Geoscientific Model Development Discussions

Verification of SpacePy's radial diffusion radiation belt model

D. T. Welling, J. Koller, and E. Camporeale

Los Alamos National Laboratory, P.O. Box 1663, Los Alamos, NM 87545, USA

Correspondence to: D. T. Welling (dwelling@lanl.gov)

GMDD

4, 2165–2197, 2011

Radbelt verification

D. T. Welling et al.

Title Page

Abstract	Introduction
Conclusions	References
Tables	Figures

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Abstract

Model verification, or the process of ensuring that the prescribed equations are properly solved, is a necessary step in code development. Careful, quantitative verification guides users when selecting grid resolution and time step and gives confidence to code
5 developers that existing code is properly instituted. This work introduces the RadBelt radiation belt model, a new, open-source version of the Dynamic Radiation Environment Assimilation Model (DREAM) and uses the Method of Manufactured Solutions (MMS) to quantitatively verify it. Order of convergence is investigated for a plethora of code configurations and source terms. The ability to apply many different diffusion
10 coefficients, including time constant and time varying, is thoroughly investigated. The model passes all of the tests, demonstrating correct implementation of the numerical solver. The importance of D_{LL} and source term dynamics on the selection of time step and grid size is also explored. Finally, an alternative method to apply the source term is examined to illustrate additional considerations required when non-linear source terms
15 are used.

1 Introduction

The terrestrial radiation belts are regions of near-Earth outer space where relativistic electrons and ions are electromagnetically orbiting the planet. The belts naturally organize into two tori: a stable inner belt lying within $\approx 2\,R_E$ and a far more dynamic outer
20 belt located outside of $\approx 3\,R_E$ (van Allen and Frank, 1959). Since their discovery, the belts have been the focus of intense research due to the innumerable unknowns concerning their behavior and their damaging effects on spacecraft, both transient (Baker, 2000; Feynman and Gabriel, 2000; Koons et al., 1999; Pirjola et al., 2005, etc.) and accumulated over a satellite's lifetime (Gubby and Evans, 2002; Welling, 2010).
25 During slowly changing conditions, radiation belt particles undergo three types of periodic motion, each with its own corresponding adiabatic invariant: gyration about

field lines, bounce along field lines, and drift about the Earth. During active times, when conditions change on time scales shorter than the periods of motion, adiabaticity can be broken and particle motion can no longer be described by a simple sum of these three. Casting the particle evolution in terms of the three invariants allows non-adiabatic motion to be modeled diffusively via the Fokker-Planck equation, yielding a powerful description of belt dynamics (Schulz and Lanzerotti, 1974).

Of the three invariants, the third, which represents the total magnetic flux enclosed within a full particle orbit, is especially important. It is common to use a normalized form of this, L^* (referred to as simply L herein), described by Roederer (1970). It is analogous to radial distance from the center of the Earth (in Earth Radii) to the equatorial crossing point of the bouncing particle, exactly so if the terrestrial magnetic field is adiabatically relaxed to a simple dipole geometry. Diffusion in L alone (the other invariants shall be considered conserved) accounts for the capture and inward radial transport of radiation belt particles (Roederer, 1970; Fälthammar, 1965).

The evolution of the phase space density, f, of a radiation belt population of constant μ and K (values corresponding to the first and second adiabatic invariants, respectively) for a given diffusion rate, D_{LL}, is expressed as (Lyons and Schulz, 1989),

$$\frac{\partial f}{\partial t} = L^2 \frac{\partial}{\partial L}\left(\frac{D_{LL}}{L^2}\frac{\partial f}{\partial L}\right) + Q \tag{1}$$

where Q represents combined sources and losses. Solving this equation for f has been the focal point of many important radiation belt studies. Brautigam and Albert (2000) and Miyoshi et al. (2003), using different boundary conditions at high L and different loss processes, demonstrated the importance of radial diffusion in radiation belt dynamics. Shprits et al. (2006) showed that radial diffusion and magnetopause shadowing (particle loss as the magnetopause intersects drift paths) can account for observations of rapid flux dropouts at the onset of particular storms. Lam et al. (2007) used a radial diffusion model to explore the importance of plasmaspheric hiss driven losses. The results compared favorably to measurements of the radiation belts around $L \approx 4$. These are but a few examples; a more complete review can be found in Shprits

et al. (2008). Clearly, radial diffusion representation of the belts is a powerful tool for investigation of their dynamics.

Many models and studies have moved beyond simple one dimensional diffusion. The Salammbo model (Beutier and Boscher, 1995), for example, is a three-dimensional diffusion model, which allows not only radial but also pitch angle and energy space diffusion. The Dynamic Radiation Environment Assimilation Model (DREAM, Reeves et al., 2005) incorporates data assimilation to drive radial diffusion results towards more realistic values (Koller et al., 2007). Other models, such as the Radiation Belt Environment (RBE) model (Fok et al., 2008), use bounce-averaged kinetic representations of the belts along with time-accurate magnetic and electric field specifications instead of simpler diffusion equations. Despite these important modeling advances, radial diffusion still remains a core part of radiation belt investigations.

This work introduces a new, open source version of the DREAM radial diffusion radiation belt code and, as part of the code's development, verifies the model. Code verification is the process of testing for proper implementation of a code's numerics and other processes. *Verification* asks, "Is the model solving the prescribed equations correctly?" and is separate from model *validation*, which asks, "Are the equations solved representative of the real world?"; examples of quantitative model validation can be found in Wang et al. (2008) and Welling and Ridley (2010). Verification studies are key steps in code development and guide user decisions in terms of code configuration. Proper verification also expedites future development; as additional features and expanded physics are added to a model, it is critical to ensure that the existing features are instituted properly. Because RadBelt is currently being expanded to include diffusion in pitch angle and energy space, verification at this early stage is crucial.

This study leverages the Method of Manufactured Solutions, described below, to quantitatively assess the code's performance and ensure that the results are accurate to typical machine limitations. Code exercises are repeated for a plethora of configurations to demonstrate proper implementation and convergence. Alternative source functions and numerical source applications are applied to test all facets of the model.

2 Methodology

2.1 Model description

The code being investigated here is the RadBelt module of the SpacePy software library (Morley et al., 2011). This model solves Eq. (1) for a single μ and K. The bulk of the code is written in Python, making the model unique among others in terms of flexibility and capability- the model is initialized, configured, executed, and visualized all through a Python interface. Code domain (typically $L = [1 : 10]$), sources and losses, D_{LL}, time step and grid size are all specified in an object-oriented manner, e.g., setting object attributes. Required input data, such as K_p indices, are obtained automatically by SpacePy from the Qin et al. (2007) input set housed at the Virtual Radiation Belt Observatory (ViRBO). Results can be quickly visualized through built-in object methods. The RadBelt module is currently being expanded to include full three dimensional diffusion and data assimilation capabilities.

Several built-in empirical relationships act as default sources and losses in RadBelt. The default electron source is an empirical acceleration term given by,

$$S = \gamma (K_P)^2 e^{\frac{-(L - L_{center})^2}{2L^2_{width}}} \tag{2}$$

where γ is the magnitude of the source (typically $0.1\,\text{Days}^{-1}$), L_{center} is the center of the source curve (typically $L = 5.6$), and L_{width} is the width of the source curve (typically 0.3). This function represents electrons being accelerated into the current μ-K slice as a function of magnetospheric activity. Though simple, it gives users a quick, built-in source of phase space density. An important loss mechanism is particles diffusing into open drift shells and escaping from the magnetosphere, known as magnetopause shadowing. The location of the last closed drift shell, L_{max}, a value strongly controlled by the location of the magnetopause, is therefore vital for accurate radiation belt modeling. In RadBelt, the following empirical relationship yields the default L_{max} values:

$$L_{max} = 6.07 \times 10^{-5}\,\text{Dst}^2 + 0.0436\,\text{Dst} + 9.37 \tag{3}$$

This relationship, introduced by Koller and Morley (2010), is the result of fitting a second order polynomial to L_{max} versus Dst values from July 2002 to December 2002 at a five minute time resolution. L_{max} was calculated using the Tsyganenko 2001 storm-time empirical magnetic field model (Tsyganenko, 2002; Tsyganenko et al., 2003). The

5 root-mean-squared error of this relationship is 0.64 R_E. Loss due to plasmaspheric hiss (Millan and Thorne, 2007), important for developing the slot region, is included by using a loss lifetime of 10 days. The radial extent of the plasmapause is obtained via the K_p-dependent relationship defined in Carpenter and Anderson (1992). Though these relationships are the defaults, RadBelt is flexible and any arbitrary functions can

10 be used.

