viscous Ultrarelativistic Smoothed Particle hydrodynamics

V-USPhydro Manual

Dr. Jacquelyn Michelle Noronha-Hostler

Original Authors: Jacquelyn Noronha-Hostler, Gabriel Denicol, Jorge Noronha, Rone P. G. Andrade, and Frederique Grassi

Universidade de São Paulo

São Paulo, 8th November, 2020

I. INTRODUCTION

v-USPhydro is a viscous relativistic 2+1 hydrodynamical code that uses the Langragian (Smooth Particle Hydrodynamics, SPH) method to solve the equations of motion. In this version there is infrastructure to write the 3+1 version but it is not yet set up. The Cooper Frye freeze-out and decays are found either in subfolders df (analytical solutions, no decays) or sampling (Monte Carlo sampling+decays). The respective flow harmonics are calculated within each routine. The main papers for the code are [1] and all equations of motion and in depth details can be found there. When writing papers about v-USPhydro make sure to check Sec. II to know what papers to cite.

II. WHAT TO CITE

- Must cite these every time the code is used [1]
- Reasoning behind low switching temperature [2]
- If you use a temperature dependent η/s or ζ/s cite this for the low temperature region of η/s and when discussing the peak of the bulk at T_c [3], the high temperature region is from lattice data [4], the paramaterization is done is [5], the shear relaxation time is from [6, 7], the bulk is inspired by Buchel's formula for a strongly coupled plasma [8], and the bulk relaxation time is from [9]
- Other papers that investigate bulk viscosity in relativistic hydrodynamics [10, 12–15, 29]
- Papers studying the effects of η/s alone [16]
- Papers studying the consequence of large ζ/s (besides our own): [17–19]
- Various types of δf: moments [1, 26, 27], Hirano [12], Dusling [15], η_i/η corrections [28], and Bozek [29]
- Original SPH citations [20–22]
- For the MC Glauber and mckln (CGC) initial condtions [23] and NeXus [24]
- The Equation of state is EOS S95n-v1 from [25] (for lower temperatures, i.e., $T < 50$ MeV this equation of state is matched to that of a massive gas of pions).
- When you use UrQMD you must cite [30]. There is a citation from Chun Shen that will hopefully be out soon that is also needed.

Work done with `VISHNU`+UrQMD (2+1 code) that is good to read both for understanding and good to cite is:

- Fitting RHIC spectra [31], switching temperature/Equation of State [32] and flow harmonics [33].
- The effect of various number of resonance decays on the final flow harmonics [34] (note this is NOT with UrQMD)
- Spectra and elliptical flow for LHC and RHIC [35, 36]

III. INPUT FILE

Almost all necessary parameters can be defined within your input file. There are a handful of exceptions that must be edited in the code directly (and they will be covered in their own sections later on in this manual. It's also important to note that the input file must be in the order listed here, otherwise it will not run correctly.

The input can be called whatever you like, however, if you want to do multiple runs you MUST include a number after the name i.e. input2.dat and name the run and corresponding `dfinputfilename` with the same number (see Sec. IV).

The input file should have the following structure:

```
1   h:  # dt:  #
2   equationsofmotion:  ideal/bulk/shear+bulk EOS: type+files
3   low:  on/off
4   viscfac_bulk_shear:  # #
5   t0:  #
6   freezeoutT: #
7   eventbyevent/giorgio/gubser #
8   ICs:  foldername
9   output_folder:  foldername
10  # #
11  freezeout:  0/1/2 (if 1 see Sec. III C and if 2 see Sec. III B)
```

Line 1: Define h (the SPH smoothing parameter), for MC Glauber h=0.3 is reasonable. Define dt (the time step), dt=0.1 is normally reasonable unless doing the Gubser test. Note that the relaxation time should NOT be smaller than dt.

Line 2-3: Define the equation of motion (either ideal, bulk or shear+bulk). If you want to run shear only (no bulk), use shear+bulk and then set the bulk constant to zero on Line 4. Define the Equation of State type either ideal or table. If you use ideal then Line 3 should be set to off. If you use table then you must list the tables used (see Sec. VIII for the format of the files) and Line 3 should be set to off. An example of this would be:

```
12  equationsofmotion:  ideal EOS: table
13  temp.dat
14  derv.dat
15  low:  on
```

Line 4: define the multiplying factor times ζ/s and η/s, respectively. The form of the function can be changed in the code itself (see Sec. IX but a multiplying constant can still be included in front of any form to run checks.

