Main Function

This is a summary describing the sequence of functions and class objects run in **main.cpp**

1. Parameters

The parameters in the file **parameters.dat** are stored in the **ParameterReader** class object ***paraRdr**. For more information see **ParameterReader.h**

The **ParameterReader** class functions, along with some tools in **arsenal.cpp**, read the equations in **parameters.dat** and store the parameter's **name** and its corresponding **value**. The comments in the parameter file describe what each parameter setting means.

ParameterReader class functions:

phraseOneLine(..) sorts out the line: *x = 1 # comments*. It then passes the equation part, x = *1*, to the function **phraseEquationWithoutComments(..)**.

phraseEquationWithoutComments(..) separates the l.h.s. and r.h.s. of the equation: $x = 1$

setVal(..) amends x to the string vector ***names** and 1 to the double vector ***values**

getVal(..) takes in the parameter's **name** and returns its corresponding **value**.

2. Particlization surface

The **FO_data_reader** class object **freeze_out_data(..)** loads the data from the directory **input/surface.dat** to the **FO_surf** struct ***surf_ptr**, which stores the freezeout cells' information: spacetime coordinates, volume element, and hydrodynamic variables. For more information, see **readindata.h**

FO_data_reader class functions:

get_number_of_cells() gets the total number of freezeout cells in the data file

read_surf_switch(..) contains a variety of reader functions to read in different file formats: viscous hydrodynamics, anisotropic hydrodynamics and viscous hydroynamics with the thermal vorticity tensor. Each reader function describes what file format is needed. The **mode** parameter determines the type of file format to be read in.

The **dimension** parameter specifies whether the freezeout surface came from a 2+1d or 3+1d hydrodynamic simulation. The file format is the same for either case. For the case of 2+1d surfaces (**dimension = 2**), the η column should be zero in the data file, and we use the data file **tables/eta_trapezoid_table_xpt.dat** to extend the freezeout surface in η during the Cooper Frye integration.

read_resonances_list(..) see below...

3. Particle data

The function **read_resonances_list(..)** reads the PDG file **pdg.dat**, which contains information about all the relevant hadron species, and returns the number of hadrons (including anti-baryons) in the file. It also stores the PDG information (name, Monte Carlo ID, mass, etc) in the **particle_info** struct ***particle_data**. For more information, see **readindata.h**

4. δf coefficients

The **DeltafReader** class object **deltaf(..)** loads the coefficients appearing in the viscous correction δf to the **deltaf_coefficients** struct **df**. For more information, see **deltafReader.h**

The coefficients are tabulated in the directory **deltaf_coefficients**. The **df_mode** parameter determines what coefficients we read in and store in **df** using the class function **load_coefficients(..)**.

deltaf_coefficients/vh contains the 14-moment and Chapman Enskog δf coefficients of the hadron resonance gas for standard viscous hydrodynamics. The files are tabulated in (T, μ_B). For a given switching temperature and chemical potential, we bilinearly interpolate the data points. Currently, we assume $(T_{\rm sw}, \mu_{B,\rm sw})$ are fixed across the entire freezeout surface, although they can fluctuate numerically. We may relax this assumption in a future update.

deltaf_coefficients/vah so far contains PL-matching vahydro 14-moment coefficients (to be updated in the future). They are tabulated in (Λ, α_L). The freezeout cells have different values of (Λ, α_L), so we store these coefficients not in **df** but in ***surf_ptr**.

5. Tables

The necessary data tables such as the chosen particles in **chosen_particles.dat** and momentum arrays are set up before the main calculation.

6. Emission function

The **EmissionFunctionArray** class object **efa(..)** computes the particle spectra using the class function **calculate_spectra()**. It takes in the parameters, freezeout surface and particle information needed to compute the Cooper Frye formula for each hadron species in **chosen_particles.dat**

$$E\frac{dN_i}{d^3p} = \frac{g_i}{(2\pi\hbar)^3}\int_\Sigma d^3\sigma_\mu p^\mu (f_{{\rm eq},i} + \delta f_i) \tag{1}$$

More code details on the **EmissionFunctionArray** class can be found in **emissionfunction.h**

calculate_spectra()

First, we allocate arrays which hold the mass, quantum statistics sign, spin degeneracy, baryon number, and Monte Carlo ID of the chosen hadrons. The freezeout cells' variables are also loaded into individual allocated arrays. The δf coefficients, the type of which depends on **mode** and **df_mode**, are loaded into an array.

The **mode** and **df_mode** parameters let you calculate the spectra in different modes: continuous spectra or particle sampling, viscous hydro or anisotropic hydro δf, linearized δf or modified distribution $f_{\rm eq}^{\rm (mod)}$. These class functions will be described in separate pdfs.

After the calculation is finished, the results are written to file in the **results** directory. The arrays used in the calculation are deallocated.

Options

There are several options available to the user:

Setting **operation** = 1 computes the chosen particles' continuous spectra, which is stored in the array

***dN_pTdpTdphidy**. boldfwrite_dN_pTdpTdphidy_toFile() writes the continuous spectra to file.

Setting **operation** = 2 samples particles, which are stored in the **Sampled_particle**-type vector **particle_list**. The **Sampled_particle** class holds the Monte Carlo ID, mass, spacetime coordinates and four momentum of a sampled particle (recommend including all hadron resonances for any kind of afterburner calculation). **write_particle_list_toFile()**and **write_particle_list_OSC1997A()** write the sampled particles list to file in regular format and OSCAR format, respectively.

Setting **mode = 1** uses a distribution function $f = f_{\text{eq}} + \delta f$ computed from viscous hydrodynamics while **mode = 2** uses $f = f_a + \delta\tilde{f}$ from anisotropic hydrodynamics. **mode = 3** is idle at the moment...

Setting **mode = 5** allows you to compute the spin polarization vector S^μ for all particle species using the class function **calc_spin_polzn(..)**. For more details, see **emission_function_polzn_kernels.cpp**

One can include baryon chemical potential and various forms of the δf correction using the parameters **include_baryon**, **df_mode** and **include_x_deltaf**, respectively. The **df_mode** value needs to be consistent with the **mode** parameter (e.g. vhydro to vhydro)

Setting **do_resonance_decays = 1** allows you to include the effects of resonance decays on the continuous particle spectra (**operation = 1**) using the class function **do_resonance_decays(..)** **write_dN_pTdpTdphidy_with_resonance_decays_toFile()** writes the continuous spectra with resonance decays to a separate file