An example of RadBelt output is shown in Fig. 1, which shows the results from a simulation of the well known "Halloween Storms" occurring from 20 October 2003 to 5 December 2003. The simulation was run using the Brautigam and Albert (2000) specification of D_{LL} and the default source and losses. The top frame shows electron

15 phase space density for $\mu = 2083\,\mathrm{MeV\,G^{-1}}$ and $K = 0.03\sqrt{G}R_E$ over the L^* domain. These values will be used throughout this study. The over-plotted white line shows the location of the last closed drift shell for this particular μ and K combination. The bottom frame shows the Dst and K_p indices during the simulated period. These values drive both the D_{LL} and the empirical functions described above.

20 The driving physics of any radial diffusion model is contained in the diffusion coefficient, D_{LL}. Recognizing this, RadBelt presently includes several different D_{LL} formulations. Adding a new formulation is as simple as defining a new function and assigning it as a RadBelt object attribute. The currently included D_{LL} models are taken from Selesnick et al. (1997), Brautigam and Albert (2000), Fei et al. (2006), and Ukhorskiy

25 and Sitnov (2008), denoted as S1997, BA2000, FC2006, and U2008 herein for convenience. Each of these has the generic form,

$$D_{LL} = \alpha L^{\beta} \tag{4}$$

The values of α and β are summarized in Table 1; each is compared in Fig. 2 over the typical spatial domain of RadBelt.

RadBelt uses a modified Crank-Nicolson implicit finite difference solver which is second-order accurate in space and time (Crank and Nicolson, 1947). Equation (1) has several characteristics that make the standard Crank-Nicolson approach inappropriate, namely a space (and potentially time) dependent diffusion coefficient as well as a right hand side that contains factors of L^2 in both the numerator and denominator. The traditional solver is modified by combining the factor of L^{-2} with D_{LL} and discretizing the right hand side via a second order difference scheme that takes into account the spatial dependence of D_{LL}, outlined in Press et al. (2007). The possibility of a time-dependent D_{LL} is accounted for by computing it at the midpoint between t and $t + \Delta t$. This approach maintains unconditional stability of the scheme as demonstrated by Tadjeran (2007). The resulting discretized form of Eq. (1) is,

$$\frac{f_j^{n+1} - f_j^n}{\Delta t} = Q_j^{n+\frac{1}{2}} \tag{5}$$

$$+ \frac{L_j^2}{2} \left[\frac{D_{j+\frac{1}{2}}^{n+\frac{1}{2}} \left(f_{j+1}^n - f_j^n \right) - D_{j-\frac{1}{2}}^{n+\frac{1}{2}} \left(f_j^n - f_{j-1}^n \right)}{\Delta L^2} \right.$$

$$\left. + \frac{D_{j+\frac{1}{2}}^{n+\frac{1}{2}} \left(f_{j+1}^{n+1} - f_j^{n+1} \right) - D_{j-\frac{1}{2}}^{n+\frac{1}{2}} \left(f_j^{n+1} - f_{j-1}^{n+1} \right)}{\Delta L^2} \right]$$

where indexes n and j represent discretization in time and space, respectively. The derivation then follows the usual path of separating f^{n+1} terms from f^n terms, resulting in the familiar arrangement,

$$\mathbf{A} f^{n+1} = \mathbf{B} f^n + \Delta t Q^{n+\frac{1}{2}} \tag{6}$$

where $\mathbf{A}$ and $\mathbf{B}$ are tridiagonal matrices of coefficients that include factors of L, Δt, ΔL, and D_{LL}. In RadBelt, routines to invert $\mathbf{A}$ via standard LU tridiagonal matrix

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

decomposition (Press et al., 2007) to advance the solution forward in time are written in the C programming language to obtain the fastest possible execution speed.

2.2 Verification technique

The prototypical code verification approach begins by obtaining an analytical solution to the equation or system of equations. The known solution is compared to the numeric solution in a systematic way covering a range of grid spacings and time steps, demonstrating the code's ability to properly converge to the correct solution as expected given the implemented solver. For complicated systems, however, obtaining an analytical solution becomes either prohibitively difficult or outright impossible.

One way to overcome this difficulty is to employ the Method of Manufactured Solutions (MMS) (Roache, 2002). This method begins by selecting a solution independent of the governing equations. The manufactured solution is substituted into the governing equation to produce a source term that, when added to the original governing equation, yields a new equation for which the manufactured solution is an exact solution. The manufactured solution can now be applied to convergence studies of the code, which, if passes, can be considered verified – it correctly solves the governing equations and properly applies the given numerical scheme (Roache, 1998)

This process is illustrated clearly for the situation at hand. The arbitrary solution is selected with no regard for Eq. (1) outside of the variables we wish to exercise (t and L):

$$f = \sin(aL - a)\sin(bt) \tag{7}$$

where a and b will be set to satisfy prescribed boundary conditions. This "solution" is > 2 times differentiable in both t and L, which is required once substituted into Eq. (1). Substituting Eqs. (7) and (4) into Eq. (1) yields the source term,

$$Q_{MMS} = b\cos(bt)\sin(aL - a) \tag{8}$$
$$-\alpha a(\beta - 2)L^{\beta-1}\cos(aL - a)\sin(bt)$$
$$+\alpha a^2 L^{\beta}\sin(aL - a)\sin(bt)$$

For this exercise, Dirichlet boundary conditions will be used in L (set to zero for convenience using a typical domain of $L = [1 : 10]$) while the initial conditions will be no phase space density. This restricts the choice of a to $\frac{2\pi}{9}$. The constant b is then chosen to ensure a source that has a short period so that the verification simulations can be performed quickly. Given a simulation run to $t = 600s$, b is set to $\frac{2\pi}{300}$ such that two full periods are completed in one simulation.

The behavior of both the manufactured solution and the associated source term for the chosen values of a and b are shown in Fig. 3. As desired, the solution shows plenty of variation in just a short time period (Fig. 3, top frame). For a moderate D_{LL} (S1997, whose behavior is shown as a black dashed-dotted line in Fig. 2), the first term of Eq. (8) dominates and the artificial source is similar to the manufactured solution, but phase-shifted and scaled (Fig. 3, center frame). However, for a strong D_{LL}, this pattern changes drastically (bottom frame). This occurs for the BA2000 D_{LL} when K_p is high. As α increases by several orders of magnitude, the L-dependent second and third terms of Q_{MMS} (Eq. 8) become dominant. This is to overcome the strong storm time diffusion of electrons at high L and maintain the shape of the manufactured solution.

With this manufactured solution and source term, the RadBelt model was run thousands of times with different combinations of Δt and ΔL. Each run covered ten minutes of simulation time, or two full periods of the manufactured solution, over the typical radial domain of $L = [1 : 10]$. This was repeated for each D_{LL} model currently included in the code.

