Active Shielding Particle Pusher (ASPP): Charged-Particle Tracking Through Electromagnetic Fields

D. Fry
Space Radiation Analysis Group
NASA Johnson Space Center, Houston TX

M. Lund
Nuclear Engineering Program
University of Utah, Salt Lake City, UT

A. A. Bahadori, R. Pal. Chowdhury and L. Stegeman
Alan Levin Department of Mechanical and Nuclear Engineering
Kansas State University, Manhattan, KS

S. Madzunkov
Planetary Mass Spectrometry
JPL, Pasadena, CA
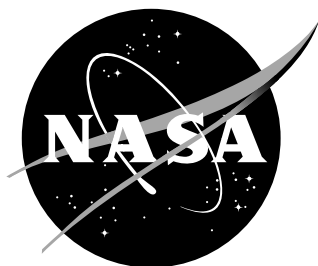
NASA STI Program . . . in Profile

NASA/TP–2020–5002408

Active Shielding Particle Pusher (ASPP): Charged-Particle Tracking Through Electromagnetic Fields

D. Fry
Space Radiation Analysis Group
NASA Johnson Space Center, Houston TX

M. Lund
Nuclear Engineering Program
University of Utah, Salt Lake City, UT

A. A. Bahadori, R. Pal. Chowdhury and L. Stegeman
Alan Levin Department of Mechanical and Nuclear Engineering
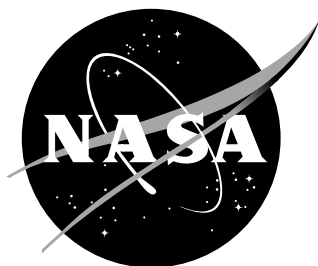*Kansas State University, Manhattan, KS*

S. Madzunkov
Planetary Mass Spectrometry
JPL, Pasadena, CA

National Aeronautics and
Space Administration

Johnson Space Center

May 2020

Acknowledgments

This report is available in electronic form at
http://

Abstract

Contents

List of Figures

4

1 Introduction and Purpose

The search for an effective, feasible solution to radiation protection in space using electromagnetic fields, i.e. active shielding, has a lengthy history spanning more than five decades. There is no question of the impact this method of shielding could have on the future of human space exploration. The risk of Radiation Exposure Induced Death (REID) resulting from astronaut radiation exposure is a substantial risk that must be dealt with if humans are ever going to progress beyond Low Earth Orbit (LEO) for extended periods of time.

Passive shielding techniques are not a viable solution for mitigating exposure from galactic cosmic rays (GCR). GCR kinetic energies often exceed a few GeV per nucleon and may interact with matter via fragmentation, which results in lighter ions with greater ranges. Passive shielding required to shield GCR is mass prohibitive; it is not possible to build, launch, and operate space vehicles with shielding materials that are several meters in thickness.

This report outlines development of a source code package, titled Active Shielding Particle Pusher (ASPP) for tracking charged particles through electromagnetic fields. The prime development objective was to produce a tool enabling fast, 'rapid-prototyping' of active shielding field configurations utilizing the latest technology in graphical processing unit (GPU) architecture to drastically lower computation time. Moreover, the code is structured to enable users to add additional field types in a plug-and-play fashion, allowing the code to expand in capability over time. This has the added benefit of expanding the utility of the package beyond active shielding studies into other applications where particle tracking simulations in electromagnetic fields is needed.

This report is organized as follows. We first give a brief overview of relativistic dynamics pertaining to a charged particle moving in the presence of electromagnetic fields. This is done in a general fashion from the perspective of the covariant and contravariant tensor forms of electrodynamic fields, including the radiation reaction. Assumptions are presented on simplifying the resultant equations to derive a generalized set of particle equations of motion. Justification of assumptions is presented in the appendices sections. We then discuss the development of the ASPP source code for numerically integrating the particle equations of motion, including integration routines, specific applicability of each integration method, particle initialization, a collection of analytic field calculation routines included in the source code, and methods to ingest user-defined fields on a Cartesian grid. SI units are used throughout. Table 1 lists conventions used to describe the characteristics of a charged particle and the electromagnetic fields. Baseline testing results of source code are included in the Appendices along with code compilation specifics, sample input files, and an overview of past active shielding studies for completeness.

Table 1. Standard conventions used to describe a charged particle and electromagnetic fields.

Rest-mass Energy	E_0
Speed of light in vacuum	c
Permittivity of Free Space	ϵ_0
Permeability of Free Space	μ_0
Electron Charge	e
Atomic Number	Z
Atomic Mass Number	A
Number of Nucleons	n
Non-Relativistic Mass	m
Position 3-vector	$\vec{r}(t) = (x(t), y(t), z(t))$
Position 4-vector	$\vec{x}(t) = (ct, x(t), y(t), z(t))$
Relativistic Momentum 3-vector	$\vec{p}(t)$
Kinetic Energy	E_k
Lorentz Factor	γ
Relativistic Velocity 3-Vector	$\vec{u}(\vec{r},t), \vec{v}(\vec{r},t)$
Relativistic Mass	γm
Relativistic Velocity 4-Vector	$\vec{U} = (u_0, \vec{u})$
Covariant Velocity 3-Vector	u_α
Contravariant Velocity 3-Vector	u^α
Covariant Momentum 4-Vector	p_α
Contravariant Momentum 4-Vector	p^α
Kinetic Energy Units	MeV/n
Electric Field	$\vec{E}(\vec{r},t)$
Magnetic Field	$\vec{B}(\vec{r},t)$
Magnetic Dipole Moment	$\vec{M}$
Electric Dipole Moment	$\vec{M}_e$

2 Generalized Equations of Motion

The goal of the developed toolset is to enable accurate calculation of particle trajectories, and hence spatially-dependent particle flux density as a function of the 3d field distribution, in a manner that enables rapid iteration over many field configurations. This translates to source code that must run within an order of hours instead of days per single run, with high accuracy, including relativistic effects.

For relatively low incident particle energy, dynamics can be successfully modeled with classical theory, i.e. non-relativistic Newtonian dynamics. For a particle of kinetic energy E_k and speed v, this regime can be characterized by the Lorentz factor

$$\gamma = \frac{1}{\sqrt{1 - \frac{v^2}{c^2}}} = 1 + E_k/E_0 \tag{1}$$

where c is the speed of light in vacuum. In the limit $\gamma \approx 1$, or $E_k \ll E_0$, Newtonian dynamics sufficiently describes the particle physics.

