

Thermal-FIST package

Volodymyr Vovchenko (University of Houston)

2023 MUSES Collaboration Meeting, UIUC



May 17, 2023

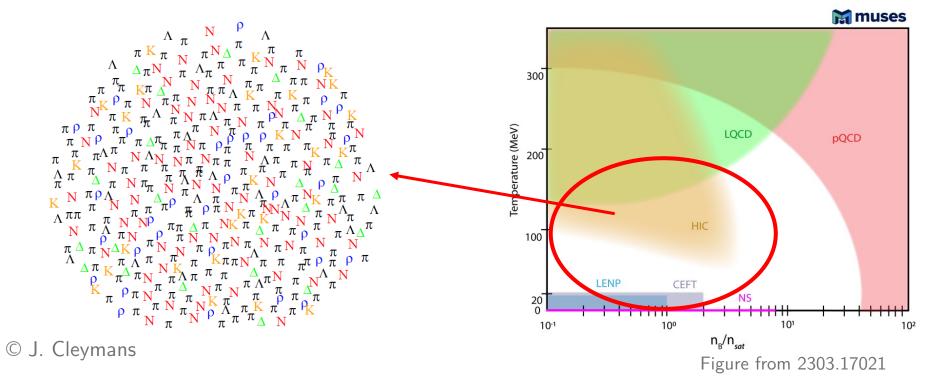


Hadron resonance gas (HRG)

HRG: Equation of state of hadronic matter as a multi-component (non-)interacting gas of known hadrons, resonances, and light nuclei

$$\ln Z \approx \sum_{i \in M,B} \ln Z_i^{id} = \sum_{i \in M,B} \frac{d_i V}{2\pi^2} \int_0^\infty \pm p^2 dp \ln \left[1 \pm \exp\left(\frac{\mu_i - E_i}{T}\right) \right]$$

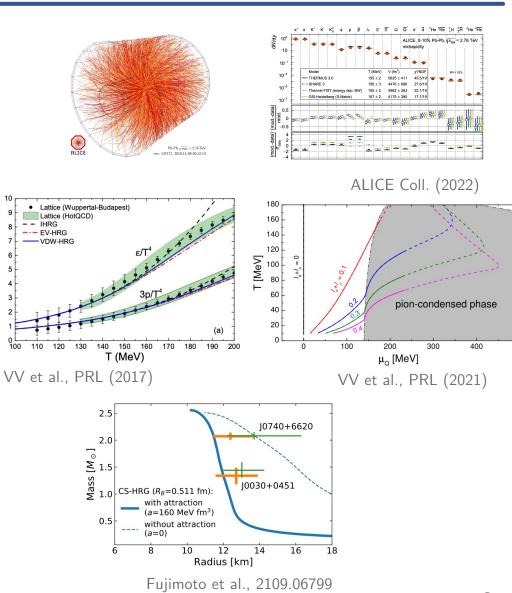
Grand-canonical ensemble: $\mu_i = b_i \mu_B + q_i \mu_Q + s_i \mu_S$ *chemical equilibrium*



HRG model applications

- Heavy-ion collisions
 - Hadrochemistry (chemical freeze-out)
 - Fluctuations of conserved charges
- Lattice QCD context
 - Understanding the degrees of freedom
 - Equation of state, susceptibilities, partial pressures
- Early universe
 - Modeling QCD contribution to cosmic EoS
 - Finite isospin density
- Neutron-star matter
 - Extending to include non-resonant interactions
 - Hadronic part of the CMF model

Natural block for MUSES



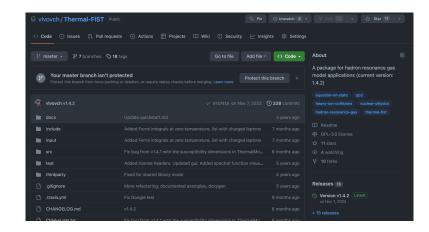
What is Thermal-FIST?