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

3 Results

3.1 Time constant D_{LL}

Figure 4 illustrates the model converging towards the analytical solution as Δt is refined for a constant $\Delta L = 0.1$ with S1997 D_{LL}. The left column displays the solution for five

5 different values of Δt, decreasing from top to bottom, while the right column shows the corresponding error (defined as the magnitude of the difference between the analytical solution and the numerical solution) throughout the domain. The maximum error for each is listed at the top of each error plot. A quick comparison between the time steps used and the corresponding maximum error suggests that the code is converging at the

10 expected order (second) given the selection of solver: when the time step is decreased by a factor of x, the error decreases by a factor of x^2.

Figure 5 illustrates this clearly and quantitatively. Time steps and associated error values are plotted on a log-log scale for 100 separate simulations (top frame). These results are naturally sorted into three distinct regimes. On the far right is the regime

15 where the time step is too large to properly capture the dynamics. This is illustrated in the top row of Fig. 4, where $\Delta t = 300\,\text{s}$ misses most of the dynamics of the analytical solution and the error is greater than the amplitude of the forcing source term. On the far left is the regime where the error is dominated by floating point error. Though the time step is refined throughout this regime, the maximum error is not reduced as it is

20 either of machine origin or requires further refinement in ΔL. This is observed in the bottom two rows in Fig. 4 where the error reduction is less than expected given the second order solver. The center region of Fig. 5, delimited by the vertical dashed red lines, is known as the asymptotic regime. Here, the time step and grid size are both appropriate for the conditions being simulated and the error drops off as expected given

25 the selected numerical scheme. The slope of this region is nearly equal to the order of the solver, indicating that the code is correctly solving the system as intended. The lower frame of Fig. 5 shows that the same is true when the grid resolution is refined.

Figure 6 takes these results one step further. The number of simulations is now increased to 8000, and the error for each Δt-ΔL combination is sorted into a two dimensional color map. The white contour lines show order-of-magnitude boundaries clearly. Figure 6 not only confirms the results from above, but also reveals more about the code's behavior for a particular D_{LL} and Q_{MMS}. For $\Delta t > 1$ s, refining the grid has almost no effect on the overall error. In this region, the contours are parallel to the ΔL axis. This is because for a moderate value of D_{LL} and a quickly varying source function, it is much more important to have a properly refined time grid as opposed to a fine L grid. Only when the periodic source term is properly resolved does reducing the grid size improve results. The curve that passes through the intersection of the vertical and horizontal contours (not shown) traces the most efficient combinations of Δt and ΔL. This information is important to code users who want to maximize code performance in terms of controlling error and maintaining fast run speeds; understanding the balance between the rate of variation in time and space is key.

This process was repeated for all other D_{LL} models. For BA2000, the simulations were performed for $K_p = [0:9]$ for a total of fourteen separate verification sets. The results are summarized in Fig. 7. Returning to Fig. 2, it would be expected that the error maps for FC2006, U2008, and BA2000 for low K_p should appear similar to the error map for S1997 (Fig. 6) as the D_{LL} models are similar in shape and have weak to moderate values. As seen in the top four frames of Fig. 7, this is indeed the case. As K_p increases, however, BA2000 D_{LL} becomes far stronger than any other values investigated to this point. This is evident in the bottom two frames of Fig. 7, as the region where ΔL refinement is most important (contour lines are dominantly parallel to the Δt axis) grows as K_p increases from 6 (lower left frame) to 9 (lower right frame). Again, it is important for code users to understand the balance between time dynamics and diffusive L dynamics when selecting time step and grid resolution.

3.2 Time dependent D_{LL}

Real world applications of RadBelt will involve a time-varying K_p. When using the BA2000 model, which is K_p dependent, D_{LL} will vary throughout the simulation time domain. Verification of RadBelt's ability to properly capture this new complexity must
5 be performed.

The new test case is summarized in the top frame of Fig. 8. K_p is allowed to change from 9 to 1 in a smooth, linear fashion over the simulated temporal domain (shown in green with the scale on the left). This results in an exponentially decreasing D_{LL} (shown in blue with the scale to the right). Real K_p values vary as a step function
10 with a three-hour window; it is often linearly interpolated to create a smoothly varying function when used as a model input. As such, the situation selected in the top frame of Fig. 8 represents conditions found in science applications.

The bottom frame of Fig. 8 illustrates that the code indeed converges properly as Δt decreases (constant $\Delta L = 0.01$) when a time-varying D_{LL} is used. Because the
15 maximum error is bounded by what can accumulate when D_{LL} is at its maximum, the curve closely matches what is observed for a constant BA2000 $D_{LL}(K_P = 9)$ (Fig. 7, lower right frame, values along $\Delta L = 0.01$). Similar tests for an impulsive K_p that jumps from 1 to 9 at $t = 300$ s were performed with similar, positive results (not shown). These results demonstrate that RadBelt properly handles a time-varying D_{LL} model.

3.3 Alternative source function

Though unlikely (Roache, 1998), false positives may be possible when using the MMS method. To test the veracity of the results to this point, portions of the above exercises are repeated using a new manufactured solution. This is the last step in ensuring proper verification of the RadBelt model.
25 The second solution selected is,

$$f = \sin(aL - a)(e^{bt} - 1) \qquad (9)$$

where a and b are again constants set to agree with boundary conditions. Unlike the original solution (Eq. 7), there is no longer periodicity in time. The constant a is again chosen to accommodate Dirichlet boundary conditions while b is set to bind the maximum value of the solution to a reasonable value within the time domain of the simulations. These requirements yield,

$$a = \frac{2\pi}{9} \qquad (10)$$

$$b = \frac{\ln(2)}{t_{\text{final}}} \qquad (11)$$

which binds the absolute maxima of the solution to ±1 when t_{final} is the total amount of time simulated. When substituted into Eq. (1), the new source term is,

$$Q_{\text{MMS}}^{\text{2nd}} = b\,\sin(aL - a)e^{bt} \qquad (12)$$
$$-a\alpha(\beta - 2)L^{\beta-1}(e^{bt} - 1)\,\cos(aL - a)$$
$$+a^2\alpha L^{\beta}(e^{bt} - 1)\,\sin(aL - a)$$

Figures 9 and 10 summarize the results when using the second manufactured solution and S1997 D_{LL}. Figure 9, similar to Fig. 4, demonstrates that the code does indeed converge on the analytic solution as Δt is progressively refined. The error values are much lower than before, likely a function of $Q_{\text{MMS}}^{\text{2nd}}$ (Eq. 12) varying far more slowly in time than the initial Q_{MMS} (Eq. 8). With the overall error drastically reduced, residual error at high L (lower right frame) becomes evident. This occurs in the region where D_{LL} is strongest. Because the new source term varies more slowly in time compared to the old, these L-dependent features can be seen at a higher Δt than before. In other words, the results for this new source term are more sensitive to ΔL refinement than Δt. This feature is illustrated clearly in Fig. 10, where the region of contours that follow constant ΔT values is reduced compared to what was seen in Fig. 6. The results

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

demonstrate proper convergence, but emphasize that the source term must be taken in to consideration when selecting time and space discretizations.

3.4 Code performance

Figure 11 benchmarks the code performance using a single core on an Intel Xeon 2.6 GHz CPU. The color map shows the time to complete each of the 8000 simulations for the BA2000 $D_{LL}(K_P = 9)$ set. The black contour lines show the same results in terms of simulation time (always ten minutes) divided by total CPU time. RadBelt runs very quickly on a modern machine, with most simulations finishing at >100-times real time. These results demonstrate that it is possible to combine the benefits of a modern interpreted programming language with the light weight, fast number-crunching abilities of a compiled language.