It should be noted that there is not a discrete cut between Newtonian and relativistic dynamics. Consider Figure (1), where the relativistic speed as a function of incident proton energy in MeV is shown along with the Lorentz factor γ (inset). It is not until proton energy exceeds 10 MeV that γ begins increasing substantially from a value of unity and the proton speed begins to deviate from classical $E^{1/2}$ behavior. For this reason, we assume that relativistic dynamics must be utilized for the entire proton energy range of interest related to human space flight (≥ 10 MeV), and generalize it to include all ions and species of kinetic energy $E_k \geq 10$ MeV/n. The question then arises as to what forces should be included in the description of particle motion. For motion in an external electric ($\vec{E}$) and a magnetic field ($\vec{B}$), the force of each field on the particle must be used to describe the resultant dynamics. For completeness, the radiation reaction force, $\vec{f}$ must also be included, although it can in many cases be neglected. This force manifests due to the acceleration of the charged particle in an external field, and its own self-field, and subsequent recoil due to it radiating away some of its energy. It is a highly non-trivial force term to include in particle dynamics due to the singular nature of the Coulomb force.

Study of the radiation reaction problem has a long history dating back roughly nine decades. The problem was first addressed classically by Abraham, followed by Lorentz, and then by Dirac. Relativistic treatment that includes the radiation reaction is given by the Lorentz-Abraham-Dirac (LAD) equation. Numerous studies have investigated solutions to this equation. There is no intention here to give an in-depth derivation of the LAD equation or possible solutions. This is left to past reviews [1–12]. In tensor form, the covariant LAD equation for a particle
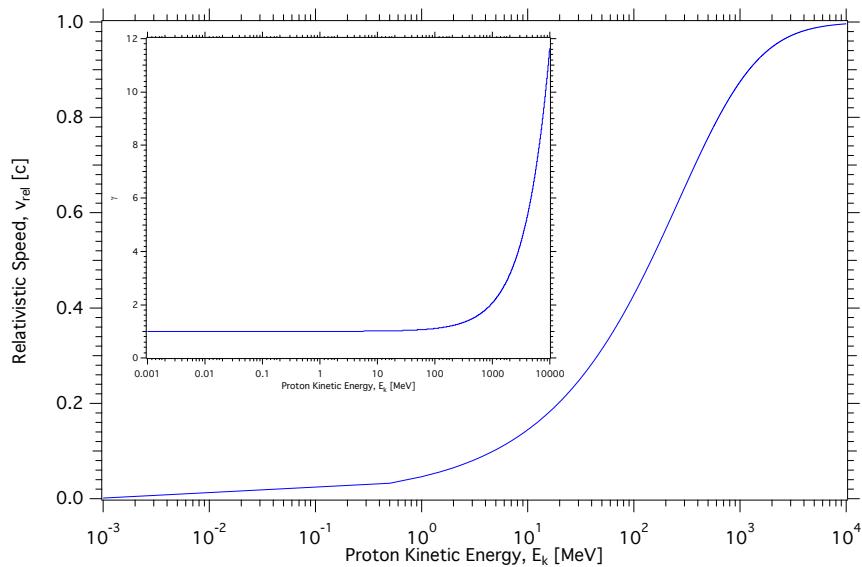
Figure 1. Relativistic speed and Lorentz factor, γ as a function of incident proton kinetic energy, E_k.

of charge q can be written as

$$\frac{dp_\alpha}{d\tau} = F_{\alpha,ext} + \frac{2}{3}q^2\left(g_{\alpha\beta} - p_\alpha p_\beta\right)\frac{d^2p^\beta}{d\tau^2}, \qquad (2)$$

where p_α is the covariant 4-momentum, $g_{\alpha\beta}$ the metric tensor, defined as,

$$g_{\alpha\beta} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1, \end{pmatrix} \qquad (3)$$

and $F_{\alpha,ext}$ the external force tensor due to external electric and magnetic fields. The proper time, τ is defined as

$$\tau = \int \frac{dt}{\gamma} = \int \left(1 - \frac{u^2(t)}{c^2}\right)^{1/2} dt. \qquad (4)$$

The left-hand side of Eq. 2 is the derivative of the relativistic 4-momentum with respect to proper time. The second term in the right-hand side represents the self-interaction of the particle with it's own field, including the reaction force imparted due to radiating away some of its energy. Note that it depends explicitly on the first derivative of the acceleration 4-vector. In what is presented in the next two subsections, all velocities will be represented by the four-velocity $U = (u_0, \vec{u}) = (u_0, u_1, u_2, u_3)$ where the indices $(1, 2, 3)$ correspond to the (x, y, z) components respectively, q the particle charge in units of elementary electric charge e, E_k

10

the particle kinetic energy, E_0 the particle rest-mass energy, t is the normal time, and τ is the proper time. Note that although it is common to denote the velocity vector as $\vec{v}$, we are reserving that notation for the three-dimensional velocity used to formulate the particle equations of motion. $\vec{u}$ is commonly used to denote the relativistic velocity four-vector. We first proceed by neglecting the radiation reaction force. Justification for this assumption will be discussed in detail in Appendix D2.

2.1 Covariant Form of the Lorentz Force Law

A proper description of particle motion can only be done within the construct of full 3d relativistic dynamics for the high-energy particles that must be considered in any active shielding study. For a charged particle of charge q and mass m moving in the presence of an electromagnetic field, the equations describing its motion can be written in covariant form as

$$m\frac{dU^\alpha}{d\tau} = qF^{\alpha\beta}U_\beta, \tag{5}$$

where $U_\beta(\beta = 0, 1, 2, 3)$ is the covariant four-velocity and $F^{\alpha\beta}$ is the contravariant electromagnetic field tensor. Standard tensor notation is used throughout where covariant tensors are denoted with a subscript (α, β) and contravariant with a superscript. Note that both 3- and 4-vectors are tensors of rank one. We assume that the electromagnetic field can be time dependent and magnetic and electric components can be expressed in terms of a magnetic vector potential, $\vec{A}$ and scalar potential, ϕ, as

$$\vec{B}(x, y, z, t) = \nabla \times \vec{A}(x, y, z, t) \tag{6}$$

$$\vec{E}(x, y, z, t) = -\nabla\phi - \frac{\partial \vec{A}}{\partial t}. \tag{7}$$

The total energy of a particle of mass m is given by

$$E = \gamma mc^2 = \frac{mc^2}{\sqrt{1 - u^2/c^2}} = \sqrt{c^2p^2 + m^2c^4}, \tag{8}$$

where $\vec{u} = c^2\vec{p}/E$ and the invariant length of the momentum four-vector is

$$p_0^2 - \vec{p}\cdot\vec{p} = p_\alpha p^\alpha = m^2c^2, \tag{9}$$

where $p_\alpha p^\alpha$ is the inner product of the relativistic momentum 4-vector. The contravariant electromagnetic field tensor is

$$F^{\alpha\beta} = \begin{pmatrix} 0 & -E_x/c & -E_y/c & -E_z/c \\ E_x/c & 0 & -B_z & B_y \\ E_y/c & B_z & 0 & -B_x \\ E_z/c & -B_y & B_x & 0 \end{pmatrix} \tag{10}$$

For non-relativistic particles, the equation of motion in the presence of external electric and magnetic fields is given by the Lorentz force

$$\frac{d\vec{p}}{dt} = q\left(\vec{E} + \vec{u} \times \vec{B}\right), \tag{11}$$

where $\vec{p} = m\vec{u}$ and $\vec{u}$ are the non-relativistic particle momentum and velocity. This can be transformed in a rather simple fashion to the appropriate relativistic form by transforming $\vec{p}$ to the space part of the four-vector of joint energy and momentum as

$$p^{\alpha} = (p_0, \vec{p}) = m(u_0, \vec{u}) \tag{12}$$

$$u_0 = \frac{dx_0}{d\tau} = \frac{dx_0}{dt}\frac{dt}{d\tau} = \gamma c. \tag{13}$$

Thus, transforming the time derivative as $\gamma d\tau = dt$, the relativistic form of the Lorentz force law is

$$\frac{d\vec{p}}{d\tau} = q\left(\vec{E} + \vec{u} \times \vec{B}\right), \tag{14}$$

where now $\vec{p} = \gamma m\vec{u}$.

