ASTROBEAR Diffusion Solvers

I. Solver Outline

Listed are the solvers we need in Astrobear for solving diffusion in multi-physics problems.

Implicit Solver for Heat Conduction

This solver uses hypre to solve for the energy diffusion with no field dependence.
This solver is suitable for small scale, large heat conductivity problems.

For the details of the hypre solver, see hypre documentation.

Explicit Solver for Field-Dependent Heat Conduction

This is solver solves explicitly the field-dependent heat conduction.
This solver is suitable for large scale, small heat conductivity problems.

Routine BDiffusion(Info) takes the info structure and udpate the domain according to equation (2). The code structure is as follows:

Mixed Implicit-Explicit Solver for the Radiation Transfer

This solver is derived from the implicit and explicit heat conduction solver. The difference is that the energy is coupled into the fluid equations in a more complicated manner. For details, see paper *Krumholz. M. R. et al, 2007, ApJ, 667, 626*

II. Explicit Solver for the anisotropic heat conduction with Flux Limiters

We use a central symmetric algorithm to evaluate the fluxes flowing into a certain cell. The fluxes applied to a certain cell may be constrained by numerical and physical conditions. Thus we introduce some limiters into our symmetric algorithm in order to fit these requirements. Once the fluxes are obtained, we use the divergence law to obtain the induced energy change at the cell center.

Consider the 2D problem. In this case, there are 4 corners for a certain cell. The symmetric diffusion algorithm requires us do a interpolation to get the corner values of the density, field strength, temperature and so on. After the corner values are obtained, we apply another interpolation in order to get the heat fluxes flowing in or out of a certain interface.

For a certain cell, the flux flowing into the cell can be written as:

$$\vec{q}=-\vec{b}\,n(\chi_C-\chi_R)(\vec{b}\cdot\nabla)T-n\chi_R\nabla T$$

where b is the unit vector along the field direction. χ are the temperature dependent thermal conductivities. The subscripts “C” and “R” denote parallel and perpendicular diffusion, respectively. The detailed expressions are:

$$\chi_C=\kappa_C T^{5/2}$$

$$\chi_R=\kappa_R\frac{n}{B^2 T^{1/2}}$$

here the parallel conductivity is much larger than the perpendicular conductivity (larger by a factor of 10^9). These expressions are from the Orlando simulation. In a different scenario (i.e. when considering some dense and high pressure plasma), the expressions may be different. Currently only two cases are implemented:

(1) $\chi_C=\kappa_C T^{5/2}$ $\quad \chi_R=\kappa_R\frac{n}{B^2 T^{1/2}}$

(2) $\chi_C=\kappa_C T^{5/2}$ $\quad \chi_R=0$

On the x direction, we should have:

$$q_x=-b_x n(\chi_C-\chi_R)(b_x\partial_x T+b_y\partial_y T)-n\chi_R\partial_x T$$

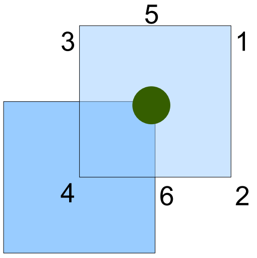
$$q_y=-b_y n(\chi_C-\chi_R)(b_x\partial_x T+b_y\partial_y T)-n\chi_R\partial_y T$$

Notice that there is a dT/dy dependence for qx, and a dT/dx dependence for qy. These are the cross terms that come along as a result of anisotropy. Let us look at qx for example, first, look at only the dT/dx terms. We have:

$$q_{xx}=-[b_x^2 n(\chi_C-\chi_R)+n\chi_R]\partial_x T$$

This term can be evaluated at the corners in our symmetric scheme. Take the first corner for instance, for the quantities that has a body centered value, we have:

$$n(\chi_C-\chi_R)=\underset{i=1}{\overset{4}{HM}}(n_i(\chi_C-\chi_R)_i)$$



$$n(\chi_R)=\underset{i=1}{\overset{4}{HM}}((n\chi_R)_i)$$

in the configuration on the right. Symbol “HM” means taking the harmonic mean.