3.5 Source term application considerations

The source term used thus far has been simple enough to include in the derivation of the Crank-Nicolson solver. This leaves nothing to the imagination when the source term is applied. In real-world, non-verification situations, more complex source terms may arise that require a more sophisticated application, for example, a non-linear source term that depends on the phase space density. A popular approach is to split out the source term as one would when applying Strang splitting (Strang, 1968) to a multi-dimensional problem. This approach splits the differential equation into two parts: a partial differential equation and an ordinary differential equation. The two subproblems, the first representing the diffusion of f and the second representing the source of f, can be solved in an alternating fashion that is mathematically equivalent to solving the whole system (Toro, 1999). A second order accurate method for this approach can be written as,

$$f^{t+\Delta t} = S^{\frac{\Delta t}{2}} C^{\Delta t} S^{\frac{\Delta t}{2}} f^t \tag{13}$$

where f is the phase space density, $S^{\frac{\Delta t}{2}}$ is the operator that advances the source sub-problem forward in time by one-half time step, and $C^{\Delta t}$ is the Crank-Nicolson operator that advances the diffusion subproblem forward in time a full time step. Such split-
5 ting yields a powerful, generalized algorithm for tackling complicated source terms but requires additional considerations not previously taken into account, namely careful selection of the source term solver.

Figure 12 illustrates the impact of the selection of source solver if splitting were applied to the artificial source terms used in this work. Source splitting is not necessary
10 for Q_{MMS} as it is only a function of t and L, but is applied to illustrate the complications that arise when it is used. The blue line labeled Case 1 shows that employing a simple, second order accurate trapezoidal integration to solve the source subproblem yields the desired results: a model that converges cleanly with the prescribed accuracy. But what if a higher accuracy quadrature method is used? Case 2 uses the fourth-order accu-
15 rate Simpson's Rule to integrate the source term; the results are displayed as the green line in Fig. 12. The error is reduced from Case 1, however the code still converges in a second order manner. This is because less error is introduced from the integration of the source term but the total error is still dominated by the Crank-Nicholson solver and converges along that lower-order rate. Finally, Case 3 (red line) is identical to Case 2
20 but the magnitude of D_{LL} is reduced by an order of magnitude. Because the rate of diffusion is smaller, a larger time step is better suited to the problem and the overall error should drop. This proves true for $\Delta t < 30.0\,\text{s}$ in Fig. 12, but for larger time steps, the error quickly rises to match Case 2. What is observed is the two solution operators trading the bulk of the error. When the source term solver accounts for the greater
25 portion of the error, the code converges with fourth order accuracy. Because this drops below the error inherent in the diffusion operator quickly, the code then begins to converge more slowly, and all three lines become parallel. All three of these curves result from slight variations of a Crank-Nicolson, split-source term implementation.

Radbelt verification

D. T. Welling et al.

Printer-friendly Version

Interactive Discussion

This example illustrates the additional complexities that must be taken under consideration when more complicated source and loss terms are included. The selected source integration scheme has dramatic implications on the overall error. Changing diffusion rates, which happens commonly when the BA2000 model is used, can push the model into different error regimes and change the behavior of the model. These effects must be kept in mind as the model is developed in the future.

4 Conclusions

This work introduced and verified the RadBelt radial diffusion model that is part of the SpacePy software library. The method of manufactured solutions was employed to provide an analytical solution where one could not be trivially obtained. The model's flexibility, especially in terms of choice of D_{LL} and sources, was put to many rigorous tests. It was found that the solver and many code features have been implemented correctly and robustly.

The results here are of special interest to code users. Repeatedly, it was demonstrated that code convergence is beholden to the choice of D_{LL} and the rate of change of the source function. While selecting Δt and ΔL, it is easy to find oneself in a regime where refining one has little impact on the final error. How the source and diffusion coefficient affect these semi-stagnant convergence regimes is vital knowledge for code users who want to maximize code performance while minimizing error.

Finally, it must be emphasized that performance in terms of convergence regimes and overall error is situation specific. The time and grid discretizations used in this study are far smaller than the typical scientific user will require. It is suggested that, when employing new source, loss or diffusion coefficient models, the user perform a quick study to ensure proper refinement for the most extreme situation they choose to encounter.

References

Baker, D.: The Occurrence of Operational Anomalies in Spacecraft and Thei Relationship to Space Weather, IEEE Trans. Plasma Sci., 28, 2007–2016, 2000. 2166

Beutier, T. and Boscher, D.: A three-dimensional analysis of the electron radiation belt by the Salammbô code, J. Geophys. Res., 100, 14853–14862, doi:10.1029/94JA03066, 1995. 2168

Brautigam, D. H. and Albert, J. M.: Radial diffusion analysis of outer radiation belt electrons during the October 9, 1990, magnetic storm, J. Geophys. Res., 105, 291–310, doi:10.1029/1999JA900344, 2000. 2167, 2170, 2187

Carpenter, D. L. and Anderson, R. R.: An ISEE/Whistler model of equatorial electron density in the magnetosphere, J. Geophys. Res., 97, 1097–1108, doi:10.1029/91JA01548, 1992. 2170

Crank, J. and Nicolson, P.: A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type, Proc. Cambridge Philos. Soc., 50–67, 1947. 2171

Fälthammar, C.-G.: Effects of Time-Dependent Electric Fields on Geomagnetically Trapped Radiation, J. Geophys. Res., 70, 2503–2516, doi:10.1029/JZ070i011p02503, 1965. 2167

Fei, Y., Chan, A. A., Elkington, S. R., and Wiltberger, M. J.: Radial diffusion and MHD particle simulations of relativistic electron transport by ULF waves in the September 1998 storm, J. Geophys. Res. (Space Physics), 111, A12209, doi:10.1029/2005JA011211, 2006. 2170

Feynman, J. and Gabriel, S. B.: On space weather consequences and predictions, J. Geophys. Res., 105, 10543–10564, doi:10.1029/1999JA000141, 2000. 2166

Fok, M.-C., Horne, R. B., Meredith, N. P., and Glauert, S. A.: Radiation Belt Environment model: Application to space weather nowcasting, J. Geophys. Res. (Space Physics), 113, A03S08, doi:10.1029/2007JA012558, 2008. 2168

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

|◄ | ►|

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Gubby, R. and Evans, J.: Space environment effects and satellite design, J. Atmos. Sol.-Terr. Phys., 64, 1723–1733, 2002. 2166

Koller, J. and Morley, S.: Magnetopause shadowing effects for radiation belt models during high-speed solar wind streams, AGU Fall Meeting Abstracts, A1787+, 2010. 2170

Koller, J., Chen, Y., Reeves, G. D., Friedel, R. H. W., Cayton, T. E., and Vrugt, J. A.: Identifying the radiation belt source region by data assimilation, J. Geophys. Res. (Space Physics), 112, A06244, doi:10.1029/2006JA012196, 2007. 2168

Koons, H. C., Mazur, J. E., Selesnick, R. S., Blake, J. B., and Fennell, J. F.: The Impact of the Space Environment on Space Systems, NASA STI/Recon Technical Report N, 69036–69041, 1999. 2166

Lam, M. M., Horne, R. B., Meredith, N. P., and Glauert, S. A.: Modeling the effects of radial diffusion and plasmaspheric hiss on outer radiation belt electrons, Geophys. Res. Lett., 34, L20112, doi:10.1029/2007GL031598, 2007. 2167

Lyons, L. R. and Schulz, M.: Access of energetic particles to storm time ring current through enhanced radial 'diffusion', J. Geophys. Res., 94, 5491–5496, doi:10.1029/JA094iA05p05491, 1989. 2167

Millan, R. M. and Thorne, R. M.: Review of radiation belt relativistic electron losses, J. Atmos. Sol.-Terr. Phys., 69, 362–377, doi:10.1016/j.jastp.2006.06.019, 2007. 2170

Miyoshi, Y., Morioka, A., Misawa, H., Obara, T., Nagai, T., and Kasahara, Y.: Rebuilding process of the outer radiation belt during the 3 November 1993 magnetic storm: NOAA and Exos-D observations, J. Geophys. Res. (Space Physics), 108, 1004, doi:10.1029/2001JA007542, 2003. 2167

Morley, S. K., Koller, J., Welling, D. T., and Larsen, B. A.: Spacepy – A Python-based library of tools for the space sciences, in: Proceedings of the 9th Python in science conference (SciPy 2010), Austin, TX, in press, 2011. 2169