Thermal-FIST* (current version: v1.4.2) [VV, H. Stoecker]

Open-source C++ package for general-purpose HRG model analysis *Cross-platform* (Linux, Mac, Windows) through **cmake** License: GPL-3.0 GitHub: https://github.com/vlvovch/Thermal-FIST physics manual: Comput. Phys. Commun. 244, 295 (2019)

- 2014-2018: Initial development and applications (closed source)
- June 2018: First public release (v0.6)
- Jan 2019: Code documentation and CPC article release (v1.0)
- 2019-2022: Incremental upgrades
- **Soon(?):** Version 1.5 with new features (dense matter EoS, cosmology)





HRG model aspects in Thermal-FIST

- Equation of state and related properties
 - thermodynamics, hadron yields and fluctuations
- Extensions of the base HRG model
 - finite resonance widths
 - repulsive (excluded volume) and van der Waals interactions (criticality)
 - (non-)conserved charges fluctuations and correlations
 - partial chemical equilibrium
- Heavy-ion applications
 - thermal fits
 - small systems and canonical effects
 - Monte Carlo event generator
 - partial chemical equilibrium
 - light nuclei
- Other applications
 - Neutron star matter
 - Early universe (cosmic EoS)

Thermal-FIST structure

- Core library (libThermalFIST)
 - Ideal (base) HRG model (HRGBase)
 - Interacting HRG model (HRGEV/HRGVDW)
 - Partial chemical equilibrium (HRGPCE)
 - Monte Carlo mode (HRGEventGenerator)
 - Thermal fits (HRGThermalFit)
- Graphical user interface (QtThermalFIST)
 - Based on Qt5
 - Wrapper around libThermalFIST
- Sample console applications
 - Essentially just C++ macros linking to libThermalFIST

External dependencies:

- Eigen library for linear algebra (header-only, built-in)
- Minuit2 (built-in, i.e. ROOT **not needed**)
- Qt5 (for GUI only)

Using Thermal-FIST

Installation using git and cmake

```
# Clone the repository from GitHub
git clone https://github.com/vlvovch/Thermal-FIST.git
cd Thermal-FIST
# Create a build directory, configure the project with cmake
# and build with make
mkdir build
cd build
cmake ../
make
# Run the GUI frontend
./bin/QtThermalFIST
# Run the test calculations from the paper
./bin/examples/cpc1HRGTDep
./bin/examples/cpc2chi2
./bin/examples/cpc3chi2NEQ
./bin/examples/cpc4mcHRG
```

Quick start guide

Documentation

Physics manual

Using Thermal-FIST: Console mode

```
#include "HRGBase.h"
#include "HRGEV.h"
#include "HRGFit.h"
#include "HRGVDW.h"
#include "ThermalFISTConfig.h"
using namespace std;
#ifdef ThermalFIST USENAMESPACE
using namespace thermalfist;
#endif
 / Temperature dependence of HRG thermodynamics at \mu = 0
  Three variants of the HRG model:
 / 1. Ideal HRG: <config> = 0
 / 2. EV–HRG with constant radius parameter r = 0.3 fm for all hadrons (as in 1412.5478): <config> = 1
 / 3. QvdW-HRG with a and b for baryons only, fixed to nuclear ground state (as in 1609.03975): <config> = 2
  Usage: cpc1HRGTDep <config>
int main(int argc, char *argv[])
 // Particle list file
  // Here we will use the list from THERMUS-2.3, for comparing the results with THERMUS-2.3
 string listname = string(ThermalFIST_INPUT_FOLDER) + "/list/thermus23/list.dat";
 // Alternative: use the default PDG2014 list
 //string listname = string(ThermalFIST INPUT FOLDER) + "/list/PDG2014/list.dat";
 // Create the hadron list instance and read the list from file
 ThermalParticleSystem TPS(listname);
 // Which variant of the HRG model to use
  int config = 0;
```

```
if (config == 0) // Ideal HRG
  model = new ThermalModelIdeal(&TPS);
 printf("#Calculating thermodynamics at \\mu = 0 in Id-HRG model\n");
  modeltype = "Id-HRG";
else if (config == 1) // EV-HRG, r = 0.3 fm, to reproduce 1412.5478
  model = new ThermalModelEVDiagonal(&TPS);
  double rad = 0.3:
  for (int i = 0; i < model->TPS()->ComponentsNumber(); ++i)
   model->SetRadius(i, rad);
 printf("#Calculating thermodynamics at \mu = 0 in EV-HRG model with r = %lf fmn, rad);
  modeltype = "EV-HRG";
else if (config == 2) // QvdW-HRG, to reproduce 1609.03975
 model = new ThermalModelVDWFull(&TPS):
 // vdW parameters, for baryon-baryon, antibaryon-antibaryon ONLY, otherwise zero
 double a = 0.329; // In GeV*fm3
  double b = 3.42; // In fm3
```

