A New Propagation Code For Cosmic Ray Nucleons

Ingo Büsching,[1] Andreas Kopp,[2,1] Martin Pohl[1] and Reinhard Schlickeiser[1]
(1) Institut für Weltraum- und Astrophysik, Ruhr-Universität Bochum, Bochum, Germany
(2) MPI für Aeronomie, Katlenburg-Lindau, Germany

Abstract

We present a newly developed numerical code solving the time-dependent Cosmic Ray (CR) propagation equation for three spatial dimensions, momentum p being a free parameter. Assuming the distribution of interstellar gas in the Galactic disc to be independent of the Galactocentric radius, r, and of the azimuth, φ, a series ansatz is used to transform the three-dimensional spatial problem into a system of one-dimensional equations for the coefficients of this expansion. The resulting equations for the coefficients can be solved very efficiently, using a modified leapfrog/DuFort-Frankel scheme, on normal PC-style hardware. Since the resolution in r and φ merely depends on the number of expansion coefficients, the spatial resolution in these directions is only limited by the available CPU-time. The grid z-direction may be customized to the problem in question.

1. Introduction

Supernovae are regarded as the most probable sources of Galactic CRs. Viewing the whole Galaxy, these sources may be considered as point like. This means, in order to investigate the influence of the spatial distribution of these kind of sources on CR spectra, one needs to solve the cosmic ray propagation equation (1) with high resolution in all three spatial dimensions and time.

We therefore seek for a way for a distributed computation of the solution of the PDE for the cosmic ray density N

$$\frac{\partial N}{\partial t} - S = k\left(\frac{\partial^2 N}{\partial r^2} + \frac{1}{r}\frac{\partial N}{\partial r} + \frac{1}{r^2}\frac{\partial^2 N}{\partial \varphi^2} + \frac{\partial^2 N}{\partial z^2}\right) - \Omega N \qquad (1)$$

in a cylindrical geometry with radius R and height $2H$. Here, $k = k(p)$ is the spatial diffusion coefficient and $S = S(r,\varphi,z,t)$ represents the sources. N has to satisfy homogeneous boundary conditions in r and z. Under the condition, that the loss term Ω is independent of r and φ (i.e. $\Omega = \Omega(z)$), we use the ansatz

$$N = \frac{1}{\pi}\sum_n \sum_\alpha \left(A_{n,\alpha}\cdot\cos(n\varphi) + B_{n,\alpha}\cdot\sin(n\varphi)\right)\frac{\jmath_n(\alpha r)}{\left(\jmath_n'(\alpha R)\right)^2} \qquad (2)$$

pp. 1985–1988

α are the solutions of $J_n(\alpha R) = 0$, and J_n the Bessel function of order n. Inserting Eq. (2) into Eq. (1) one gets equations for the expansion coefficients $A_{n,\alpha}$

$$\frac{\partial A_{n,\alpha}}{\partial t} - S_{n,\alpha}^{(A)} = k(p)\left\{-\alpha^2 A_{n,\alpha} + \frac{\partial^2 A_{n,\alpha}}{\partial z^2}\right\} - \Omega(z)\frac{A_{n,\alpha}}{T(p)} \tag{3}$$

and similar equations for $B_{n,\alpha}$ with

$$S_{n,\alpha}^{(A)} = \int_{\varphi=0}^{2\pi}\int_{r=0}^{R} S(r,\varphi,z,t)\cos(n\varphi)\, J_n(\alpha r)\, r dr d\varphi \tag{4}$$

$$S_{n,\alpha}^{(B)} = \int_{\varphi=0}^{2\pi}\int_{r=0}^{R} S(r,\varphi,z,t)\sin(n\varphi)\, J_n(\alpha r)\, r dr d\varphi \tag{5}$$

2. The code

As the ansatz (2) requires the solution of (3) for a large number of coefficients α, we were looking for a method that is able to very efficiently compute these solutions. We choose a combined DuFort-Frankel/leapfrog scheme, as this scheme is explicit and of second order accuracy in space and time. Thus, the values of the next time step may be calculated in a straightforward manner (i.e without time consuming matrix inversions). Below this scheme and the original DuFort-Frankel scheme are described. The equations (3) are of the type

$$\frac{\partial N}{\partial t} = a\frac{\partial^2 N}{\partial z^2} + b\frac{\partial N}{\partial z} + cN + d \tag{6}$$

The second derivative is of particular interest, and an appropriate method to solve this equation is the DuFort-Frankel scheme. It was originally developed for the case that only $a \neq 0$ [4,5]. The scheme uses a time average on the r.h.s., because otherwise the scheme would be unconditionally unstable. The discretization reads:

$$\frac{N_i^{t+1} - N_i^{t-1}}{2\Delta t} = a\frac{N_{i+1}^t - \left(N_i^{t+1} + N_i^{t-1}\right) + N_{i-1}^t}{(\Delta z)^2} \tag{7}$$

where N_i^t is the density at grid position i at time step t. This semi-implicit scheme can be easily solved for N_i^{t+1}, giving with $\eta = a\frac{2\Delta t}{(\Delta x)^2}$

$$N_i^{t+1} = \frac{1}{1+\eta}\left(N_i^{t-1} + \eta\left(N_{i-1}^t + N_{i+1}^t\right)\right) \tag{8}$$