Pirjola, R., Kauristie, K., Lappalainen, H., and Viljanen, A.: Space weather risk, Space Weather, 3, S02A02, doi:10.1029/2004SW000112, 2005. 2166

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P.: Numerical Recipes, 3rd Edn., Cambridge University Press, New York, NY, 2007. 2171, 2172

Qin, Z., Denton, R. E., Tsyganenko, N. A., and Wolf, S.: Solar wind parameters for magnetospheric magnetic field modeling, Space Weather, 5, S11003, doi:10.1029/2006SW000296, 2007. 2169

Reeves, G. D., Friedel, R. H. W., Bourdarie, S., Thomsen, M. F., Zaharia, S., Henderson,

M. G., Chen, Y., Jordanova, V. K., Albright, B. J., and Winske, D.: Toward Understanding Radiation Belt Dynamics, Nuclear Explosion-Produced Artificial Belts, and Active Radiation Belt Remediation: Producing a Radiation Belt Data Assimilation Model, Washington DC American Geophysical Union Geophysical Monograph Series, 159, 221–235, 2005. 2168

5 Roache, P. J.: Verification and Validation in Computational Science and Engineering, Hermosa, Albuquerque, NM, 1998. 2172, 2176

Roache, P. J.: Code Verification by the Method of Manufactured Solutions, J. Fluid. Eng., 124, 4–10, doi:10.1115/1.1436090, 2002. 2172

Roederer, J. G.: Dynamics of geomagnetically trapped radiation, 1970. 2167

10 Schulz, M. and Lanzerotti, L.: Particle Diffusion in the Radiation Belts, Spring-Verlag, New York, 1974. 2167

Selesnick, R. S., Blake, J. B., Kolanski, W. A., and Fritz, T. A.: A quiescent state of 3 to 8 MeV radiation belt electrons, Geophys. Res. Lett., 24, 1343, doi:10.1029/97GL51407, 1997. 2170

Shprits, Y. Y., Thorne, R. M., Friedel, R., Reeves, G. D., Fennell, J., Baker, D. N., and Kanekal,

15 S. G.: Outward radial diffusion driven by losses at magnetopause, J. Geophys. Res. (Space Physics), 111, A11214, doi:10.1029/2006JA011657, 2006. 2167

Shprits, Y. Y., Elkington, S. R., Meredith, N. P., and Subbotin, D. A.: Review of modeling of losses and sources of relativistic electrons in the outer radiation belt I: Radial transport, J. Atmos. Sol.-Terr. Phys, 70, 1679–1693, doi:10.1016/j.jastp.2008.06.008, 2008. 2167

20 Strang, G.: On the Construction and Comparison of Difference Schemes, SIAM J. Numer. Anal., 5, 506–517, 1968. 2178

Tadjeran, C.: Stability analysis of the Crank-Nicholson method for variable coefficient diffusion equation, Commun. Numer. Meth. En., 23, 29–34, doi:10.1002/cnm.879, 2007. 2171

Toro, E. F.: Riemann solvers and numerical models for fluid dynamics, Springer-Verlag, New

25 York, 1999. 2178

Tsyganenko, N. A.: A model of the near magnetosphere with a dawn-dusk asymmetry - 1. Mathematical Structure, J. Geophys. Res., 107, 1179–1195, doi:10.1029/2001JA000219, 2002. 2170

Tsyganenko, N. A., Singer, H. J., and Kasper, J. C.: Storm-time distortion of the inner

30 magnetosphere: How severe can it get?, J. Geophys. Res. (Space Physics), 108, 1209, doi:10.1029/2002JA009808, 2003. 2170

Ukhorskiy, A. Y. and Sitnov, M. I.: Radial transport in the outer radiation belt due to global magnetospheric compressions, J. Atmos. Sol.-Terr. Phys., 70, 1714–1726,

Radbelt verification

D. T. Welling et al.

Printer-friendly Version

Interactive Discussion

doi:10.1016/j.jastp.2008.07.018, 2008. 2170

van Allen, J. A. and Frank, L. A.: Radiation Around the Earth to a Radial Distance of 107,400 km, Nature, 183, 430–434, doi:10.1038/183430a0, 1959. 2166

Wang, H., Ridley, A. J., and Luhr, H.: Validation of the Space Weather Modeling Framework using observations from CHAMP and DMSP, Space Weather, 6, doi:10.1029/2007SW000355, 2008. 2168

Welling, D. T.: The long-term effects of space weather on satellite operations, Ann. Geophys., 28, 1361–1367, doi:10.5194/angeo-28-1361-2010, 2010. 2166

Welling, D. T. and Ridley, A. J.: Validation of SWMF magnetic field and plasma, Space Weather, 8, doi:10.1029/2009JA014596, 2010. 2168

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Full Screen / Esc

Table 1. A comparison of the different D_{LL} formulations used in this study in terms of α and β.

