n}}{\Sigma_{n=0}^{9} \frac{b_n^{ijk}}{t^n}} + c_0^{ijk}$ where $t' = \frac{T}{154}\,MeV$

Range of applicability:

$T \rightarrow [30{,}800]MeV$

$\mu_i \rightarrow [0{,}450]MeV$

Inputs :

- a_n, b_n, c_0 → 21 coefficients for 21 susceptibilities.
- Range for T, μ_B, μ_Q, μ_S

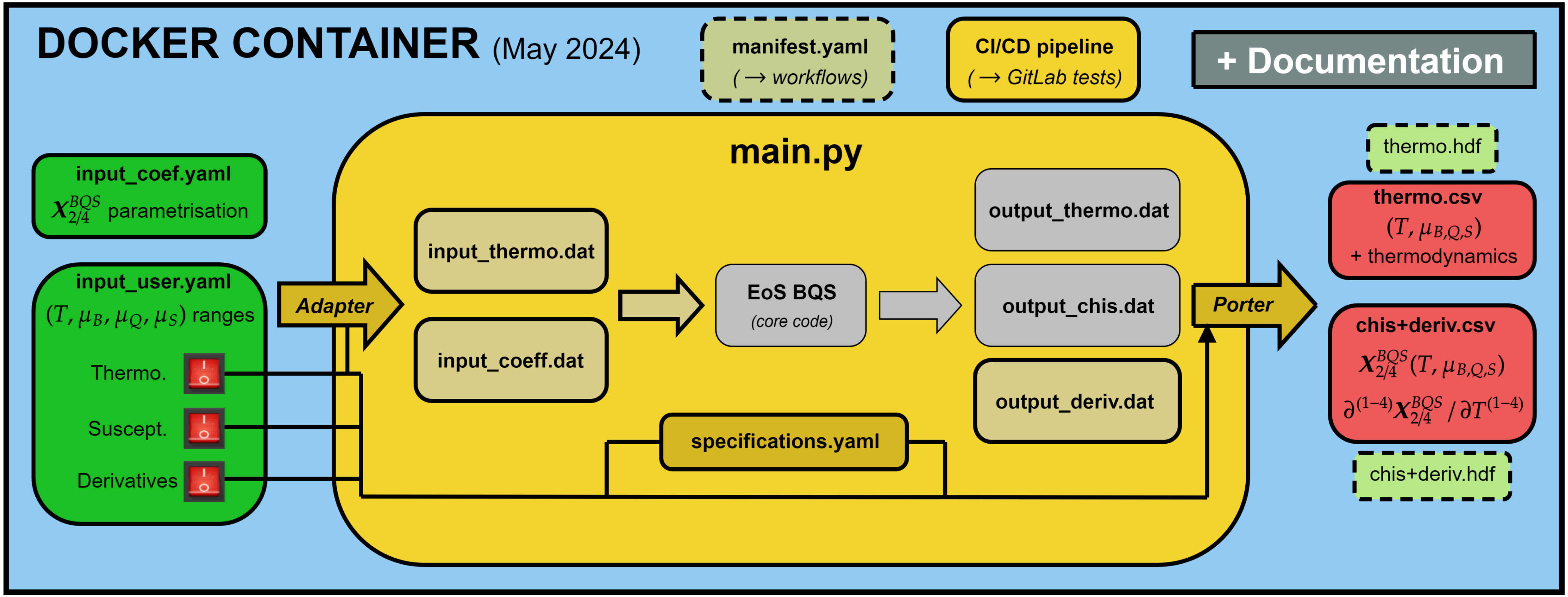
Outputs:

- $P\,(T,\mu_i), n_i\,(T,\mu_i),$ $\chi_2^i(T,\mu_i), \epsilon(T,\mu_i), s(T,\mu_i), c_s(T,\mu_i), \frac{dn^i}{dT}(T,\mu_i),$
- $\chi_2^i\,(T,\mathbf{0}), \chi_4^i(T,\mathbf{0}), \frac{d\chi_2^i}{dT}(T,\mathbf{0}), \frac{d\chi_4^i}{dT}(T,\mathbf{0})$