Line 5: Initial time, usually $\tau_0 = 1$ for RHIC and $\tau_0 = 0.6$ for LHC (in fm).

Line 6: Define the freezeout temperature (in MeV)

Line 7: Define the type of initial conditions and then the multiplying factor (if no factor is needed simply write 1) e.g.

```
16  eventbyevent 1
```

Line 8: Define the folder where the initial conditions are found (all initial conditions should be in the folder inputfiles so you do not need to include that in the definition) e.g.

```
17  ICs:  rhic
```

Line 9: Define the output folder. Often this has the same name as the input folder.

Line 10: If you defined the initial conditions as eventbyevent or giorgio then this line should have the number of the first event and the last event

Line 11: If you only want to run hydro then freezeout=0. If you want to run with the analytical cooper frye solution freezeout=1 and see Sec. III C for further definitions. If you want to run with Monte Carlo Sampling then freezeout=2 and see Sec. III B for further definitions

A. Exceptions

The following changes must be done to the code itself:

- Change in the form of ζ/s and η/s (see Sec. IX)
- Gubser and TECHQM tests (see Sec. IX and Sec. XIV)
- Output format (see Sec. X)
- Adding extra terms to Equation of motion (see Sec. VII)

B. v-USPhydro+MC Sampling+UrQMD

If you want to run v-USPhydro+MC Sampling+UrQMD you must set:

```
18      freezeout:  2
```

and then include the following lines afterwards (all parameters listed here are an example):

```
19      dfinputfilename:  vupshydroinput2.dat
20      output_folder:  osc2u/ebe/
21      df_cor_file:  input/flist150.dat
22      range(ptmax,ptstepsize,phisteps):  3.0 0.1 10
23      List_of_hadrons:  input/resoweak.dat
24      corrections:  1
25      number_of_monte_carlo_loops:  100
26      output_particles_folder:  output_particles.dat
27      decays:  1
```

Line 19: Name of the input file for the sampling code (useful to use different names if doing multiple runs at once so there is no overlap). Make sure the number here matches that of the run #.

Line 20: Name of the output folder, usually, using the following is sufficient:

```
28      output_folder:  osc2u/ebe/
```

Line 21: File where the list of E0, D0, and B0 are found for the δf correction for bulk. For the moments method at a freezeout temperature of $T = 150$ MeV, one can use:

```
29      df_cor_file:  input/flist150.dat
```

Line 22: Range for the pT and phi. We assume that pTmin=0. The numbers are pTmax, pT step size, and number of phi steps (NOT dphi but the total number of steps)

Line 23: List of hadrons needed. As long as you're not running tests, using the following should be sufficient:

```
30      List_of_hadrons:  input/resoweak.dat
```

Line 24: corrections=1 if you want the inclusion of δf corrections for viscosity and corrections=0 if you want to ignore viscous corrections to the distribution function

Line 25: Number of Monte Carlo loops

Line 26: File (within the subfolder flow) that lists the number of hadrons to be watched e.g pions, protons etc. This is for the final spectra only.

Line 27: Decays=0 if you just want to do the MC sampling, Decays=1 if you only want the spectra after decays, Decays=2 if you want the spectra both before and after decays.

If you would like to run the output for both MC Sampling with and without decay change the last line to:

```
31      decays:  2
```

C. v-USPhydro+analytical Cooper Frye

If you want to run v-USPhydro+analytical Cooper Frye you must set:

```
32      freezeout:  1
```

and then include the following lines afterwards:

```
33      dfinputfilename:  vupshydroinput1.dat
34      df_cor_file:  input/flist150.dat
35      range(ptmax,ptstepsize,phisteps):  5.0 0.1
36      Number_of_hadrons:  1
37      0.13957 -1 1
```

Line 33: Name of the input file for the sampling code (useful to use different names if doing multiple runs at once so there is no overlap). Make sure the number here matches that of the run #.

Line 34: File where the list of E0, D0, and B0 are found for the δf correction for bulk. For the moments method at a freezeout temperature of $T = 150$ MeV, one can use:

```
38      df_cor_file:  input/flist150.dat
```

Line 36: Range for the pT (phi is done automatically). We assume that pTmin=0. The numbers are pTmax and pT step size.