Before proceeding further, a few points should be mentioned. For cases where there is no external electric field component ($\|\vec{E}\| = 0$), Eq. 14 reduces to

$$\frac{d\vec{p}}{dt} = q\vec{u} \times \vec{B}. \tag{15}$$

Taking the dot product between $\vec{p}$ and both sides of Eq. 15

$$\vec{p} \cdot \frac{d\vec{p}}{dt} = q(\vec{p} \cdot (\vec{u} \times \vec{B})) = 0. \tag{16}$$

This implies that

$$\frac{d}{dt}\frac{p^2}{2m} = 0, \tag{17}$$

and thus the kinetic energy of the particle is conserved. For the case where the electric field component is nonzero this conservation does not hold. For a current density $\vec{J}$ the change in energy E with time is given by

$$\frac{dE}{dt} = \int_V \vec{J} \cdot \vec{E} d^3r. \tag{18}$$

When addressing the case of a single particle, the current density is $\vec{J} = q\vec{u}$, and the time rate of change of kinetic energy is

$$\frac{dE}{dt} = q \int_V \vec{u} \cdot \vec{E} d^3r \tag{19}$$

Thus, any active shielding system that employs only magnetic fields will shield habitable volumes by only redirecting the incoming particles, whereas any system that employs either electric fields only or a combination of electric and magnetic fields will change both the direction and energy of the incoming charged particles.

2.2 Lorentz Force Law Via the Relativistic Lagrangian

These equations can also be derived directly from either the relativistic Lagrangian or Hamiltonian. For a free particle, the Lagrangian cannot depend on particle position. The only Lorentz invariant function of the velocity available is $u_\alpha u^\alpha = c^2$. The relativistic Lagrangian in the absence of external forces is

$$\mathcal{L}_{free} = -\frac{mc^2}{\gamma}. \tag{20}$$

The interaction with an external field is included through the four-vector magnetic potential A^α, where

$$\mathcal{L}_{int} = -\frac{q}{\gamma} u_\alpha A^\alpha = -q\phi + q\vec{u} \cdot \vec{A}. \tag{21}$$

The full relativistic Lagrangian including both the free particle and interaction terms is given by

$$\mathcal{L} = -\frac{mc^2}{\gamma} - q\phi + q\vec{u} \cdot \vec{A}, \tag{22}$$

The ith equation of motion for a charged particle then follows directly from

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}_i}\right) - \frac{\partial \mathcal{L}}{\partial x_i} = 0, \tag{23}$$

where x_i are the generalized coordinates, and in Cartesian space $(x_1, x_2, x_3) = (x, y, z)$.

Consider first the x_1 component. Substituting Eq. 22 into the first term in Eq. 23 yields,

$$\begin{aligned}
\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}_i}\right) &= \frac{d}{dt}(\gamma m u_1) + q\frac{dA_1}{dt} \\
&= \frac{dp_1}{dt} + q\frac{\partial A_1}{\partial t} + q\vec{u} \cdot \nabla A_1, \tag{24}
\end{aligned}$$

where the vector potential has been broken down into components $\vec{A} = (A_1, A_2, A_3)$ and the convective derivative$(d/dt = \partial/\partial t + \vec{u} \cdot \nabla)$ has been used to recast dA_1/dt. The second term in Eq. 23 yields,

$$\frac{\partial \mathcal{L}}{\partial x} = -q\frac{\partial \phi}{\partial x} + q\left(u_1\frac{\partial A_1}{\partial x} + u_2\frac{\partial A_1}{\partial y} + u_3\frac{\partial A_3}{\partial x}\right). \tag{25}$$

Equating (24) and (25), and noting that the first term in Eq. (24) is the x-component of the force, F_x,

$$\begin{aligned}
F_x &= q\left(-\frac{\partial \phi}{\partial x} - \frac{\partial A_1}{\partial t}\right) + qu_2\left(\frac{\partial A_2}{\partial x} - \frac{\partial A_1}{\partial y}\right) \\
&\quad - qu_3\left(\frac{\partial A_1}{\partial z} - \frac{\partial A_3}{\partial x}\right). \tag{26}
\end{aligned}$$

13

Noting that

$$\vec{E} = -\nabla\phi - \frac{\partial \vec{A}}{\partial t}, \tag{27}$$

$$\vec{u} \times \vec{B} = \vec{u} \times (\nabla \times \vec{A}), \tag{28}$$

the x-component of the force due to the electromagnetic fields on a charged particle of charge q reduces to

$$F_x = qE_x + q(\vec{u} \times \vec{B})_x, \tag{29}$$

where the subscript x denotes the component of the vector operation. Similar expressions can be derived for F_y and F_z such that the final form for the vector force is

$$\vec{F} = q\left(\vec{E} + \vec{u} \times \vec{B}\right). \tag{30}$$

Since the particulate radiation field of concern is that of positively charged ions Eq. 30 can be re-written in a more standard form conducive to generating the ion equations of motion. We write the Lorentz force (Eq. 30) in standard vector notation using the relativistic momentum, denoting the velocity in terms of the standard three-dimensional form, giving

$$\frac{d\vec{p}}{dt} = Ze\left(\vec{E} + \vec{v} \times \vec{B}\right), \tag{31}$$

where the particle charge, q has been replaced by Ze, the ion atomic number and elementary electron charge. It is this form of the Lorentz force equation that is solved numerically within ASPP.

3   Code Development

The backbone of any particle trajectory simulation is the method used to integrate the equations of motion. The Runge-Kutta (RK) has often been used to solve second-order differential equations. Although algorithms such as the fourth-order RK (RK4) have relatively low cumulative error across integration steps, RK is an inherently non-symplectic integrator, i.e. it does not conserve energy. For the purposes here intrinsic to active shielding where there is zero work done for the case of a purely magnetostatic field on a charged particle and thus energy is conserved, an alternative integrator has been chosen. The Velocity-Verlet (VV) algorithm, used for decades in molecular dynamics simulations including those originally done by Störmer for investigation of the aurora, is ideally suited for active shielding simulations. It has an added advantage over the RK4 algorithm in that it requires only two function calls as opposed to the four needed in RK4, and thus provides a significant reduction in computational time without the sacrifice of accuracy.