Formulation	α	β
S1997	1.9×10^{-10}	11.7
BA2000	$10^{0.506 K_P - 9.325}$	10.0
FC2006	1.5×10^{-6}	8.5
U2008	7.7×10^{-6}	6.0
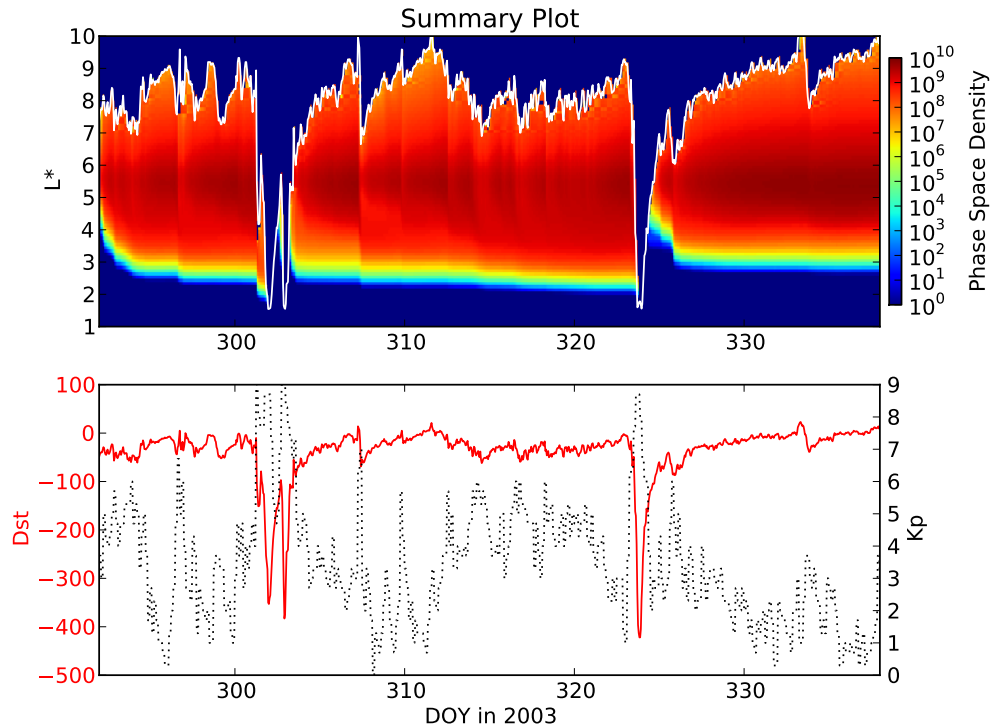
Fig. 1. Example output from a RadBelt simulation of the well known "Halloween Storms". The top frame shows phase space density with L_{max} location over plotted in white. The bottom frame shows the geomagnetic indices used to drive D_{LL} and the empirical source and loss functions.
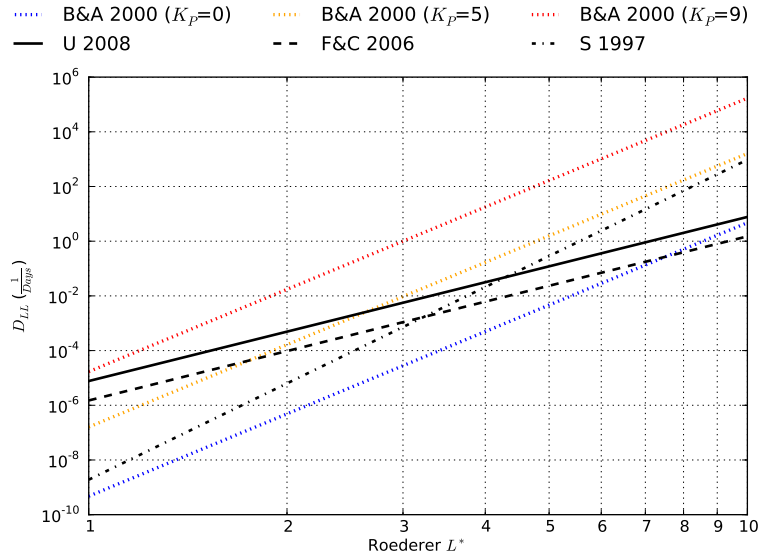
Fig. 2. Comparison of the four formulations of D_{LL} used in this study. Because the formulation of Brautigam and Albert (2000) is K_p dependent, it is shown for several different K_p values.
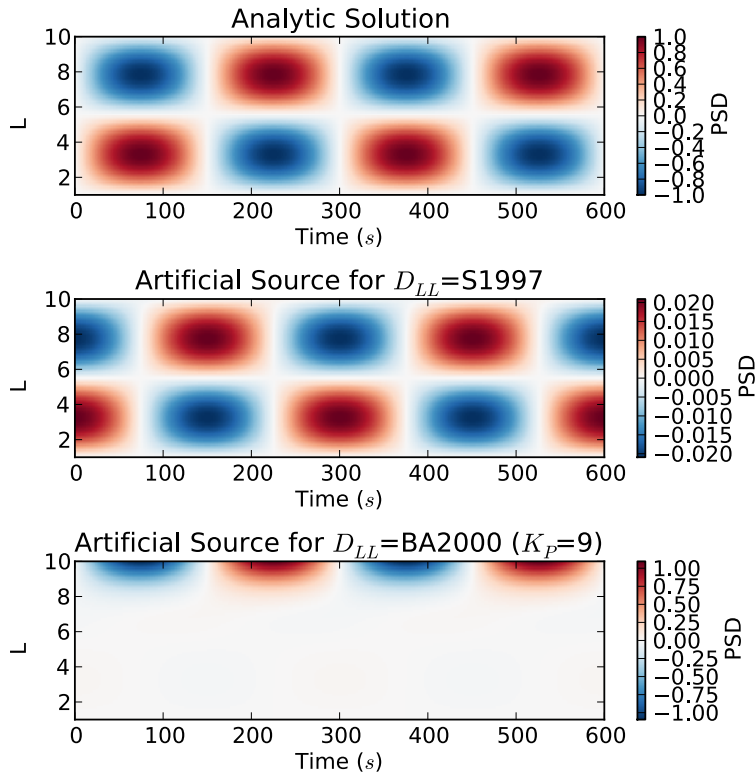
Fig. 3. Behavior of the manufactured solution (top frame), Q_{MMS} when S1997 D_{LL} is employed (center frame), and Q_{MMS} when BA2000 $D_{LL}(K_P = 9)$ is used (bottom frame). Following Eq. (8), for a smaller α factor, the source closely resembles the analytic solution, but scaled and phase-shifted (center frame). As α becomes stronger, the source becomes more complicated to counteract the strong diffusion at high L (bottom frame).