This scheme is always stable [5], but the time step has nevertheless to be chosen very carefully, as the scheme may converge to the wrong solution for larger time steps. Regrouping the scheme (8) reveals that the scheme actually solves the equation [4]:

$$\frac{\partial N}{\partial t} = a\frac{\partial^2 N}{\partial z^2} + a\frac{(\Delta t)^2}{(\Delta z)^2}\frac{\partial^2 N}{\partial t^2} \tag{9}$$

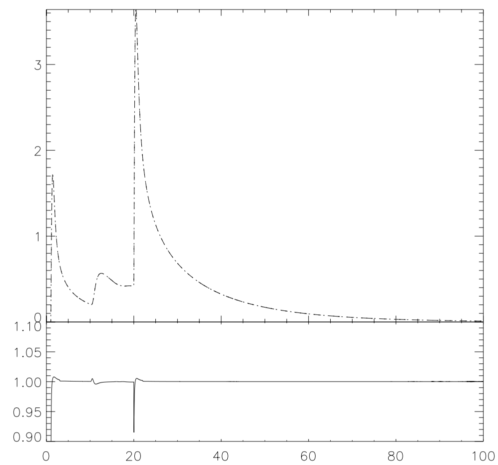

Fig. 1. Comparison of analytical (dotted) and numerical (dashed) solution of Eq. (3) (upper panel) and the quotient of numerical and analytical solution (lower panel).

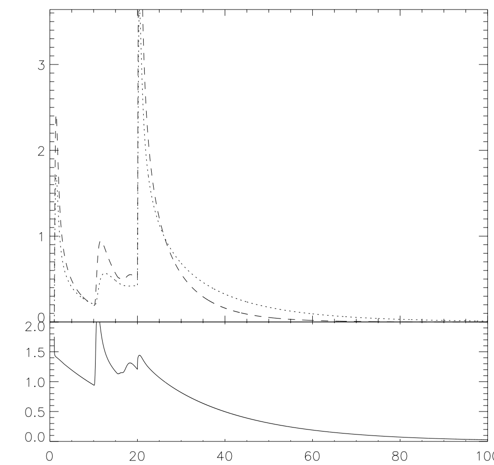

Fig. 2. Same as Fig. 1 but for a time step enhanced by a factor of five. The code is numerically stable, but produces a solution which does not solve Eq. (3).

so, in order to obtain correct results, one has to ensure that $\frac{(\Delta t)^2}{(\Delta z)^2} \ll 1$. Figs. 1 and 2 show a comparison between the analytical solution (which is available for $\Omega = \text{const}$ [3]) for only a few sources with two numerical solutions for different time steps. While a small time step (Fig. 1) shows a excellent agreement between the analytical solution and the numerical result, a larger time step (5 times higher, Fig. 2) produces a stable but unphysical result. The 10% deviation at the maximum of the first case appears due to the truncation of a series-expansion in the analytical solution.

To deal with the additional terms in (6) the DuFort-Frankel scheme is extended quite similar to a leapfrog scheme (e.g. [4], [5])

$$\begin{aligned} \frac{N_i^{t+1} - N_i^{t-1}}{2\Delta t} \quad &= \quad b\frac{N_{i+1}^t - \left(N_i^{t+1} + N_i^{t-1}\right) + N_{i-1}^t}{(\Delta z)^2} + \frac{N_i^{t+1} - N_i^{t-1}}{2\Delta z} \\ &\quad + cN_i^t + d \end{aligned} \tag{10}$$

This formula can again be solved for N_i^{t+1}. The term proportional to c needs particular considerations; taking into account that the scheme (10) deals with two staggered grids, one sees that this term belongs to the "wrong" grid (cf. [6] for details). This term contains the parameter α (cf. Eqs. (3)) which becomes rather dominating and could thus destabilize the code. This term may be replaced by a simple spatial average. This allows fast computations and works well for smaller values of α (in fact it worked well for our computations done so far [2]), but brings in some artificial diffusion. Thus, we also implemented the more elaborated method consisting of three steps by [1] and [6], which is slower, but more accurate and suitable also for higher values of α and smaller diffusion coefficients.

3. Conclusion

We introduced a newly developed code, designed for massive distributed computation of diffusion-loss type equations in cylindrical geometry on normal PC-style hardware. With this code, it is possible to obtain results of the CR propagation equation for the whole Galaxy with high resolution in all three spatial directions and time.

Partial support by the Bundesministerium für Bildung und Forschung through the DLR, grant *50 OR 0006*, is gratefully acknowledged.

4. References

1. Birk G.T, Otto A., 1996, JCP 125, 513
2. Büsching, et al. 2003, First Results Of A New Cosmic Ray Propagation Code, this proceedings
3. Butkovskiy A.G., 1982, Green‘s Functions and Transfer Functions Handbook Ellis Horwood Ltd., Chichester
4. Fletcher C.A.J., 1991, Computational Techniques for Fluid Dynamics Springer-Verlag, Berlin
5. Potter D., 1973, Computational Physics John Wiley & Sons, Lodon
6. Schröer A., Birk G.T, Kopp A. 1998, Comp. Phys. Commun 112, 7