Development of source code has been focused around the following guiding principles:

- Portability: to allow for continual adaptation and leveraging of computing technology development, standard ANSI C++ programming architecture has been maintained throughout development.

- Flexibility: there is no reason to suspect that a single field configuration will suffice. Thus all code has been structured to have an inherent capability to ingest any given field geometry with no re-write of fundamental propagation and analysis routines.

- Speed: Due to the nature of the space radiation environment, i.e. large range of incident particle energy, incident particle flux and particle species, simulations containing at least 10^6 particles are required to fully capture deflection statistics. As such, code must be written that utilizes fast array indexing, minimization of function calls, and adaptive step size techniques while maintaining high precision.

3.1 Integration Methods

Any particle dynamics simulation is only as good as the integration method. Simulation methods are required to handle both position and velocity dependent forces. This presents specific challenges in terms of which integration scheme should be chosen. The vast majority of standard integration schemes for second-order Ordinary Differential Equations (ODEs) are limited to spatially-dependent forces that can be expanded in a Taylor series in (x, y, z). There is no issue then for application here as long as the magnetic field component of the Lorentz force (Eq. 31) is set to zero.

Throughout the development of ASPP multiple integration schemes were written and tested. It was decided to incorporate all tested schemes into the final version as a tool for users to assess simulation output characteristics. The six different integrators included are

- Euler: Non-symplectic; only applicable for position-dependent forces; greatest step-size error of all included methods.

- Velocity-Verlet: Symplectic; only applicable for position-dependent forces.

- Adaptive Velocity-Verlet: Symplectic; adapted for velocity-dependent forces by adjusting the step-size dynamically utilizing the cyclotron frequency.

- 4th Order Runge-Kutta: Non-Symplectic; only applicable for position-dependent forces; integration error 5th order in time.

- Boris-Buneman: Non-symplectic but conserves phase space; applicable to both position and velocity-dependent forces;

Nearly all integration schemes employ some form of the finite difference equation. For particle tracking simulations, the generic equations that must be solved are a pair of first-order differential equations of the form

$$\frac{d\vec{r}}{dt} = \vec{u}(\vec{r}, t) \tag{32}$$

$$\frac{d\vec{u}}{dt} = \frac{\vec{F}(\vec{r}, \vec{u}, t)}{m} \tag{33}$$

Each of the integration methods and how they are applied to Eqs. (32) and (33) are discussed in the following sections. The major bottleneck to particle tracking simulations through fields is the number of function calls made during each time step. How this varies from one method to another will be discussed below. To maintain flexibility to add additional configurations in the future as well as reduce computational overhead all field functions are written as function pointers.

Consider the Lorentz force given by Eq. 31. We first re-write this as

$$\frac{d}{dt}(\gamma m \vec{v}) = Ze(\vec{E} + \vec{v} \times \vec{B}). \tag{34}$$

Assume a generalized case where both electric and magnetic fields are present and each can be time-dependent. For non-zero electric fields, the particle kinetic energy, and hence γ, will change as a function of time. Expanding the first-order derivative in time and recalling the explicit dependence of γ on v, Eq. 34 is written as

$$m\vec{v}\frac{d\gamma}{dt} + \gamma(v)m\frac{d\vec{v}}{dt} = Ze(\vec{E}(\vec{r}, t) + \vec{v} \times \vec{B}(\vec{r}, t)). \tag{35}$$

γ is not explicitly dependent on time and thus the derivative of γ is given as

$$\frac{d\gamma}{dt} = \frac{\partial \gamma}{\partial v_x}\frac{dv_x}{dt} + \frac{\partial \gamma}{\partial v_y}\frac{dv_y}{dt} + \frac{\partial \gamma}{\partial v_z}\frac{dv_z}{dt} \tag{36}$$

$$= \frac{v_x}{c^2}\gamma^3\frac{v_x}{dt} + \frac{v_y}{c^2}\gamma^3\frac{v_y}{dt} + \frac{v_z}{c^2}\gamma^3\frac{v_z}{dt} \tag{37}$$

$$= \frac{\gamma^3}{c^2}\vec{v} \cdot \frac{d\vec{v}}{dt}. \tag{38}$$

The total time derivative of relativistic momentum is then,

$$\vec{F} = \frac{d\vec{p}}{dt} = \frac{m\gamma^3}{c^2}\vec{v}\left(\vec{v} \cdot \frac{d\vec{v}}{dt}\right) + m\gamma\frac{d\vec{v}}{dt} \tag{39}$$

$$= m\gamma\vec{a} + \frac{m\gamma^3}{c^2}\vec{v}(\vec{v} \cdot \vec{a}). \tag{40}$$

Note that the first term represents the classical form of Newton's Second Law with a multiplicative factor of γ. The second term couples each of the three component equations in velocity.

16

The above relativistic expression for the force can be decoupled by considering the projection of the force with the particle velocity. Taking the dot product between the velocity vector and vector force,

$$\vec{F} \cdot \vec{v} = \left(\frac{m\gamma^3}{c^2}(\vec{v} \cdot \vec{a})\vec{v} \right) \cdot \vec{v} + m\gamma(\vec{a} \cdot \vec{v}) \tag{41}$$

$$= \frac{m\gamma^3}{c^2}\left((\vec{v} \cdot \vec{a})\vec{v} \cdot \vec{v} \right) + m\gamma(\vec{v} \cdot \vec{a}) \tag{42}$$

$$= \left(\frac{m\gamma^3}{c^2}\vec{v} \cdot \vec{v} \right)(\vec{v} \cdot \vec{a}) + m\gamma(\vec{v} \cdot \vec{a}) \tag{43}$$

$$= \left(\frac{m\gamma^3}{c^2}v^2 + m\gamma \right)(\vec{v} \cdot \vec{a}) \tag{44}$$

$$= m(\vec{v} \cdot \vec{a})\left(\gamma^3 \left(1 - \frac{1}{\gamma^2} \right) + \gamma \right) \tag{45}$$

$$= m\gamma^3(\vec{v} \cdot \vec{a}). \tag{46}$$

Using Eq. (46) and replacing the $(\vec{v} \cdot \vec{a}$ term in Eq. (40),

$$\vec{F} = m\gamma\vec{a} + \frac{m\gamma^3}{c^2}(\vec{v} \cdot \vec{a})\vec{v} \tag{47}$$

$$= m\gamma\vec{a} + (\vec{F} \cdot \vec{v})\frac{\vec{v}}{c^2}. \tag{48}$$

Solving Eq. (48) for $\vec{F}$ and setting equal to the Lorentz force equation yields,

$$m\gamma\vec{a} = \vec{F} - (\vec{F} \cdot \vec{v})\frac{\vec{v}}{c^2} \tag{49}$$

$$= q\vec{E} + q\vec{v}x\vec{B} - q(\vec{E} \cdot \vec{v})\frac{\vec{v}}{c^2} - q(\vec{v}x\vec{B}) \cdot \vec{v}\frac{\vec{v}}{c^2}. \tag{50}$$