$i \rightarrow \{B,Q,S\}$

J. Noronha-Hostler et al., PRC 100 (2019) 064910

BQS EoS Module - Flowchart



DOCKER CONTAINER (May 2024)

manifest.yaml
(→ workflows)

CI/CD pipeline
(→ GitLab tests)

+ Documentation

input_coef.yaml
$\chi_{2/4}^{BQS}$ parametrisation

input_user.yaml
(T, μ_B, μ_Q, μ_S) ranges
Thermo.
Suscept.
Derivatives

Adapter

main.py

input_thermo.dat

input_coeff.dat

EoS BQS
(core code)

output_thermo.dat

output_chis.dat

output_deriv.dat

specifications.yaml

Porter

thermo.hdf

thermo.csv
$(T, \mu_{B,Q,S})$
+ thermodynamics

chis+deriv.csv
$\chi_{2/4}^{BQS}(T, \mu_{B,Q,S})$
$\partial^{(1-4)}\chi_{2/4}^{BQS} \,/\, \partial T^{(1-4)}$

chis+deriv.hdf

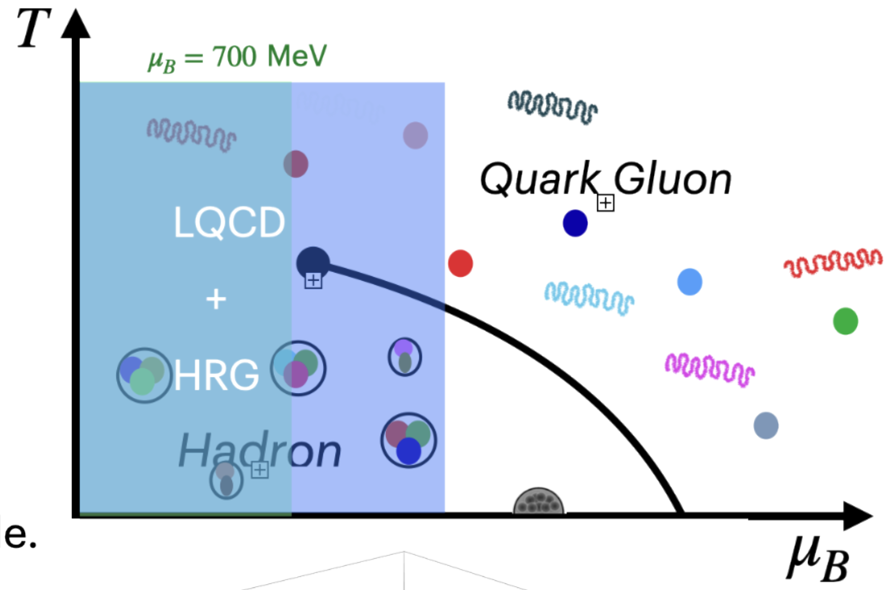
Original status

Current status

Remaining tasks

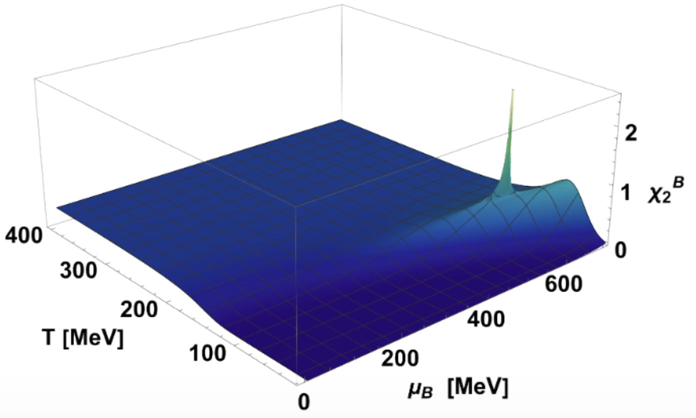
Ising-TeX – Theory (Micheal)