Title Page

Abstract | Introduction

Conclusions | References

Tables | Figures

|◄ | ►|

◄ | ►

Back | Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion
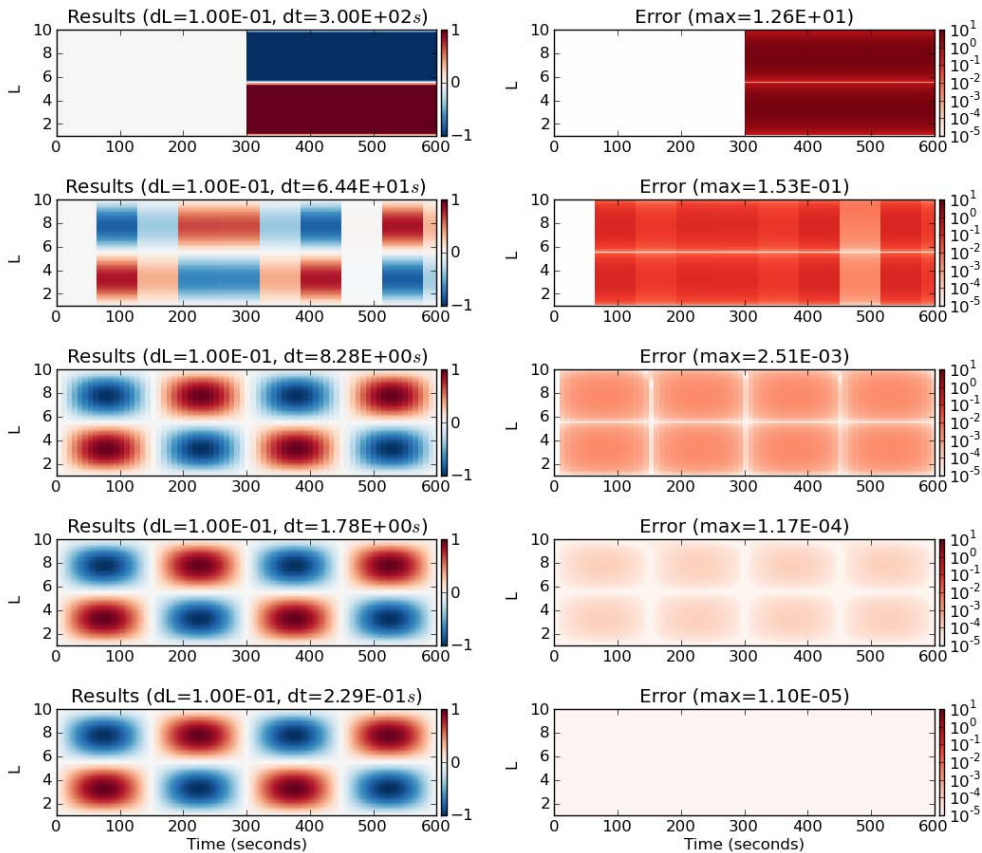
Fig. 4. A demonstration of the code converging towards the analytical solution as the time step is refined. The left column shows the numerical solution while the right column shows the corresponding error. Because the maximum error, listed in the title of each plot, reduces as the square of the time step, this figure hints at the expected convergence rate given the numerical scheme used.
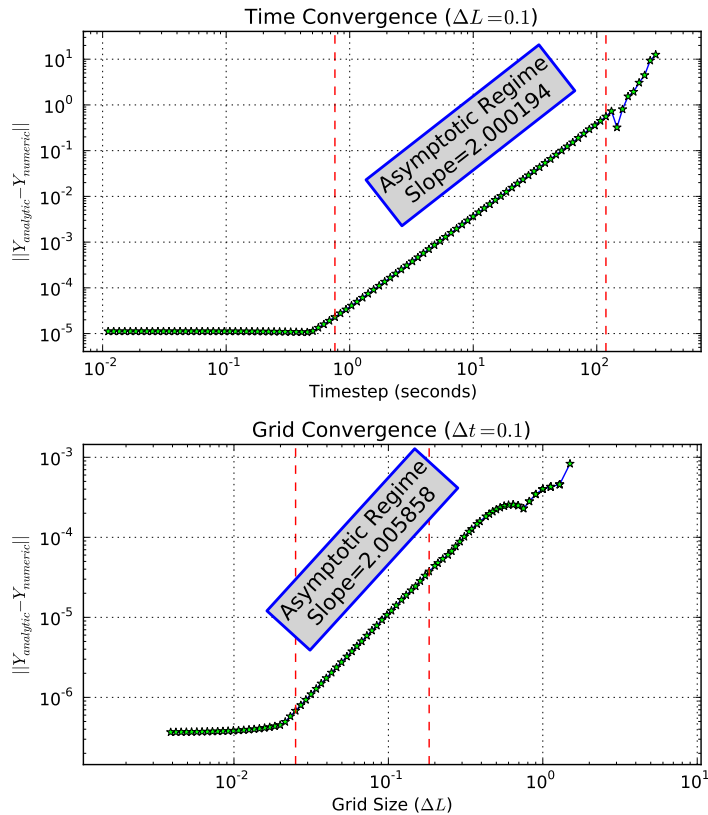
Fig. 5. Top frame (bottom frame): convergence as Δt (ΔL) is refined. Each star represents the results from an individual simulation. The approximate location of the asymptotic regime is delimited by vertical red dashed lines. The slope of the line indicates the order of convergence; for each frame the correct, expected behavior is observed.
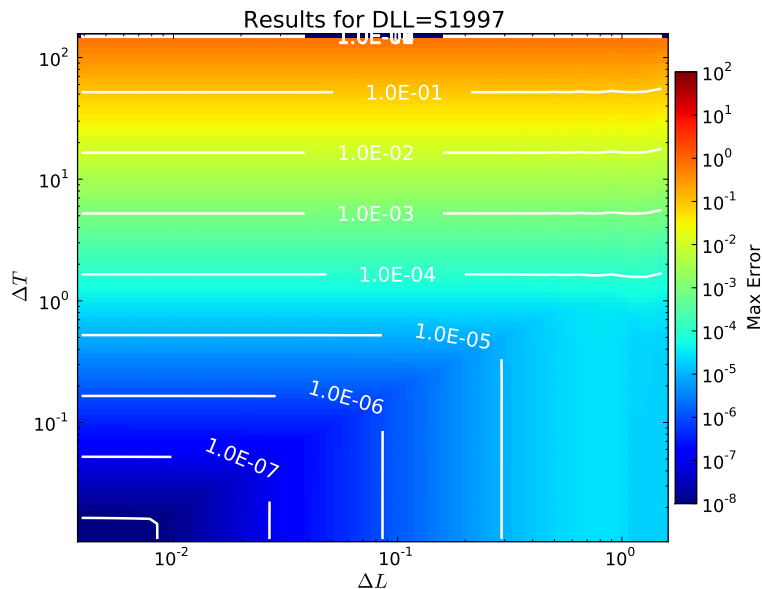
Fig. 6. Error map for 8000 simulations as a function of time step (vertical axis) and grid size (horizontal axis). White lines denote contours of constant error at each order-of-magnitude. These results were obtained using the S1997 D_{LL} model. A cut of this map along a line of constant ΔL (Δt) will yield a curve similar to what is seen in the top (bottom) frame of Fig. 5 with the slope indicating the order of convergence. The proper behavior is observed here over a large range of time steps and grid sizes.
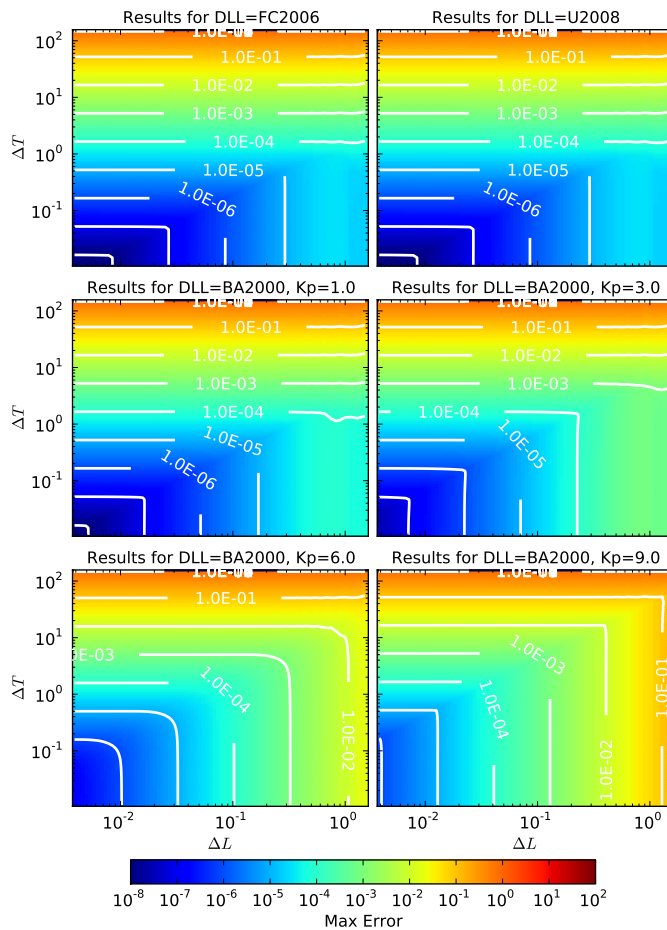
Fig. 7. Similar to Fig. 6, but for other D_{LL} models.
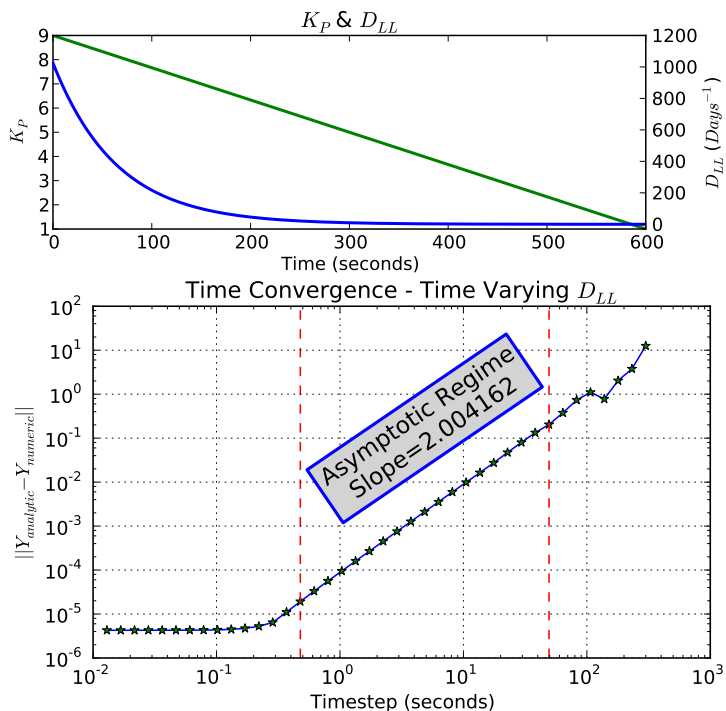
Fig. 8. Results for time verification when using a time-varying D_{LL}. $\Delta L = 0.01$ for this set. The top frame shows the values of K_p (green line) and BA2000 D_{LL} (blue line). The bottom frame shows the expected convergence of the model in the same manner as the top frame of Fig. 5.
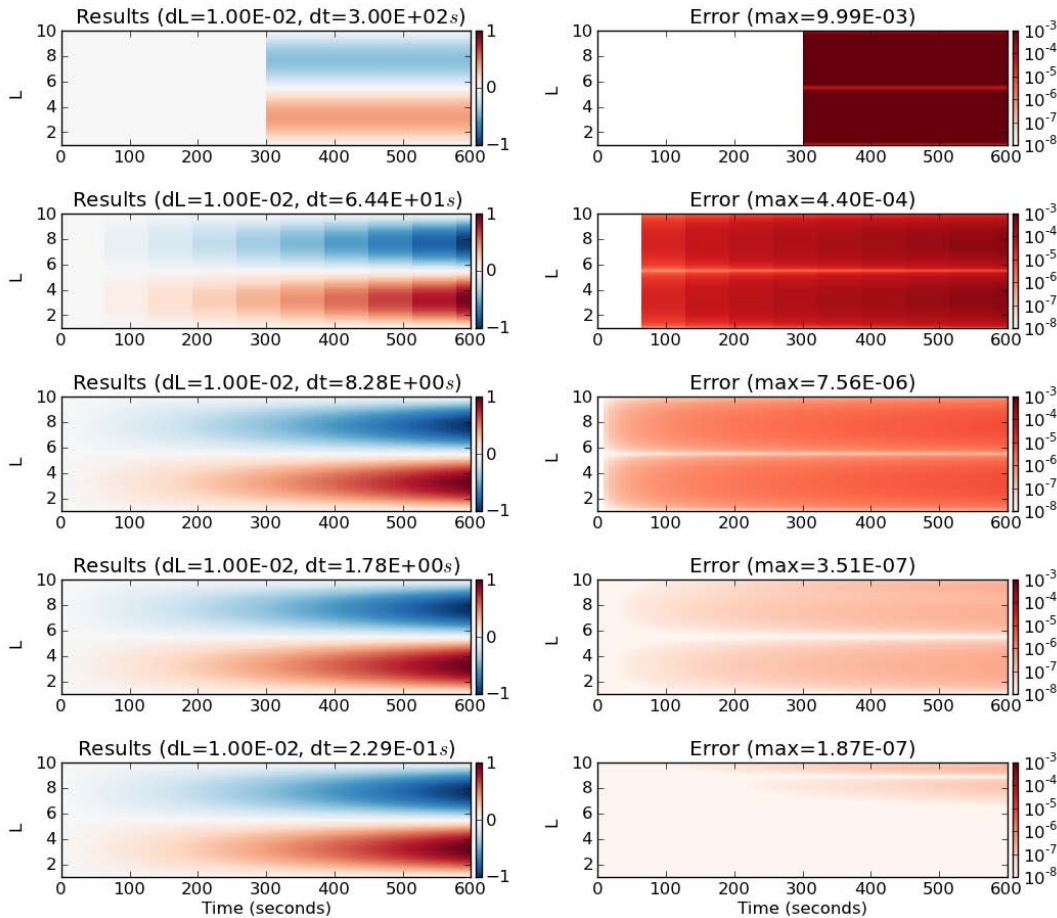
Fig. 9. Similar to Fig. 4, but showing results when the second, non-periodic manufactured solution is used (Eq. 9).
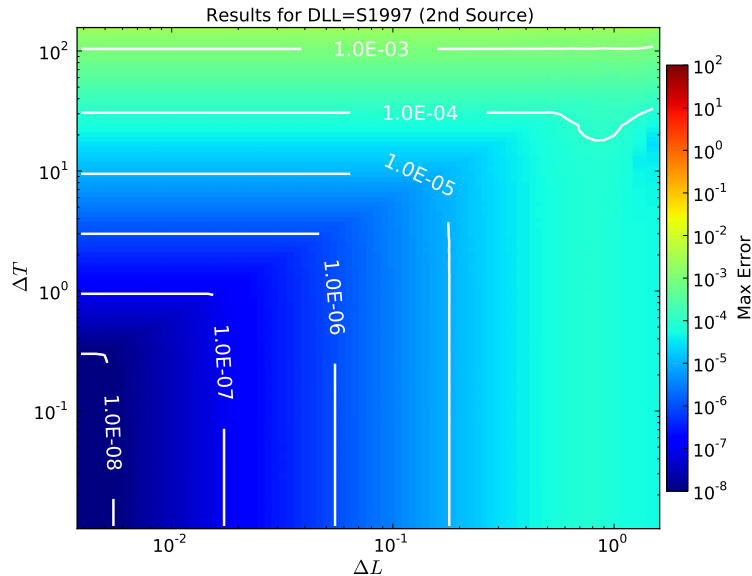
Fig. 10. Similar to Fig. 6 but showing results when the second, non-periodic manufactured solution is used (Eq. 9).
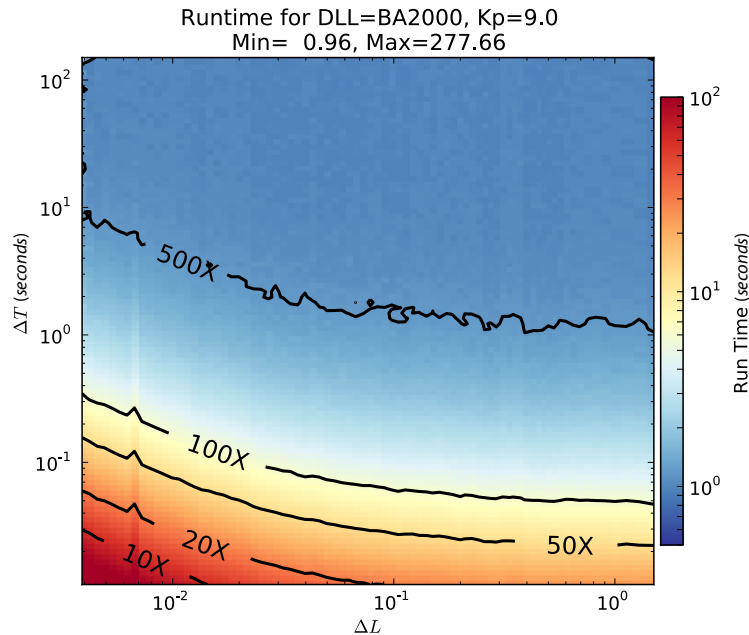
Fig. 11. Run speed benchmark for all 8000 simulations performed using BA2000 $D_{LL}(K_P = 9)$. The color map shows raw simulation completion time (seconds) while black contour lines show the ratio of simulation time to CPU time. For example, for a total simulation time of 600 s and a CPU run time of 6 s (near-white on the color scale), the code simulates the radiation belts 100 times faster than real time and has a simulation-to-CPU time ratio (black contour lines) of 100X.