Since $(\vec{v}x\vec{B})$ is a vector perpendicular to $\vec{v}$ the last term is zero. The resulting equation to be numerically solved to give the particle trajectory in an external electric and magnetic field is

$$m\gamma\vec{a} = m\gamma\frac{d\vec{v}}{dt} = q\vec{E} + q(\vec{v}x\vec{B}) - q(\vec{E} \cdot \vec{v})\frac{\vec{v}}{c^2}. \tag{51}$$

Solving for the position and velocity of a charged particle then follows from standard techniques of solving second-order ODEs, namely splitting into two first-order ODEs. In what follows, we assume that the equations of motion can be split into the form:

$$\frac{d\vec{v}(\vec{r}, \vec{F}(\vec{r}, \vec{v}, t), t)}{dt} = \vec{F} \tag{52}$$

$$\frac{d\vec{r}(t)}{dt} = \vec{v}(\vec{r}, t), \tag{53}$$

where $\vec{F}(\vec{r}, \vec{v}, t)$ is the corresponding relativistic Lorentz force. In three dimensions, these two equations form a set of six, coupled first-order ODEs that must be numerically solved to map the trajectory of a charged particle.

17

3.1.1 Euler Method

The Euler method is the one of the simplest integration schemes. It utilizes only the first-order term in the Taylor series expansion of the force, $\vec{F}$. As a result, for step size Δt, the error at each step is $\mathcal{O}(\Delta t)$. Each integration step is accomplished by the following:

$$\vec{r}(t + \Delta t) \ = \ \vec{r}(t) + \Delta t \vec{v}(t) + \frac{\Delta t^2}{2}\vec{a}(t), \qquad (54)$$

$$\vec{v}(t + \Delta t) \ = \ \vec{v}(t) + \Delta t \vec{a}(t) \qquad (55)$$

Although inherently a fast algorithm, it suffers from large numerical error. Often, this error is reduced by decreasing the time step at the cost of large increases in run time. It is included here for completeness but not recommended for use in large scale runs. The Boris-Buneman (see below) utilizes an Euler-like step to perform the propagation through combined electric/magnetic field configurations with relatively low error propagation.

3.1.2 Velocity-Verlet (VV)

The Velocity-Verlet algorithm is a commonly used integrator in molecular dynamics simulations [13–15]. It is reversible, symmetric, and symplectic. The method is also referred to as the Störmer-Verlet method because it appeared in early work by Störmer on explaining the Aurora from trajectory simulations of particles moving in Earth's geomagnetic field [16].

The algorithm is similar to the Euler method with the addition of an additional velocity update. It can be derived by keeping terms out to third order in the Taylor series expansion of positions and velocities. By performing the expansion forwards and backwards in time $(t + \Delta t)$ and $(t - \Delta t)$, and subtracting the two, the resulting equation only contains a term second order in t. Thus, the local error is $\mathcal{O}((\Delta t)^4)$ in position and $\mathcal{O}((\Delta t)^2)$ order in velocity. This is considerably better than the Euler method, with minimal additional overhead of two function calls.

The method proceeds by first updating the positions by one time step, Δt. After the positions are updated, the velocity is propagated by a half step $(\Delta t/2)$, and the force (acceleration) is updated with the new positions. The velocity is then updated to a full step value with the new,

half-step update of acceleration. Mathematically this is given by,

$$\vec{a}(t) \quad = \quad \frac{\vec{F}(\vec{r}(t), \gamma(v(t))}{m}, \tag{56}$$

$$\vec{r}(t + \Delta t) \quad = \quad \vec{r}(t) + \Delta t \vec{v}(t) + \frac{\Delta t^2}{2} \vec{a}(\vec{r}(t), t) \tag{57}$$

$$\vec{v}(t + \Delta t/2) \quad = \quad \vec{v}(t) + \frac{\Delta t}{2} \vec{a}(t), \tag{58}$$

$$\vec{a}(t + \Delta t) \quad = \quad \frac{\vec{F}(\vec{r}(t + \Delta t), \gamma(v(t + \Delta t/2)))}{m}, \tag{59}$$

$$\vec{v}(t + \Delta t) \quad = \quad \vec{v}(t + \Delta t/2) + \frac{\Delta t}{2} \vec{a}(t + \Delta t) \tag{60}$$

This method has been implemented in a relativistic fashion by up-dating the Lorentz factor as a two-step process, once after each velocity update (Eqs. 58 and 60) as,

$$\gamma(t + \Delta t/2) \quad = \quad \left(\sqrt{1 - \frac{|\vec{v}(t + \Delta t/2)|^2}{c^2}} \right)^{-1} \tag{61}$$

$$\gamma(t + \Delta t) \quad = \quad \left(\sqrt{1 - \frac{|\vec{v}(t + \Delta t)|^2}{c^2}} \right)^{-1} \tag{62}$$

Caution must be stressed here when used with velocity-dependent forces. Note that the force function ($\vec{F}$) in the above routine is expressed as only a function of position ($\vec{r}$), as it should be for the standard Velocity-Verlet algorithm. This can produce large errors when using for velocity-dependent forces as will be discussed in Appendix C.1.

3.1.3   Adaptive Velocity-Verlet (AVV)

The AVV algorithm is identical the VV method with the exception of modification of Δt with the cyclotron frequency. For external magnetic fields $\vec{B}$ the field is first calculated at the current position, $\vec{r}(t)$. The time step is then calculated from the inverse cyclotron frequency as

$$\Delta t = 0.001 \frac{2\pi \gamma m}{e|\vec{B}|}. \tag{63}$$

The prefactor of 0.001 has been added to ensure that any move in po-sition translates the particle at most $0.001 R_c$ where R_c is the cyclotron radius, ensuring high-accuracy in high magnetic field regions by limiting the step size in time. By restricting the time step, the resulting tra-jectory in regions where momentum transfer is high will be of sufficient resolution. This can result in an increase in total simulation run time but provides a way to ensure the trajectory is captured without large spatial steps. In regions of low magnetic field strength it is possible that

19

Eq. 63 could result in very large Δt values causing large jumps in particle trajectories between steps. To avoid this the modified step size is used only if $\Delta t < \Delta t_0$ where Δt_0 is the step size set by the user. The precise value of the prefactor is somewhat arbitrary. The value of 0.001 was chosen to yield enough resolution in the trajectory to be able to visualize with common plotting routines when running simulation where the magnetic field intensity is on the order of several Tesla. Note that the same cautions hold for this algorithm as for the VV algorithm with respect to velocity-dependent forces.