- This module uses current lattice (W B) merged w/ HRG model and T′-expansion scheme to extend the previous (Taylor) coverage from $\mu_B = [0,\ 450\ \text{MeV}]$ to $\mu_B = [0,\ 700\ \text{MeV}]$ & $T = [25,\ 800\ \text{MeV}]$
- Introduces a critical point by Mapping 3D-Ising to QCD
- Mapping has free parameters chosen by the User
  - 6 **Grid Inputs** $T_{min},\ T_{max},\ \delta T,\ \mu_{Bmin},\ \mu_{Bmax},\ \delta\mu_B$
  - 4 **Mapping Inputs** $\mu_{BC},\ w,\ \rho,\ \alpha_{12}$
- **Output** thermodynamics , $n_B(T,\mu_B), \chi_2^B(T,\mu_B),\ P(T,\mu_B),\ S(T,\mu_B),$
- $\epsilon(T,\mu_B),\ c_s^2,\ C_v$ with a chosen Grid in T and μ_B.
- The module can easily be reparameterized with new lattice data if available.



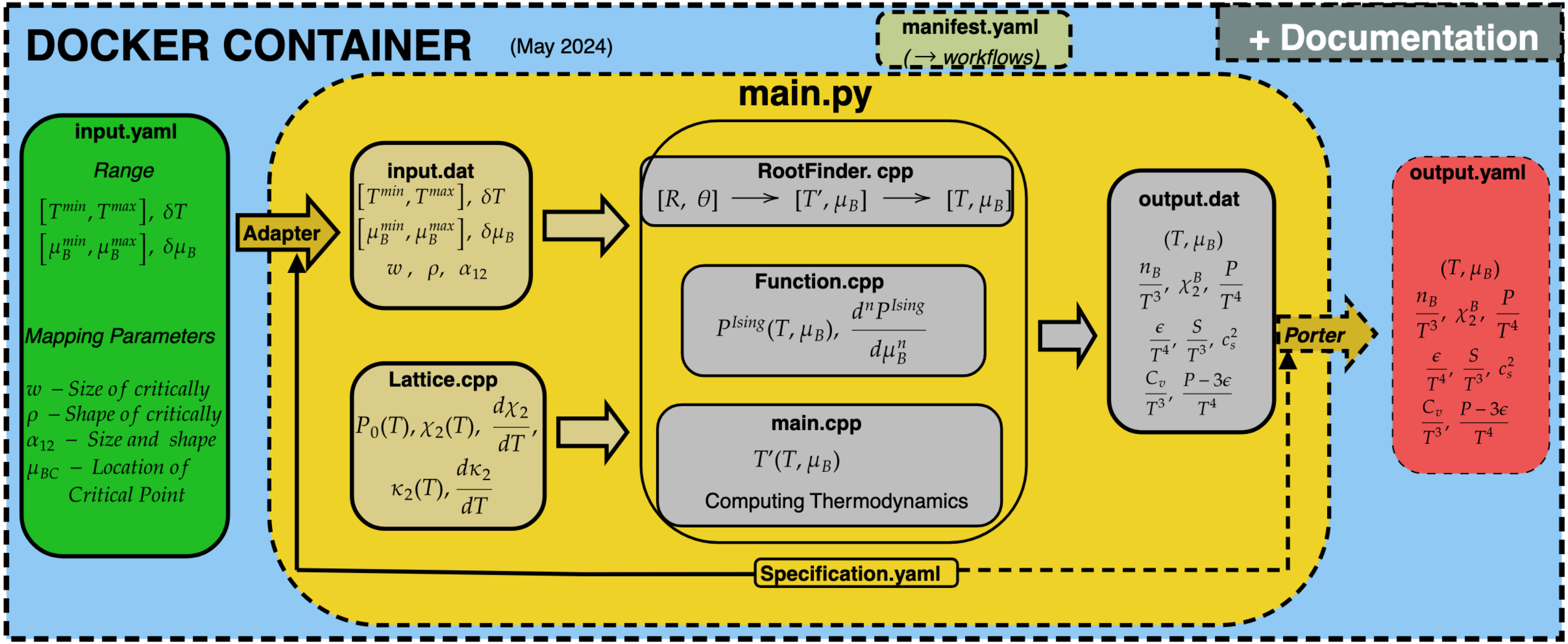
Challenges

- Numerical noise this limit computation high order derivatives
- Adaptive grind (more points around the critical region).
- Parameter Scan for stable EoS for each possible input choice.