Link to **libThermalFIST** and write a C++ macro doing whatever calculation you want

The most flexible way of using the code

git submodule is useful

MUSES use case: write a wrapper for **libThermalFIST**?

Using Thermal-FIST: Jupyter notebooks



Interactive notebooks through Jupyter (xeus kernel and ROOT-cling)*

💭 jupyte	r FitExample Last Checkpoint: несколько секунд назад (unsaved changes)	G	Logout										
File Edit	View Insert Cell Kernel Widgets Help	Trusted	C++14 O										
₽ + %	$\begin{array}{c c c c c c c c c c c c c c c c c c c $												
	Initialize and run the fitter												
In [7]:	<pre>// Set chemical potentials to zero model.SetBaryonChemicalPotential(0.0); model.SetElectricChemicalPotential(0.0); model.SetStrangenessChemicalPotential(0.0); model.SetCharmChemicalPotentials(); // Initialize the fitter ThermalModelFit fitter(&model); // Do not fit muB, it is zero at LHC fitter.SetParameterFitFlag("muB", false); // Pass the data to the fitter fitter.SetQuantities(dataPbPb010); // Perform the fit ThermalModelFitParameters fitResult = fitter.PerformFit(false);</pre>												
	Print the fitted parameters and the χ^2												
In [8]:	<pre>cout << "Extracted parameters:" << endl; cout << setw(15) << "T [MeV]" << " = " << setw(15) << 1.e3 * fitResult.T.value << " +- " << 1.e3 * fitResult. cout << setw(15) << "R [fm]" << " = " << setw(15) << fitResult.R.value << " +- " << fitResult.R.error << cout << setw(15) << "chi2/dof" << " = " << setw(15) << fitResult.chi2 << "/" << fitResult.ndf << endl;</pre>		cor <										
			•										
	Extracted parameters: T [MeV] = 155.28 +- 2.78665 R [fm] = 10.3342 +- 0.545205 chi2/dof = 15.3832/6												