Line 37: List of hadrons needed. The number of hadrons should be at least 1 or great. Then the following lines include the list (each hadron/line) in the order of mass, boson/fermion, degeneracy

IV. RUNNING THE CODE

There are various ways to run v-USPhydro. The easiest way to run both the hydro combined with decays. However, each separate part of the code can be run separately too. For this section, I will assume you have already generated initial conditions (otherwise see Sec. VI).

Additionally, the only requirement for running v-USPhydro is having c++ and gfortran installed (the former is needed for UrQMD). No other files or programs are necessary.

A. v-USPhydro+Analytical Cooper Frye or UrQMD

- Ensure that your input file (see Sec. IV) is properly set up.

-
  ```
  39      cd v-USPhydro
  40      ./all.sh run#
  ```

  where run# is the number of your run with a corresponding input file with that number see Sec. III

- wait for the results :)

B. v-USPhydro only

There are a few different options. If you want to define all parameters in the inputfile ”manualfile2.dat” all you need to do is run:

- Ensure that your input file (see Sec. IV) is properly set up.

-
  ```
  cd v-USPhydro
  make
  ./vusphydro
  ```

If you would like to specify a different input file:

- Ensure that your input file (see Sec. IV) is properly set up.

-
  ```
  cd v-USPhydro
  make
  ./vusphydro inputfilename run#
  ```

  where run# is the number of your run with a corresponding input file with that number see Sec. III

If you would like to specify a different input file and event range

- Ensure that your input file (see Sec. IV) is properly set up.

-
  ```
  cd v-USPhydro
  make
  ./vusphydro inputfilename run# event_start event_end
  ```

  e.g.

  ```
  ./vusphydro input2.dat 2 1 150
  ```

  where run# is the number of your run with a corresponding input file with that number see Sec. III

C. analytical Cooper Frye

There are a few different options. If you want to define all parameters in the inputfile ”input.dat” all you need to do is run:

- Ensure that your input file (see Sec. IV) is properly set up.

-
  ```
  cd v-USPhydro/df
  make
  ./fo
  ```

If you would like to specify a different input file:

- Ensure that your input file (see Sec. IV) is properly set up.

-
```
cd v-USPhydro/df
make
./fo inputfilename
```

Not, there is no need to specify a `run#` with the analytical cooper frye.

If you would like to specify a different input file and event range

- Ensure that your input file (see Sec. IV) is properly set up.

-
```
cd v-USPhydro/df
make
./fo inputfilename event_start event_end
```

  e.g.

```
./fo input.dat 1 150
```

D. Monte Carlo Sampling+UrQMD

There are a few different options. The easiest way to run Monte Carlo Sampling+UrQMD is using `ebe.sh` and by defining a run# with a corresponding inputfile. For instance, you can run

```
./ebe.sh 3
```

where 3 is your run# and within `ebe.sh` you have your inputfile defined e.g. line 4 within `ebe.sh` could read

```
./fo vusphydroinput$1.dat $1
```

where `vusphydroinput$1.dat` is the name of your input file (`$1` is the variable run# so in our example above the input file would be `vusphydroinput3.dat`).

You can also run a specific number of events by changing your `ebe.sh` file on line 4 to

```
./fo vusphydroinput$1.dat $1 event_start event_end
```

If you want to run only the sampling and no decays, you can specify that in your input file, see Sec. III.

V. STRUCTURE OF THE CODE

This is the main file of v-USPhydro. The files are organized as follows:

- EOS- eos.cpp/eos.h but tables to read them in are in tables.h and eostables.h and files to read them in happen within enetric.h/enetric.cpp. The EOS subroutines are meant to be a black box in the sense that you can change the subroutin directly and it will be consistently affected across the entire code

- Hydro evolution equations- primarily hydrosim.cpp/hydrosim.h but parts are found withint subroutines of Linklist.h (specifically the quantities that are affected by the link list) and particle.h

- Initial conditions- are processed and turned into SPH particles within enetric.h/enetric.cpp but some parts are processed within Linklist.h

- Math framework for vectors and matrices: vector.h/vector.cpp and matrix.h/matrix.cpp
- Integration subroutines- rungekutta.h
- Linklist- Linklist.h
- Calculation of freezeout hypersurface - particles are indentified in hydrosim.cpp but normal vectors are calculated in Linklist.h
- output- output.h

VI. INITIAL CONDITIONS

See attached ics.ps for instructions.