M. Kahangirwe et al., PRD (2024) in press

Ising-TeX - Workflow



DOCKER CONTAINER (May 2024)

manifest.yaml
(→ workflows)

+ Documentation

main.py

input.yaml

Range

$[T^{min}, T^{max}]$, δT

$[\mu_B^{min}, \mu_B^{max}]$, $\delta\mu_B$

Mapping Parameters

w – *Size of critically*
ρ – *Shape of critically*
α_{12} – *Size and shape*
μ_{BC} – *Location of Critical Point*

Adapter

input.dat

$[T^{min}, T^{max}]$, δT
$[\mu_B^{min}, \mu_B^{max}]$, $\delta\mu_B$
w, ρ, α_{12}

Lattice.cpp

$P_0(T), \chi_2(T), \frac{d\chi_2}{dT}, \kappa_2(T), \frac{d\kappa_2}{dT}$

RootFinder. cpp

$[R, \theta] \longrightarrow [T', \mu_B] \longrightarrow [T, \mu_B]$

Function.cpp

$P^{Ising}(T, \mu_B), \frac{d^n P^{Ising}}{d\mu_B^n}$

main.cpp

$T'(T, \mu_B)$

Computing Thermodynamics

output.dat

(T, μ_B)

$\frac{n_B}{T^3}, \chi_2^B, \frac{P}{T^4}$
$\frac{\epsilon}{T^4}, \frac{S}{T^3}, c_s^2$
$\frac{C_v}{T^3}, \frac{P-3\epsilon}{T^4}$

Porter

output.yaml

(T, μ_B)

$\frac{n_B}{T^3}, \chi_2^B, \frac{P}{T^4}$
$\frac{\epsilon}{T^4}, \frac{S}{T^3}, c_s^2$
$\frac{C_v}{T^3}, \frac{P-3\epsilon}{T^4}$

Specification.yaml

Original status
Current status
Remaining tasks

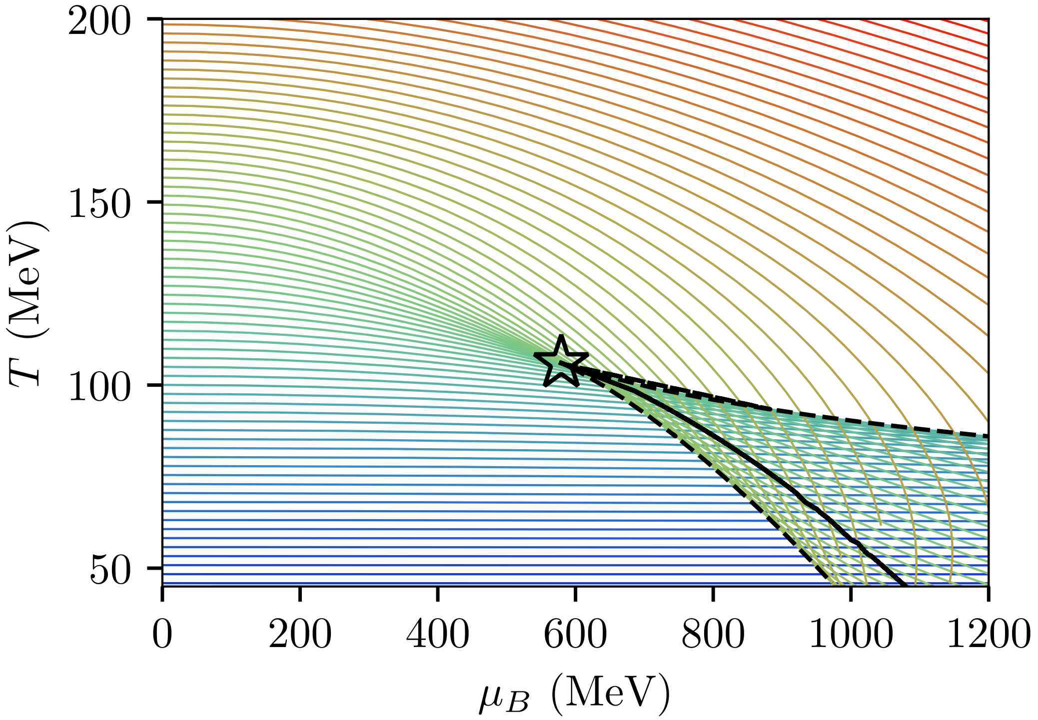
Holography – Theory (Yumu)