*Since version 1.2.1, example at github.com/vlvovch/FIST-jupyter

Using Thermal-FIST: GUI

Graphical user interface for *general-purpose* HRG model applications

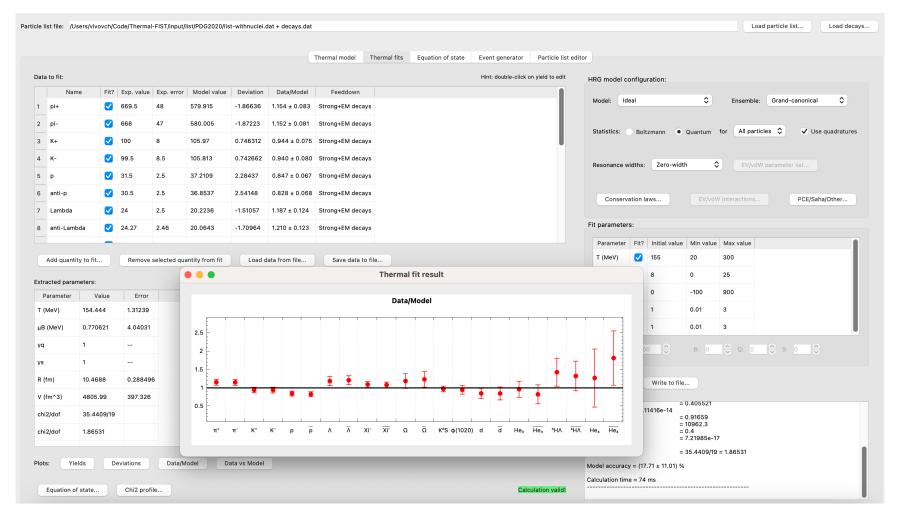
rticl	le list file: C:/I	FIST/PDG	2014/list-with	nnuclei.dat																				Load p	article list.	Load deca
The	rmal model	Thermal	fits Equat	ion of state	E	vent gener	ator Parti	icle list	editor																	
Data	a to fit:												Hint: do	uble-clic	k on yiel	d to edit	HRG	model o	configur	ation:						
	Name	Fit?	Exp. value	Exp. error	Mo	odel value	Deviation	Data	/Model	Fe	eddowr	n				^	Mod	el: Idea	al			•	Ensem	nble: G	rand-cano	nical 🔹
1	pi+	\checkmark	669.5	48	605	.439	-1.33461	1.106	± 0.079	Strong	+EM de	ecays					Stati	stics: C	Boltzn	nann (Ouant	um for	All pa	rticles	▼	se quadrature
2	pi-	668 47		47	605.474 -1.3			1.103 ± 0.078		3 Strong+EM decays																
3	K+	\checkmark	100	8	108	.722	1.09024	0.920	± 0.074	Strong	J+EM de	ecays					Resc	onance v	viatns:	евw		•				
4	к-	\checkmark	99.5	8.5	108	.657	1.07724	0.916	± 0.078	Strong	+EM de	ecays					Con	servatio	n laws	. E	v/vdW i	nteractio	ns	Othe	r options.	
5	р	\checkmark	31.5	2.5	33.4	4123	0.764917	0.943	± 0.075	Strong	I+EM de	ecays					Fit pa	aramete	rs:							
5	anti-p	\checkmark	30.5	2.5	33.2	2773	1.11091	0.917	± 0.075	Strong	+EM de	ecays					Pa	rameter	Fit?	Initia	l value	Min va	ue I	Max valı	ue	1
,	Lambda		24	2.5	19.3	3002	-1.87991	1.244	± 0.130	Strong	I+EM de	ecays				~	т (М	1eV)	\checkmark	155		20	5	00		
Add	d quantity to fi	it Rem	ove selected	quantity fro	m fit	Load dat			data to			,					R (f	m)	\checkmark	8		0	2	5		
Extra	acted paramet			· · ·													uB ((MeV)		0		-100	9	00		_
Pa	arameter	Value	Error	🌱 Ther	mal fit i	result																	-		I X	
т (М	MeV) 15	54.766	1.19547										D	ata/M	1odel											
μB ((MeV) 0.	323424	3.94532	2.5										_												-
γq	1			2.5																					ŢÌ	
γs	1			2																				т		
R (f	fm) 10).5875	0.263019	1.5																		T	Т		•	
•	,	971.27	370.495		Ŧ	Ŧ				Ŧ	Ŧ	Ŧ	Ŧ	Ŧ	Ţ					Т			•	+		
•	,	5.0616/19					<u>t</u>	£	÷					Ţ	-	£	t	Ī	ł	1	•	-	1		-	
lots			itions Data/	0.5 M															-		T			T	-	
Eau	lation of state.		profile		п+	<u>п-</u> к	(+ K-	p		۸	Ā	E-	=-	Ω	Ω	K⁰S φ	(1020)	d	đ	³ He	зНе	зн∧	зНЛ	⁴He	4He	