3.1.4 Fourth-Order Runge-Kutta (RK4)

Although the RK4 algorithm continues to enjoy wide-spread use in numerical integration, it is non-symplectic, i.e. non-energy conserving. In many applications this does not present a problem, as the total propagation time is short enough that accumulated error remains low. However, for purposes including tracking charged particle motion through Earth's dipole-like field, where particles need to be tracked over distances spanning more than 10^4 km, accumulated error can be substantial. RK4 is included here as a user option mainly due to the advantage of very low truncation error of $\mathcal{O}((\Delta t)^5)$, making it useful for testing of active shielding configurations over short distances.

The largest disadvantage of RK4 is the four function calls to calculate the force made at each integration step. VV and AVV require only two. Each of the force function calls is used to estimate the slope between two consecutive points in time, calculated by a weighted sum of coefficients, $KX_{i,j}$, where $X = r$ for position and v for velocity, respectively. Each of the four is indexed by $i = 1..4$ and each of the Cartesian components by $j = 1...3$, forming a set of six coefficients. Note that although there are two first-order ODEs to solve, one for position and one for velocity, there are only four calls to evaluate the force due to the ODEs being coupled. The routine iterates consecutively through the following to propagate

the particle from time t to time $t + \Delta t$:

$$Kr_{1j} = \Delta t\, v_j(t) \tag{64}$$

$$Kv_{1j} = \Delta t\, F(\vec{r}, \vec{v}, t) \tag{65}$$

$$r'_j = r_j(t) + 0.5 Kr_{1j} \tag{66}$$

$$v'_j = v_j(t) + 0.5 Kv_{1j} \tag{67}$$

$$t = t + \frac{\Delta t}{2} \tag{68}$$

$$Kr_{2j} = \Delta t\, v_j(t) + 0.5 Kv_{1j} \tag{69}$$

$$Kv_{2j} = \Delta t\, F(\vec{r}', \vec{v}', t) \tag{70}$$

$$r'_j = r_j(t) + 0.5 Kr_{2j} \tag{71}$$

$$v'_j = v_j(t) + 0.5 Kv_{2j} \tag{72}$$

$$Kr_{3j} = \Delta t\, v_j(t) + 0.5 Kv_{2j} \tag{73}$$

$$Kv_{3j} = \Delta t\, F(\vec{r}', \vec{v}', t) \tag{74}$$

$$t = t + \Delta t \tag{75}$$

$$Kr_{4j} = \Delta t\, v_j(t) + Kv_{3j} \tag{76}$$

$$Kv_{4j} = \Delta t\, F(\vec{r}', \vec{v}', t) \tag{77}$$

$$v_j(t + \Delta t) = v_j(t) + \frac{1}{6}(Kv_{1j} + 2Kv_{2j} + 2Kv_{3j} + Kv_{4,j}) \tag{78}$$

$$r_j(t + \Delta t) = r_j(t) + \frac{1}{6}(Kx_{1j} + 2Kx_{2j} + 2Kx_{3j} + Kx_{4,j}). \tag{79}$$

Note that all of the Kv terms contain $\vec{r}'$ and $\vec{v}'$, all three vector components of the spatial and velocity components in the calculation of the force, F. This is due to the fact that the force term is coupled across the three components in velocity space, i.e. the cross product $\vec{v} \times \vec{B}$.

3.1.5 Boris-Buneman

The Boris-Buneman (BB) integration scheme dates back to the 1970s. It was initially introduced by Boris in the form of the CYLRAD plasma simulation code that integrates the equations of motion of a fully relativistic system with retarded, self-consistent electric and magnetic fields [17]. It has become the standard integration method for a wide variety of plasma physics simulations.

The BB scheme is broken down into three serial steps: (1) propagation by a half step through the external electric field; (2) propagation by a full step through the external magnetic field, and; (3) propagation by a second half step through the electric field. By breaking the electric field 'push' steps into two sections, one before the magnetic field step and one after, irreversibility is maintained over the entire integration step. Steps 1 and 3 utilize and 'Euler-like' approach and will not be discussed further. Step (2) however is more complex. Boris determined that incorporation of the velocity-dependent force due to the magnetic field could be done by rotating to a new frame of reference.

The idea is relatively simple but highly effective. Under the assumption that $|\vec{E} X \vec{B}| < |\vec{B}|^2$ there exists a frame of reference where the electric field is zero. The integration step thus proceeds by first propagating a half step with $\vec{B} = 0$, then rotating to a frame of reference where $\vec{E} = 0$ and propagating a full step using only the magnetic field B, and finally, propagating another half step with $\vec{B} = 0$. Although not symplectic, this method is highly stable, is highly accurate and surprisingly conserves the phase space. The only drawback is in cases of high electric field gradients. Here, the step size would likely need to be substantially reduced since the initial and final half-time step push uses a standard Euler method.

The Boris algorithm contained within ASPP proceeds as follows. First, copy positions at time t to new vector $\vec{r}'(t)$ and update to time $t = t + \Delta t/2$ assuming $\vec{B} = 0$

$$\vec{r}\prime(t + \Delta t/2) = \vec{r}(t) + \frac{\Delta t}{2}\vec{v}(t). \tag{80}$$

Define a pseudo-momentum vector as

$$\vec{p}(t) = \gamma(t)\vec{v}(t). \tag{81}$$

We then iterate consecutively through each the following steps,

$$\vec{p}_- = \vec{p}(t) + \left(\frac{Ze\Delta t}{2m}\right)\vec{E}(\vec{r},t) \tag{82}$$

$$\gamma_- = \sqrt{1 + |\vec{p}_-|^2/c^2} \tag{83}$$

$$\vec{t} = \frac{Ze\vec{B}}{m}\frac{\Delta t}{2} \tag{84}$$

$$\vec{s} = \frac{2\vec{t}}{1 + |\vec{t}|^2} \tag{85}$$

$$\vec{p}' = \vec{p}_- + \vec{p}_- \times \vec{t} \tag{86}$$

$$\vec{p}_+ = \vec{p}_- + \vec{p}' \times \vec{s} \tag{87}$$

$$\vec{p}(t + \Delta t) = \vec{p}_+ + \left(\frac{Ze\Delta t}{2m}\right)\vec{E}(\vec{r},t) \tag{88}$$

$$\vec{v}(t + \Delta t) = \frac{\vec{p}(t + \Delta t)}{\gamma(t + \Delta t)} \tag{89}$$

$$\vec{r}(t + \Delta t) = \Delta t\vec{v}(t + \Delta t). \tag{90}$$

The last two equations define the final position and velocity vectors at the end of the step Δt. Note that relativistic effects are taken into account by updating the value of γ. Although there is not an increase in function calls over the VV method, the calculation of matrix multiplication does come at a cost. As a result, the Boris method is not as computationally efficient as the VV, but is more accurate when including magnetic field components.

3.2 Particle Initialization

ASPP contains two different particle initialization methods aimed at investigating active shielding configuration shielding efficacy in the presence of the in-space particle field and ground-based measurements using particle beams. For in-space modeling, it is crucial that particles are initialized according to the appropriate initial velocity distribution and initial charge state. For particle beam simulations, the initialization routine assumes a mono-energetic distribution of particles with a single, non-zero velocity vector component aligned with the beam axis.