VII. RELATIVISTIC HYDRODYNAMICS

See attached pdf's entitled General Metric etc.

VIII. EQUATION OF STATE

We use two different tables to read in for the EOS. temp.dat lists the temperature (in MeV) and two columns of zeros (would be used if there is a non-zero chemical potential (currently there is no infrastructure for $\mu_b \neq 0$ in the code, though). derv.dat lists the corresponding energy density (in GeV/fm^3), pressure (in GeV/fm^3), entropy density (in $1/fm^3$), and $\frac{dw}{ds}$ (where $w = e + p$) for each line of temperature in temp.dat. In both files the total number of table lines is listed at the top. A spline function is already implemented to allow for smooth derivatives of the table.

IX. VISCOSITY

The form of the viscosity can be changed within the file particle.h, specifically the subroutine:

```
template <int D>
void Particle<D>::setvisc(double bvf, double svf, int type)
```

where bvf is the constant to multiple the form of ζ/s and svf is the multiplying constant for η/s.

Edit type=1 for the bulk equation ONLY or type=3 for shear and/or shear+bulk. The variables are then defined as follows:
zeta=ζ
tauRelax=τ_Π
setas=η
stauRelax=τ_π

Note that the relaxation time CANNOT be smaller than dt, thus, a cutoff is made to ensure that both relaxation times are large enough.

X. OUTPUT

A. v-USPhydro

All output formats are within output.h. However, they must be uncommented with the file hydrosim.cpp, otherwise they will not be called. For instance, calling

```
out.eprofile(linklist);
```

within hydrosim.cpp will print off the energy density profile. However, it is usually left commented out to save space and time.

Other possibilities that can be used but are usually commented out are:

The energy conservation

```
out.conservation(linklist);
```

And the gubser test comparison

```
out.gubcheckux(linklist); // out.gubcheckuy(linklist);
```

Note that some of these may have specific versions for viscous Equations of motion (but they are already included in the code and you simply need to uncomment them to use them.

B. flow harmonics

Results for the spectra and flow harmonics will be located in folders `v-USPhydro/sampling/flow/out/foldername/decays` and `v-USPhydro/sampling/flow/out/foldername/nodecays` when sampling is used, otherwise they are listed in `v-USPhydro/df/out/foldername`. The user defines `foldername`. Each folder contains the spectra (in `ev#_dNdphidpp.dat` is listed), the $v_n(p_T)$ (`vn_ev#.dat`) and integrated v_n's (`vnintlistrun#.dat`) such that for event=7 and run=3 they would be `ev7_dNdphidpp.dat`, `vn_ev7.dat`, and `vnintlistrun3.dat`. The integrated vn file contains the results of all the events for that run.

The spectra format is:

`hadron#` p_T `[GeV]` ϕ $\frac{dN}{dp_T d\phi}$

The $v_n(p_T)$ format is (and the first line lists this also):

`hadron#` p_T `[GeV]` ϕ $\frac{dN}{dp_T}$ v_1 v_2 v_3 v_4 v_5

The integrated v_n format is:

`hadron#` v_1 ψ_1 v_2 ψ_2 v_3 ψ_3 v_4 ψ_4 v_5 ψ_5

Note: to compare to most experimental data you may need to calculate $\frac{dN}{2\pi p_T dp_T d\phi}$ to compare with experimental data.

Also, if you are using the analytical Cooper Frye then the integrated v_n's are listed `foldernameint_typec.dat` where `foldername` is defined by the user, `type` is either ideal, bulk or shearbulk, and c is present for corrected δf and not present if no viscous corrections are included.

XI. ANALYTICAL COOPER FRYE

The derivations of the equations are shown already in [1] and will not be repeated here. This program calculations the spectra only from the distribution function and includes no decays. It is located in the subfolder df. Input files should all be stored in the folder input and output files should be stored in the folder out. This code calculations both the spectra and the flow harmonics. It is automatically set to output v_1 through v_5 and both the p_T dependent flow harmonics and the integrated v_n's.