Thermal-FIST and HRG model equation of state

							Thermal mod	del Thermal fits	Equation of sta	ate Event gener	ator Particle list ed	ditor
	Name	PDG ID	Mass	Stable?	Neutral? B	Q	S	Prim. density	Prim. multiplicity	Total multiplicity	Scaled varian	HRG model configuration:
1	pi0	111	0.134977	•	·			0.0461954	99.0734	302.421	1.12117	Model: Quantum van der Waals 🗘 Ensemble: Grand-canonical 🗘
2	pi+	211	0.13957	•		+1		0.0456136	97.8257	264.084	1.11731	
3	pi-	-211	0.13957	•		-1		0.0456136	97.8257	264.084	1.11731	Statistics: Boltzmann
1	f(0)(500)	9000221	0.475	2 decays	•			0	0	1.12992	1	
5	К+	321	0.493677	•		+1	+1	0.0121721	26.1051	48.3981	1.01192	Resonance widths: Zero-width Zero-width EV/vdW parameter list
6	к-	-321	0.493677	•		-1	-1	0.0121721	26.1051	48.3981	1.01192	
,	anti-K0	-311	0.497611	•			-1	0.0119642	25.6592	47.0889	1.01163	Conservation laws EV/vdW interactions PCE/Saha/Other
3	ко	311	0.497611				+1	0 0119642		47 0889	1 01163	
9	eta	221	0.547862		•			Correl	ations			Parameters:
10	rho(770)-	-213	0.77526	Feeddo	wn: Final	0	Type:	Particle	0	Quantity: S	usceptibility 📀	
1	rho(770)+	213	0.77526									μ_{0} (MeV): 0.00 \updownarrow μ_{0} (MeV): 0.00 \diamondsuit μ_{s} (MeV): 0.00 \diamondsuit
12	rho(770)0	113	0.77526	pi0		0.4	pi0 120794	pi+	pi-	K+	K-	R (fm): 8.0000 🗘 R _c : 8.0000 🗘 V (fm ³): 2144.66
13	omega(782)	223	0.78265	pi+			939612	0.281041	0.107052	0.00201561	0.0123788	B: 0 🗘 Q: 0 🗘 S: 0 🗘
4	K*(892)+	323	0.8955	pi-			939612	0.107052	0.281041	0.0123788	0.00201561	
15	K*(892)-	-323	0.8955	K+			0733736	0.00201561	0.0123788	0.0469004	0.00381837	Compute fluctuations and correlations 🗸 Reset mu's
16	K*(892)0	313	0.89555	к-			0733736	0.0123788	0.00201561	0.00381837	0.0469004	Calculate Calculate from fit tab Write to file
17	anti-K*(892)0	-313	0.89555		_							ammaS = 1 = 2144.66 fm^3
18	anti-p	-2212	0.938272									rarticle density = 0.339627 fm^-3
19	p	2212	0.938272		+1	+1		0.00273362	5.86268	16.3447	0.993893	Net baryon density = 3.04932e-20 fm^-3 Net baryon number = 6.53975e-17 Net electric charge = 1.45328e-17
20	n	2112	0.939565	•	+1			0.00271554	5.82392	16.3059	0.993935	Net strangeness= 0 Absolute baryon number = 98.8937
21	anti-n	-2112	0.939565	•	-1			0.00271554	5.82392	16.3059	0.993935	E/N = 0.918198 S/N = 6.90547 S/[S] = 0

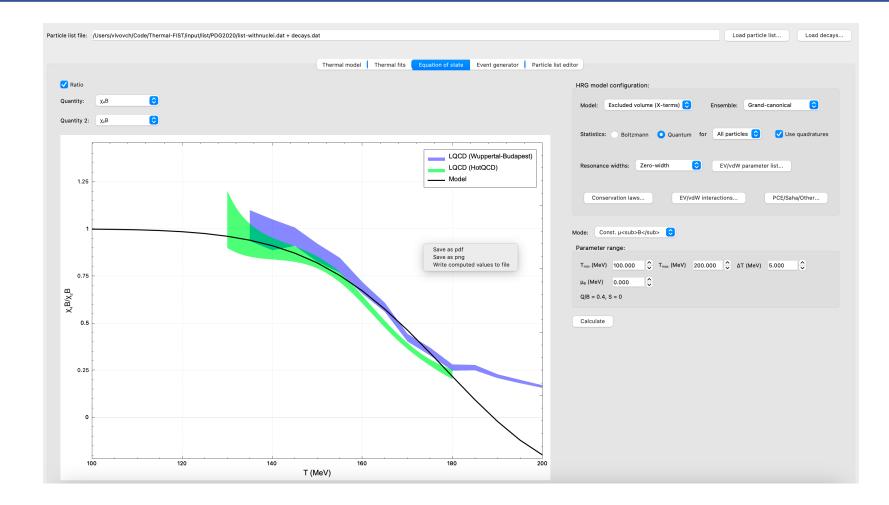
- Base calculation is at fixed T & $\mu_{B,Q,S}$ (alternatively at fixed s/n_B, Q/B, S/|S|)
- Thermodynamic functions, hadron abundances, feeddown, correlations and fluctuations