3.2.1 Directionally-Isotropic Field

The ion population in free space is largely isotropic in direction. Thus, to perform an assessment of deflection characteristics, ion velocities must be distributed accordingly. It should be noted that interaction of ions with the Interplanetary Magnetic Field (IMF) are neglected in this analysis since the IMF magnitude is on the order of nT, much too weak to cause substantial particle deflection over length scales comparable to shield field thicknesses. Assuming the absence of a CME shock front, the energetic ion distribution is, in general, uniform in regions of the heliosphere where crewed-missions will be conducted. Cartesian coordinates are used throughout the following analysis.

The method implemented here to ensure any field configuration is illuminated with a directionally-isotropic particle field was presented by Rajon and Bolch [18]. For each run the incident particle distribution is assumed monoenergetic with particle speeds, v_0 defined as

$$v_0 = c \left(1 - \gamma^{-2}\right)^{1/2}, \tag{91}$$
$$\gamma = 1 + E_k/E_0. \tag{92}$$

Taking the entire 4π steradians to distribute ion velocities would be highly inefficient as half of the ions would have trajectories directed away from the field region. Instead, this method ensures that all particles initialized have at least one component of the velocity directed towards the field source.

The algorithm is as follows. First, we calculate the angles of the ith particle associated with a vector position from the origin as

$$\theta_i = \cos^{-1}(1 - 2R_1) \tag{93}$$
$$\phi_i = 2\pi R_2, \tag{94}$$

where R_1 and R_2 are uniformly distributed random numbers on $[0, 1]$. We then calculate a random vector in the plane tangent to the surface of the launch sphere from

$$\rho_{p,i} = R_L \sqrt{R_3} \tag{95}$$
$$\phi_{p,i} = 2\pi R_4 \tag{96}$$

where R_3 and R_4 are also uniformly distributed random numbers on $[0, 1]$, and R_L is denoted as the launch radius. The starting positions of ith particle is then calculated from the position on the tangent plane defined by solid angle direction and disk sampling, as,

$$x_i = R_L \sin(\theta_i) \cos(\phi_i) + \rho_{p,i}(\sin(\phi_i) \cos(\phi_{p,i}) \tag{97}$$
$$+ \cos(\theta_i) \cos(\phi_i) \sin(\phi_{p,i})),$$
$$y_i = R_L \sin(\theta_i) \sin(\phi_i) - \rho_{p,i}(\cos(\phi_i) \cos(\phi_{p,i}) \tag{98}$$
$$- \cos(\theta_i) \sin(\phi_i) \sin(\phi_{p,i})),$$
$$z_i = R_L \cos(\theta_i) - \rho_{p,i} \sin(\theta_i) \sin(\phi_{p,i}), \tag{99}$$

where $\rho_{p,i}$ and $\phi_{p,i}$ is the polar distance and angle, respectively, within the tangent plane. The cartesian components of the initial particle velocity are then calculated as

$$v_{x,i} = -v_0 \sin(\theta_i) \cos(\phi_i) \tag{100}$$
$$v_{y,i} = -v_0 \sin(\theta_i) \sin(\phi_i) \tag{101}$$
$$v_{z,i} = -v_0 \cos(\theta_i). \tag{102}$$

It is important to note that the the initial particle positions are not confined to the surface of a sphere of radius R_L. This is due to the nature of generating locations uniformly over the surface of the tangent plane. Figure 2 shows the distribution of positions as the square of the position vector $r^2 = |\vec{r}|^2$. Note that it is distributed uniformly between R_L^2 and $2R_L^2$. One could make the distribution narrower by making the size of the tangent plane disk much smaller than the radius of the launch sphere, R_L. However, this would result in particle velocities being radially directed, destroying the isotropic nature of the incident field. In general, the tangent plane disk radius should be on the order of R_L.

An example distribution of initial velocity components is shown in Figure 3. It is plotted as a ratio of v_i/v_0 where v_0 is the maximum speed corresponding to the kinetic energy selected at run time. Each velocity component is clearly distributed uniformly over $[-v_0, v_0]$, as expected for an directionally-isotropic particle field.

For 3d simulations of active shielding configurations, one of the key metrics when performing trajectory calculations is the degree of flux reduction. This is simulated by finding the fraction of particles that cross the surface of a spherical volume of radius R_{cr} to be protected. Even with zero electric and/or magnetic fields, the number of particles crossing this surface will always be less than the number initialized except in the limit $R_{cr} \to R_L$. A series of simulations were conducted with zero external fields to verify that the flux density at the protected volume surface is the same as flux density initialized. This was accomplished by counting particles crossing the surface, which is expected to scale as R_{cr}^2/R_L^2 if the incident particle field is truly modeled in an isotropic fashion. Results are shown in Figure 4.