- Model QGP via AdS_5 black-hole dual: Einstein-Maxwell-Dilaton model

$$S = \frac{1}{2\kappa_5^2}\int_{\mathcal{M}_5} d^5x\sqrt{-g}\left[R - \frac{(\partial_\mu\phi)^2}{2} - V(\phi) - \frac{f(\phi)F_{\mu\nu}^2}{4}\right]$$

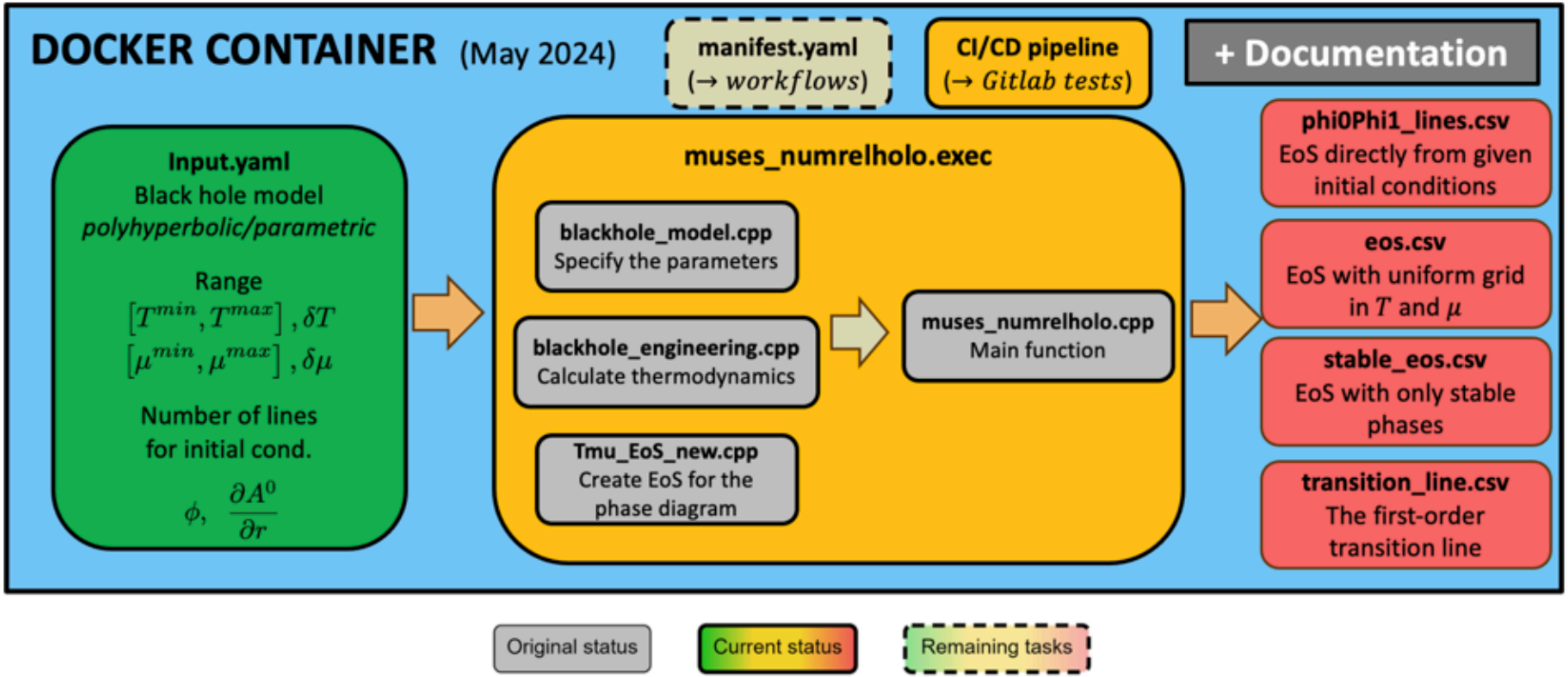
- Potentials $V(\phi)$ and $f(\phi)$ are tweaked to fit the lattice results
- Dual to baryon chemical potential μ : Abelian gauge field A^μ
- Initial conditions of $\left(\phi, \frac{\partial A^0}{\partial r}\right)$ can be mapped in to (T, μ)
- Coverage of the phase diagram:

$$\mathrm{T} \in (20, 400)\ \mathrm{MeV} \qquad \mu \in (0, 2000)\ \mathrm{MeV}$$



J. Grefa et al., PRD 104 (2021) 034002

Holography - Workflow



DOCKER CONTAINER (May 2024)

manifest.yaml
(→ *workflows*)

CI/CD pipeline
(→ *Gitlab tests*)

+ Documentation

Input.yaml
Black hole model
polyhyperbolic/parametric

Range
$[T^{min}, T^{max}]\,, \delta T$
$[\mu^{min}, \mu^{max}]\,, \delta \mu$

Number of lines
for initial cond.

$\phi, \ \frac{\partial A^0}{\partial r}$

muses_numrelholo.exec

blackhole_model.cpp
Specify the parameters

blackhole_engineering.cpp
Calculate thermodynamics

Tmu_EoS_new.cpp
Create EoS for the phase diagram

muses_numrelholo.cpp
Main function

phi0Phi1_lines.csv
EoS directly from given initial conditions

eos.csv
EoS with uniform grid in T and μ

stable_eos.csv
EoS with only stable phases

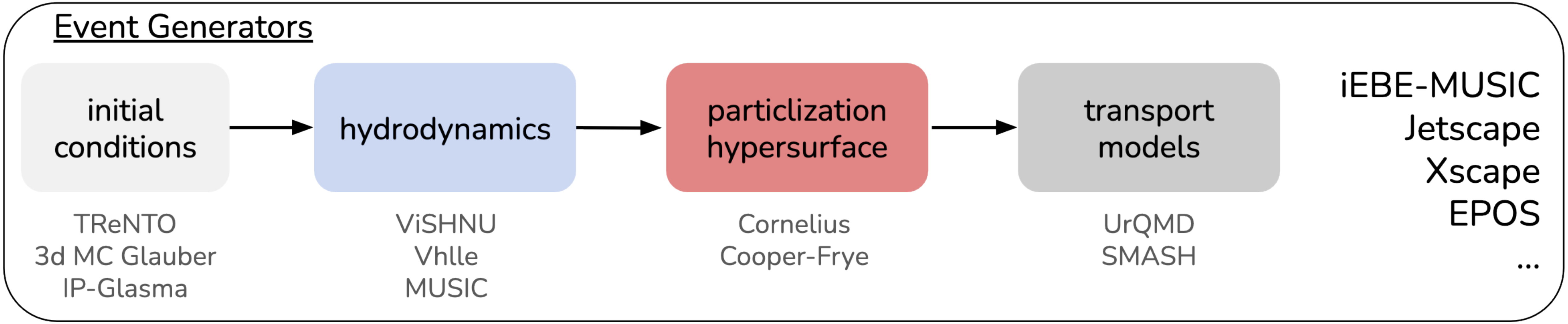
transition_line.csv
The first-order transition line

Original status

Current status

Remaining tasks

Modeling heavy-ion collisions (Gregoire)