Thermal-FIST and Thermal-FITS



- Extract chemical freeze-out parameters from heavy-ion hadron abundances
- χ^2 minimization

Thermal-FIST and equation of state



- Compute HRG model quantities along a fixed T, $\mu_{\rm B}$, or $\mu_{\rm B}/T$
- Impose conservation laws [e.g. strangeness neutrality (heavy-ions) or charge neutrality (neutron stars)]

Thermal-FIST and equation of state

Console mode provides more flexibility

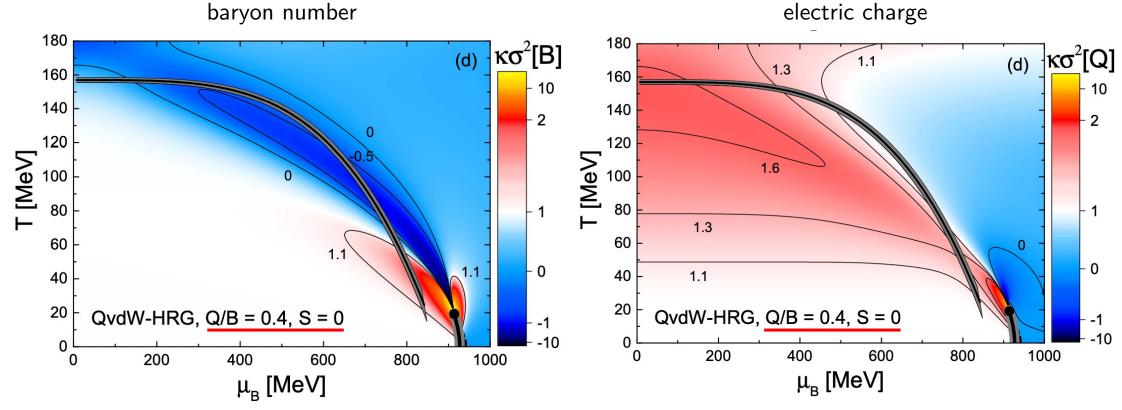


Figure from R.V. Poberezhnyuk et al., PRC 99, 024907 (2019)