The integrated v_n's are given as a list with their file name on it. The p_T dependent flow harmonics are outputed on an event-by-event basis. There is a small code within the subfolder out that can be run to average over all the events for each centrality class/type of equation of motion. Simply define the parameters in the file:

```
out/in.dat
```

and then run the code:

```
    c++ main.cpp
    ./a.out
```

Make sure that you have a folder named average already created in the respective folder, otherwise the program will not run.

As for the input paramters your file `in.dat` should look like

```
    visc:  bulk
    folder:  eventbyevent_glauber/30
    centrality:  30
    events:  1 150
    integrate:  0
    bulkcorrection:  0
    ptsize:  0 3 0.1
```

where the `visc:` is your viscosity, `folder:` is the folder where the spectra are stored in, `centrality:` is your centrality class, `events:` are the number of events to average over, `integrate:` can be ignored (and will be removed eventually), `bulkcorrection:` is if you want the corrected or uncorrected δf (with corrections =1, otherwise 0), and `ptsize:` is the minimum p_T, max p_T, and the step size.

It is wise to keep the individual spectra for each event because it is much quicker to calculate the flow harmonics from the already calculated spectra rather than recalculating everything. There is a small program to do so in section Sec. XIII.

XII. MONTE CARLO SAMPLING+ URQMD

Within the subfolder sampling there are various other programs that are all needed to run MC Sampling+UrQMD. The folders are the following:

osc2u: osc2u is a Fortran code that coverts the results of our Monte Carlo sampling into a format that can be read into UrQMD. Note that even osc2u requires a very specific format to run, so it would not be adviseable to change the output format from the main MC sampling code. Furthermore, this is almost exactly the same code that is given out from S. Bass, however, I have changed the output from the double fortran format (i.e. 10D05) into the exponential format (i.e. 10E05) to make it readable by C++.

urqmd: urqmd runs the hadronic interactions. You can choose what hadrons you would like to feed into the code but clearly, the more hadron you use, the more accurate it will be. Furthermore, this is almost exactly the same code that is given out from S. Bass, however, I have changed the output from the double fortran format (i.e. 10D05) into the exponential format (i.e. 10E05) to make it readable by C++. As far as I know, this version of UrQMD is the only one compatible with hooking up to a hydro code. If you switch between a 32 bit and 64 bit computer you may need to delete all the object files an executables and rerun the make file.

flow: flow reads in the output from UrQMD and puts all the Monte Carlo events into a histogram to calculation the spectra and the following flow harmonics. Additionally, it can do the same for the MC sampling even without urqmd. The first is saved in the folder decays and the second is in the folder no decays (for each respective output folder). As with df the code calculates both the p_T dependent flow harmonics and the integrated v_n's. Furthermore, you would still need to average over the events to obtain the average flow harmonics.

In the folder sampling itself is the MC sampling program that computes the spectra on an event-by-event basis and lets you define the number of Monte Carlo loops. The input files can be found in the folder input. It is easiest to run the program using the ebe.sh executable (making sure that you have the correct input file defined within it) since there are multiple program that need to be run before obtaining the end results.

To average over the events you can simply use the same code as described in the Analytical Cooper Frye (Sec. XI)

It is wise to keep the individual spectra for each event because it is much quicker to calculate the flow harmonics from the already calculated spectra rather than recalculating everything. There is a small program to do so in section Sec. XIII.

XIII. FLOW HARMONICS FROM THE SPECTRA

Within the file `v-USPhydro/fromspec` you can calculate the flow harmonics from the spectra themselves. This is the quickest way to handle changes to how you want to calculate the flow harmonics (if you want to change the ψ_n, for instance). The program itself reads in the various spectra (you must include the files in a folder, which you define within the file `input.dat`).

XIV. TESTING

A. Gubser Test

B. TECHQM

XV. TROUBLESHOOTING

- Error that mkdir can't make a folder. This is because the terminal does not like to create a folder and subfolder at the same time. If you want to output your files into something like rhic/ideal/50 then either create the folders yourself ahead of running the program or store your data in just one folder at a time (as in run first for rhic, then rhic/ideal, and finally rhic/ideal/50)
- Error involving object files/structure. Most likely you ran the code first on either a 32 or 64 bit computer and then switched to the other. You need to delete all object files and executables and compile the program again.
- Memory allocation problem. Run the executable using valgrind (i.e. valgrind ./a.out) and it will show you where your memory allocation problems are showing up.
- Segmentation fault. Either you edited the code and there is an error or there are some problems with your input file. Some common problems are: too wide of a range of events for initial conditions that you don't actually have or not putting the number of SPH grid points at the top of an initial condition file.
- Can't open list.dat (or whatever your list is called). This is usually because the folders weren't created correctly, sometimes rerunning everything fixes the problem.
- If you want to run multiple types of viscosity at one time (i.e. ideal, bulk, shear+bulk etc) then it is best to run each in a separate folder. You can use the same initial conditions, but then make sure the output folder is set to different types of folders.