Event Generators

initial conditions → hydrodynamics → particlization hypersurface → transport models

initial conditions	hydrodynamics	particlization hypersurface	transport models
TReNTO 3d MC Glauber IP-Glasma	ViSHNU Vhlle MUSIC	Cornelius Cooper-Frye	UrQMD SMASH

iEBE-MUSIC
Jetscape
Xscape
EPOS
...

hydrodynamics

conservation equations

$$\partial_\mu T^{\mu\nu} = 0$$

$$J^\mu_{B,Q,S} = 0$$

$$\mathrm{P}(\epsilon, n_B, n_Q, n_S)$$

transport coefficients

$$\tau_\pi = \frac{5\eta}{\epsilon + \mathrm{P}}$$

$$\tau_\Pi = \frac{\zeta}{15(1/3 - c_s^2)(\epsilon + \mathrm{P})}$$

particlization hypersurface

Momentum distribution of specie s on the hypersurface

$$f_s(\mathrm{P}, \epsilon, u, \pi, \Pi)$$

transport models

Particles taken into account should match the particles considered in EoS!

Muses?

- **Dynamic variables**: $\mathrm{P} = \mathrm{P}(\epsilon, n_B, n_Q, n_S)$

Inversion method

$(T, \mu_X) \rightarrow (\epsilon, n_X)$

- **Efficiency**: Interpolated at each spacetime point

Regular grid in

(ϵ, n_X)

- **Cutoffs**: need cutoffs for local fluctuations

Large ranges. Extrapolations?

- **Flexibility**: adapt the EoS to AB particle list

Easy access, calculation on the fly?

What do you want MUSES to do with respect to Heavy Ions? (Dekra)

- **Criticality and critical point**:
  - Implementation in multiple charge sectors. Little has been explored for multi-component hydrodynamic evolution in the presence of critical region, and it is demanding to keep up with the BES new results on cumulants ..etc
  - Control over the transition regimes for different EoS when matched. This is useful for Bayesian analysis tools.
  - Hydro input needed from the EoS goes beyond the $p(\varepsilon, \mu_q)$ to various thermodynamic derivatives. A user friendly format for the calculation of those derivative and a way to keep the list up-to-date is appreciated.

- **Extensions: HRG and low T**
  - Hydro simulation requires coverage of the low T regime, these are not available for the dense QCD regime where pion gas cannot be used.
  - A machinary to be able to always have the most up to date HRG PDG matched to the LQCD. Ideally, the user should be able to easily select a particular list of PDG for the HRG part of the EoS.

What do you want MUSES to do with respect to Heavy Ions? (Dekra)

- **Convergence and uncertainty**
  Independent convergence tests available for numerics as well as systematic errors. Systematic uncertianties are important for setting constriants on the EoS in regions where a given approach breaks down. For e.g, they allow us to relax the precision on the interpolator within some range to help extend the rootfinding search.

- **Validation**
  Fluid simulations REQUIRE thermodynamics stability. When we use MUSESE EoS, we expect these to have been perofrmed and checked :)

- **Format and efficiency**
  \- Having the EoS input in terms of densities will help avoid issues with degenrate solutions for example.
  \- A default version of the EoS available with the most tested/reliable parameters from MUSES in a format easy to handle, e.g hdf. This helps different hydro groups to compare their results with the same input.

What do you want MUSES to do with respect to Heavy Ions? (Volodya)

Equations of state at $\boldsymbol{\mu_B = 0}$

- Validation/parameter scans using lattice QCD constraints
- Output conserved charge susceptibilities (at least 2nd order, preferably to 4th order, ideally to 6th order)

Equations of state at finite temperature and density

- Main use case: using in hydro simulations of heavy-ion collisions
- Needs pressure as a function of energy and conserved charge densities; (T, μ_B) table is not enough!
  - 2D grid $p = p(\varepsilon, n_B)$ under conditions n_Q/n_B=0.4, $n_S = 0$
  - 4D grid $p = p(\varepsilon, n_B, n_Q, n_S)$
  - Transport coefficients, speed of sound

Integration (interface) of Thermal-FIST into MUSES