Thermal-FIST and HRG event generator

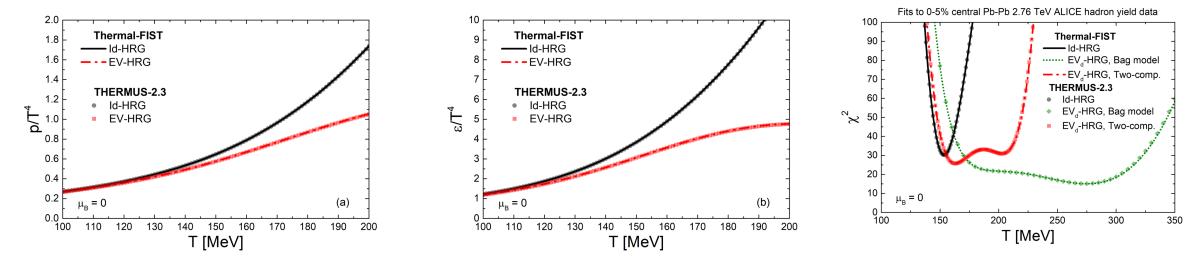
arti	cle list:						Edit par	ticle list for analysis	HRG model configuration: Model: Ideal
	Name	PDG ID	m [GeV]	Multiplicity	Variance	Scaled variance	Skewness	Kurtosis	
	pi0	111	0.134977	290.816 ± 1.652	428.352 ± 43.031	1.473 ± 0.148	-1.062 ± 5.296	-173.711 ± 236.898	Statistics: • Boltzmann • Quantum for All particles 😌 • Use quadratures
2	pi+	211	0.13957	253.070 ± 1.101	190.457 ± 20.153	0.753 ± 0.080	-0.661 ± 3.770	-43.994 ± 113.645	
	pi-	-211	0.13957	253.785 ± 1.155	209.384 ± 22.476	0.825 ± 0.089	0.894 ± 4.021	-37.564 ± 126.978	Resonance widths: Zero-width 😒 EV/vdW parameter list
	К+	321	0.493677	48.468 ± 0.493	38.110 ± 4.092	0.786 ± 0.084	0.810 ± 1.641	-6.793 ± 20.484	
	К-	-321	0.493677	47.494 ± 0.536	45.085 ± 5.403	0.949 ± 0.114	0.648 ± 2.086	12.127 ± 30.228	Conservation laws EV/vdW interactions PCE/Saha/Other
	anti-K0	-311	0.497611	46.816 ± 0.467	34.188 ± 3.277	0.730 ± 0.070	0.598 ± 1.381	-18.758 ± 15.980	
-	ко	311	0.497611	45.873 ± 0.526	43.465 ± 5.021	0.947 ± 0.109	2.185 ± 2.022	4.727 ± 31.026	Chemical freeze-out parameters:
	anti-p	-2212	0.938272	17.101 ± 0.314	15.483 ± 1.590	0.905 ± 0.093	0.696 ± 1.009	-5.153 ± 8.265	T _{ch} (MeV): <u>155.00</u> • Y _e : <u>1.0000</u> • Y _s : <u>1.0000</u> •
	р	2212	0.938272	17.139 ± 0.294	13.576 ± 1.773	0.792 ± 0.103	1.453 ± 1.443	9.415 ± 14.709	μ ₆ (MeV): 0.00
0	n	2112	0.939565	16.962 ± 0.281	12.429 ± 1.242	0.733 ± 0.073	0.539 ± 0.923	-5.257 ± 7.220	R (fm): 8.0000 🗘 R _{sc} (fm): 8.0000 🗘 V (fm ³): 2144.66
1	anti-n	-2112	0.939565	17.253 ± 0.286	12.809 ± 1.496	0.742 ± 0.087	0.718 ± 1.126	1.998 ± 9.596	B: 0 🗘 Q: 0 🗘 S: 0 🗘
									Blast-wave momentum spectrum:
str	ibution: dN/dy	6			0			Binning	Spherically symmetric Cracow model T _{BW} (MeV): 155.00
dN/dy	150 125 100 75								Perform decays Events: 10000 Generate
	25			ŧ.					Generated 158 events Effective event number = 158 CE acceptance rate: 2.30457e-05

- Monte Carlo sampling of hadron abundances, momenta, and coordinates
- Superimposed on blast-wave flow velocity profile
- Realistic modeling of acceptance effects, especially for correlations and fluctuations

Thermal-FIST in THERMUS mode: cross-check

THERMUS* is an early open-source implementation of some HRG model features [S. Wheaton, J. Cleymans, B. Hippolyte, et al.]

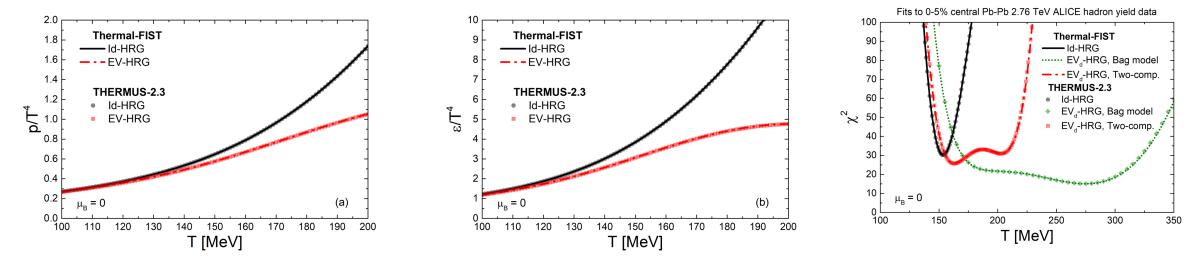
Use exactly the same input (particle list, finite widths, and excluded volume parameters) and compare



Thermal-FIST in THERMUS mode: cross-check

THERMUS* is an early open-source implementation of some HRG model features [S. Wheaton, J. Cleymans, B. Hippolyte, et al.]