[1] J. Noronha-Hostler, J. Noronha and F. Grassi, arXiv:1406.3333 [nucl-th]. J. Noronha-Hostler, G. S. Denicol, J. Noronha, R. P. G. Andrade and F. Grassi, Phys. Rev. C **88**, 044916 (2013) [arXiv:1305.1981 [nucl-th]].

[2] J. Noronha-Hostler, J. Noronha, G. S. Denicol, R. P. G. Andrade, F. Grassi and C. Greiner, Phys. Rev. C **89**, 054904 (2014) [arXiv:1302.7038 [nucl-th]].

[3] J. Noronha-Hostler, J. Noronha and C. Greiner, Phys. Rev. Lett. **103**, 172302 (2009). J. Noronha-Hostler, J. Noronha and C. Greiner, Phys. Rev. C **86**, 024913 (2012).

[4] A. Nakamura and S. Sakai, Phys. Rev. Lett. **94**, 072305 (2005) [hep-lat/0406009].

[5] H. Niemi, G. S. Denicol, P. Huovinen, E. Molnar and D. H. Rischke, Phys. Rev. C **86**, 014909 (2012) [arXiv:1203.2452 [nucl-th]].

[6] G. S. Denicol, T. Koide and D. H. Rischke, Phys. Rev. Lett. **105**, 162501 (2010) [arXiv:1004.5013 [nucl-th]].

[7] G. S. Denicol, J. Noronha, H. Niemi and D. H. Rischke, Phys. Rev. D **83**, 074019 (2011) [arXiv:1102.4780 [hep-th]].

[8] A. Buchel, Phys. Lett. B **663**, 286 (2008).

[9] X. -G. Huang, T. Kodama, T. Koide and D. H. Rischke, Phys. Rev. C **83**, 024906 (2011).

[10] H. Song and U. WHeinz, Phys. Rev. C **81**, 024905 (2010) [arXiv:0909.1549 [nucl-th]].

[11] P. Bozek, Phys. Rev. C **81**, 034909 (2010) [arXiv:0911.2397 [nucl-th]].

[12] A. Monnai, T. Hirano, Phys. Rev. C **80**, 054906 (2009).

[13] G. S. Denicol, T. Kodama, T. Koide and P. .Mota, Phys. Rev. C **80**, 064901 (2009) [arXiv:0903.3595 [hep-ph]].

[14] G. S. Denicol, T. Kodama and T. Koide, J. Phys. G **37**, 094040 (2010) [arXiv:1002.2394 [nucl-th]].

[15] K. Dusling and T. Schäfer, Phys. Rev. C **85**, 044909 (2012) [arXiv:1109.5181 [hep-ph]].

[16] P. Romatschke and U. Romatschke, Phys. Rev. Lett. **99**, 172301 (2007); ; H. Song, S. A. Bass, U. Heinz, T. Hirano and C. Shen, Phys. Rev. Lett. **106**, 192301 (2011) [Erratum-ibid. **109**, 139904 (2012)] [arXiv:1011.2783 [nucl-th]]; H. Song, S. A. Bass and U. Heinz, Phys. Rev. C **83**, 054912 (2011) [Erratum-ibid. C **87**, 019902 (2013)] [arXiv:1103.2380 [nucl-th]]; H. Niemi, G. S. Denicol, P. Huovinen, E. Molnar and D. H. Rischke, Phys. Rev. Lett. **106**, 212302 (2011); C. Gale, S. Jeon, B. Schenke, P. Tribedy and R. Venugopalan, Phys. Rev. Lett. **110**, 012302 (2013) [arXiv:1209.6330 [nucl-th]]; B. Schenke, S. Jeon and C. Gale, Phys. Rev. Lett. **106**, 042301 (2011) [arXiv:1009.3244 [hep-ph]].