For the quantities that has a face centered value, we have:

$$b_x=(b_{x5}+b_{x6})/2$$

The only term left is the dT/dx term. For this term, we introduce the limiter proposed by Sharma&Hammet:

$$\partial_x T=S[(\partial_x T)_{5,}(\partial_x T)_6]$$

where

S[a, b] = (a+b)/2 if min(k*a, a/k) < (a+b)/2 < max(k*a, a/k)
= min(k*a, a/k) if (a+b)/2 < min(k*a, a/k)
= max(k*a, a/k) if (a+b)/2 > min(k*a, a/k)

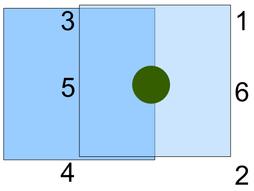
where 0 < k < 1.

By the same token, we can get:

$$q_{yy}=-[b_y^2 n(\chi_C-\chi_R)+n\chi_R]\partial_y T$$

So for each corner, we need to compute 4 mean values and 1 slope limited value.

With these terms evaluated, we can find out the diagonal terms, we can get the diagonal contributions for to the flux into a certain interface by doing arithmetic averaging over the corners. So far we have obtained qxx at the left interface. We still need the cross term contribution to the flux. Using the same example as above, now consider qxy. We no longer find the corner values first, but rather compute the interface contribution directly by applying an appropriate limiter. We evaluate:



$$\partial_y T=MC\{MC[(\partial_y T)_{1,}(\partial_y T)_2],MC[(\partial_y T)_{3,}(\partial_y T)_4]\}$$

where MC denotes the Monotonized Central limiter. We can also interpolate the interface values:

$$b_y=(b_{y1}+b_{y2}+b_{y3}+b_{y4})/4$$

$$n(\chi_C-\chi_R)=\underset{i=5}{\overset{6}{HM}}(n_i(\chi_C-\chi_R)_i)$$

Putting the above values into the following expression:

$$q_{xy}=-b_x b_y n(\chi_C-\chi_R)\partial_y T$$

we then obtain qxy.

Then finally, the left interface qx is written as:

$$q_x=q_{xx}+q_{xy}$$

The next step is to apply the physics limiter, which is proposed by Cowie & McKee in their 1977 paper. Cowie & McKee 1977. The basic idea is that the classic conductivity is correct only when the electron mean free path is much smaller than the temperature scale height. When the opposite is true, the flux should really be:

$$q_{sat}=3/2\,n_e k\,T_e v$$

where v is the characteristic velocity which may comparable to the electron thermal velocity. We therefore write down:

$$q_{sat}=\phi\,n_e k\,T_e\sqrt{kT/m}$$

where Φ is a positive factor depending on the actual physics condition. Then apparently from

$$n\,k\,T=p=\rho\,c_s^2$$

and

$$\sqrt{kT/m}=c_s$$

we can write:

$$q_{sat}=\phi\,\rho\,c_s^3$$

Some detailed calculation shows that we can write:

$$q_{sat}=5\,\phi\,\rho\,c_s^3$$

with Φ taking a positive value according to the physics situation. For fully ionized cosmic gas, an estimation is that $0.24 < \Phi < 0.35$.

In the code, we take:

$$q_{sat}=sgn(q_x)5\,\phi\,\rho\,c_s^3$$

and then perform the limiting using the harmonic mean:

$$Q_x = HM\left(q_x, q_{sat}\right)$$

This is the final result for the flux into the left interface.

The other three interface fluxes can be computed by the same procedure. Notice that the Cowie&McKee saturation flux should really act on a total flux but not its components. Because of our numerical scheme, it is meaningless to obtain a total flux at a certain cell and saturate it. This difference may (or may not) result in a discrepancy between our simulation and some strongly saturated analytical model.

III. Code Structure

At first, the code determines a preferred time step according to the stability requirement. Then it enters into sub-cycles and updates each cell by solving the fluxes around it. Currently only transparent boundary condition is implemented.

IV. Code Tests

There are many interesting problems worth looking at. The first test will be the MTI test. See the paper by Parrish&Stone.