Full Screen / Esc
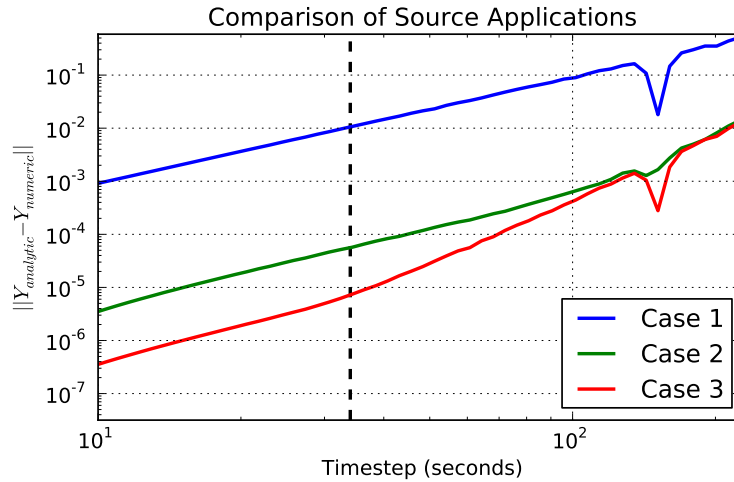
Printer-friendly Version

Interactive Discussion

Fig. 12. Convergence curves for three separate simulations that split out the source term from the Crank-Nicolson solver. Case 1 uses a simple second-order accurate source solver, while Cases 2 and 3 use a fourth order source solver. The variety in the curves demonstrate the importance of properly selecting a solver along with an appropriate grid size and time step.