[17] G. Torrieri, B. Tomasik and I. Mishustin, Phys. Rev. C **77**, 034903 (2008) [arXiv:0707.4405 [nucl-th]], G. Torrieri and I. Mishustin, Phys. Rev. C **78**, 021901 (2008) [arXiv:0805.0442 [hep-ph]].

[18] K. Rajagopal and N. Tripuraneni, JHEP **1003**, 018 (2010) [arXiv:0908.1785 [hep-ph]].

[19] M. Habich and P. Romatschke, arXiv:1405.1978 [hep-ph].

[20] L. B. Lucy, ApJ. **82**, 1013 (1977); R. A. Gingold and J. J. Monaghan, MNRAS **181**, 375 (1977); R. A. Gingold and J. J. Monaghan, Mon. Not. Roy. Astron. Soc. **181**, 375 (1977).

[21] J. J. Monaghan, Annu. Rev. Astron. Astrophys. **30**, 543 (1992); E. Chow and J. J. Monaghan, J. Comput. Phys. **134**, 296 (1997); P. Laguna, W. A. Miller and W. H. Zurek, Phys. Rev. D **41**, 451 (1990); R. J. Thacker, E. R. Tittley, F. R. Pearce, H. M. P. Couchman and P. A. Thomas, Mon. Not. Roy. Astron. Soc. **319**, 619 (2000).

[22] C. E. Aguiar, T. Kodama, T. Osada and Y. Hama, J. Phys. G **27**, 75 (2001).

[23] H. -J. Drescher and Y. Nara, Phys. Rev. C **75**, 034905 (2007); Phys. Rev. C **76**, 041903 (2007).

[24] H. J. Drescher, M. Hladik, S. Ostapchenko, T. Pierog and K. Werner, Phys. Rept. **350**, 93 (2001) [hep-ph/0007198].

[25] P. Huovinen and P. Petreczky, Nucl. Phys. A **837**, 26 (2010) [arXiv:0912.2541 [hep-ph]].

[26] G. S. Denicol, H. Niemi, E. Molnar and D. H. Rischke, Phys. Rev. D **85**, 114047 (2012).

[27] G. S. Denicol and H. Niemi, Nucl. Phys. A **904-905**, 369c (2013).

[28] D. Molnar and Z. Wolff, arXiv:1404.7850 [nucl-th].

[29] P. Bozek, Phys. Rev. C **81**, 034909 (2010) [arXiv:0911.2397 [nucl-th]].

[30] S. A. Bass, M. Belkacem, M. Bleicher, M. Brandstetter, L. Bravina, C. Ernst, L. Gerland and M. Hofmann *et al.*, Prog. Part. Nucl. Phys. **41**, 255 (1998); H. Petersen, J. Steinheimer, G. Burau, M. Bleicher and H. Stocker, Phys. Rev. C **78**, 044901 (2008); M. Bleicher, E. Zabrodin, C. Spieles, S. A. Bass, C. Ernst, S. Soff, L. Bravina and M. Belkacem *et al.*, J. Phys. G **25**, 1859 (1999).

[31] H. Song, S. A. Bass, U. Heinz, T. Hirano and C. Shen, Phys. Rev. C **83**, 054910 (2011) [Erratum-ibid. C **86**, 059903 (2012)] [arXiv:1101.4638 [nucl-th]].

[32] H. Song, S. A. Bass and U. Heinz, Phys. Rev. C **83**, 024912 (2011) [arXiv:1012.0555 [nucl-th]].

[33] H. Song, S. A. Bass, U. Heinz, T. Hirano and C. Shen, Phys. Rev. Lett. **106**, 192301 (2011) [Erratum-ibid. **109**, 139904 (2012)] [arXiv:1011.2783 [nucl-th]].

[34] Z. Qiu, C. Shen and U. W. Heinz, Phys. Rev. C **86**, 064906 (2012) [arXiv:1210.7010 [nucl-th]].

[35] H. Song, S. A. Bass and U. Heinz, Phys. Rev. C **83**, 054912 (2011) [Erratum-ibid. C **87**, 019902 (2013)] [arXiv:1103.2380 [nucl-th]].

[36] H. Song, S. Bass and U. W. Heinz, Phys. Rev. C **89**, 034919 (2014) [arXiv:1311.0157 [nucl-th]].