Use exactly the same input (particle list, finite widths, and excluded volume parameters) and compare



FIST: Fist IS Thermus

Rigorous unit testing still to be implemented

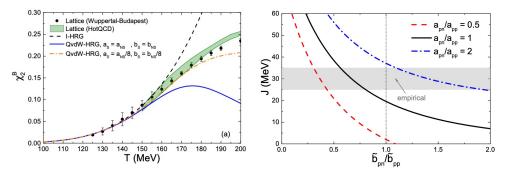
*https://github.com/thermus-project/THERMUS

Interacting HRG can be more smoothly connected to other QCD phases than ideal HRG e.g. the crossover in *T* direction, Albright, Kapusta, Young, PRC (2014)

Thermal-FIST incorporates van der Waals interactions in the most general form

$$p(T, n_1, \ldots, n_h) = \sum_i \frac{T n_i}{1 - \sum_j \tilde{b}_{ji} n_j} - \sum_{i,j} a_{ij} n_i n_j$$

- Separate excluded volume b_{ij} for each pair i,j of species
- Separate mean field a_{ii} for each pair i,j of species
- So far very little explored!
- E.g. indications for flavor-dependent parameters from
 - Lattice QCD susceptibilities [Karthein et al., 2107.00588]
 - Symmetry energy
 - Neutron-star matter EoS [Fujimoto et al., 2109.06799]



VV, Motornenko, Alba, et al., 1707.09215

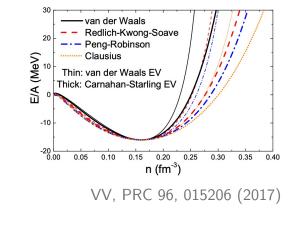
Interactions in HRG: Beyond van der Waals

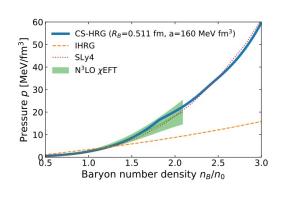
Standard van der Waals gives too stiff EoS beyond the saturation density

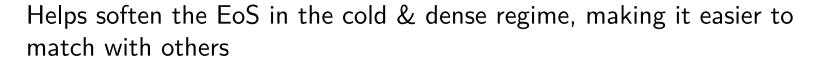
Beyond vdW:

- Generalized (non-linear) excluded volume
 - Carnahan-Starling (CS)
- Density-dependent mean-field
 - Real gases
 - Skyrme
 - VDF model

A. Sorensen, V. Koch, PRC 104, 034904 (2021)

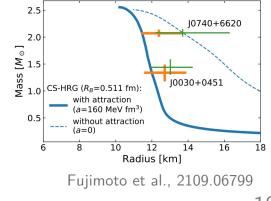






Available out-of-the-box in FIST in next version (already present in **devel** branch on github), adding leptons into the list one can do neutron-star matter

Another extension: pion interactions and condensation at finite isospin density VV et al., PRL (2021)



Thermal-FIST & MUSES: Summary

- Thermal-FIST is an open-source implementation of the HRG model equation of state with many knobs
 - Particle lists, interaction parameters, and other settings easily customizable
 - Provides EoS properties (averages as well as susceptibilities) at given T & $\mu_{B,Q,S}$
 - Works both under heavy-ion and neutron star regimes
- Standalone C++ implementation with minimal external dependencies
 - Only the base library **libThermalFIST** really needed to be built and linked against
 - Integration into MUSES with a wrapper?
- Interaction parameters still need to be constrained

Thanks for your attention!