24
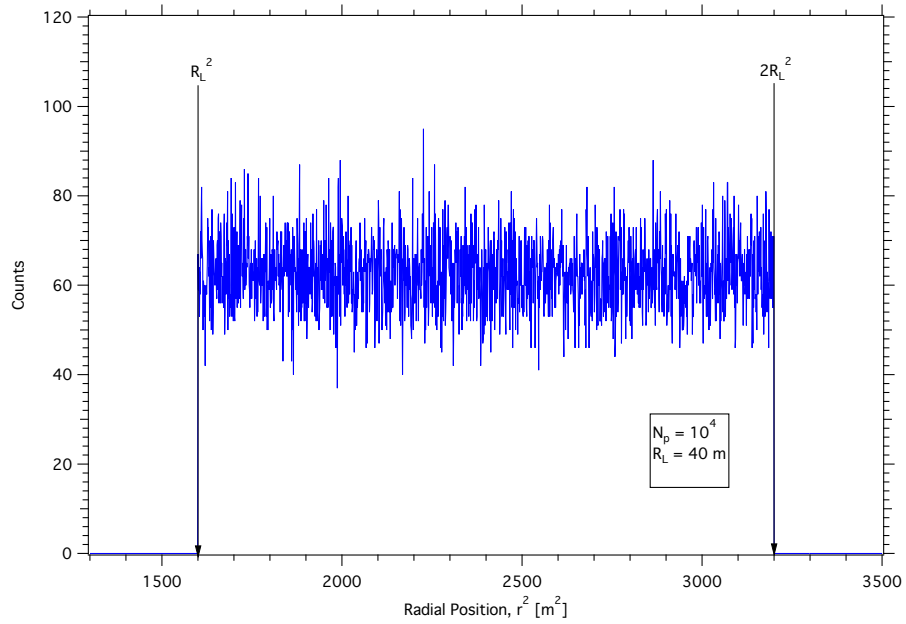
Figure 2. Distribution of initial particle radial position plotted as r^2 for a modeled directionally-isotropic particle field according to Eqs 97- 99.
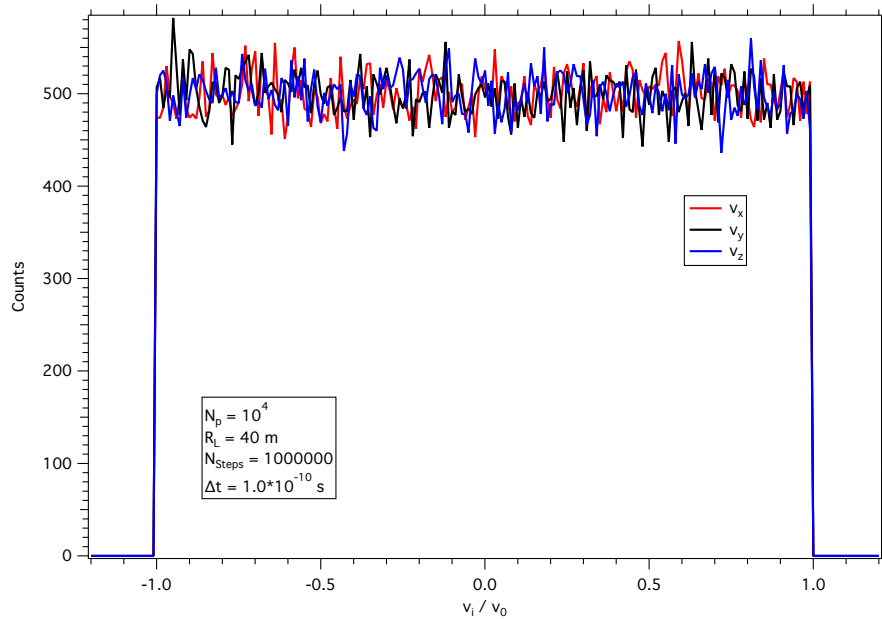
Figure 3. Distribution of initial particle velocity components for a modeled directionally-isotropic particle field according to Eqs. 100-102.
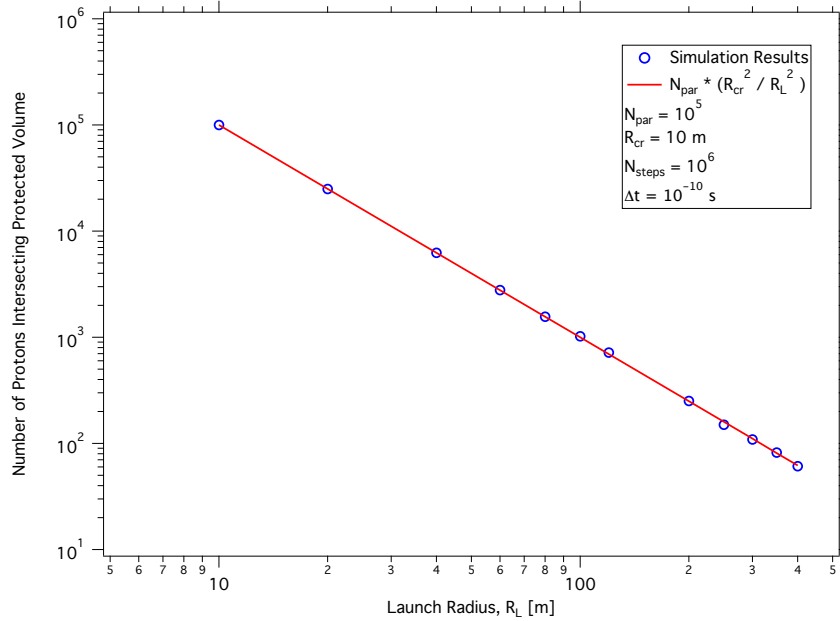
Figure 4. Number of particles crossing inner protected volume surface as a function of the launch radius, R_L. The red solid line is a least-squares regression to the simulation results.

3.2.2 1d Beamline Initialization

ASPP is capable of initializing particles to simulate particle beams. This allows the user to simulate beam line measurements for comparison. Here, particles are assumed to be initialized within a circular disk, of radius R_d, whose normal vector is aligned with the ion beam axis. Initial velocity components are generated as

$$\begin{aligned}
v_{x,i} &= -\delta_x v_0 & (103) \\
v_{x,i} &= -\delta_y v_0 & (104) \\
v_{x,i} &= -\delta_z v_0, & (105)
\end{aligned}$$

where $\delta_{x(yz)} = \pm 1$ is the Kronecker delta. The beam axis is set from within the user input file and this in tern sets one of the delta functions to a value of one and the other two to zero. Positions are then defined in the transverse plane to the beam axis according to

$$\begin{aligned}
\rho_i &= R_d R_1 & (106) \\
\phi_i &= 2\pi R_2 & (107) \\
w_i &= \rho_i cos(\phi_i) & (108) \\
t_i &= \rho_i sin(\phi_i), & (109)
\end{aligned}$$

where w_i and t_i are the two Cartesian coordinates in the (x, y, z) triplet defining the plane transverse to the beam axis defined by the user at run

26

time, and R_1 and R_2 are uniformly distributed random numbers. For instance, if the beam axis is aligned with the x-axis then $(w_i, t_i) = (y_i, z_i)$. Example output using this algorithm for the initial particle positions is shown in Figure 5. Note the visual uniformity in points, as expected.
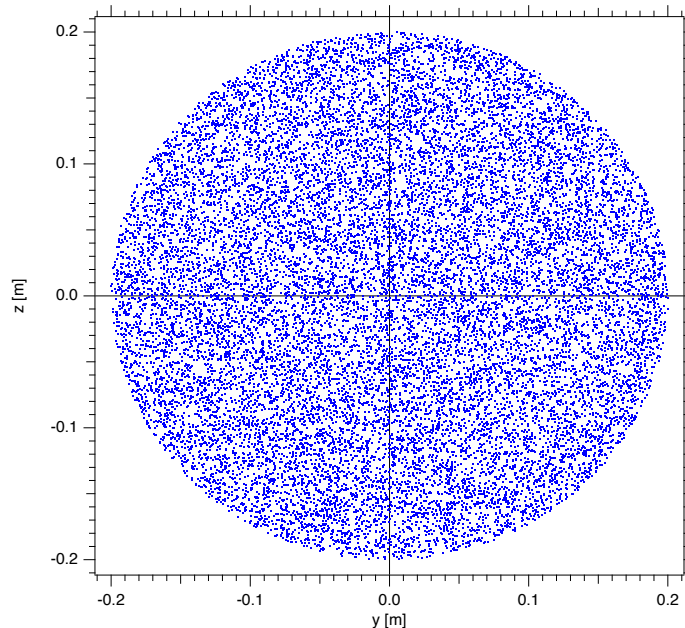
Figure 5. Initial particle positions for a beam of $N_p = 20000$ protons in the (y, z) plane with velocity directed along the x-axis, generated using Eqs. 106-109. Note the isotropic distribution of positions.

.

3.3 Angular Binning

ASPP incorporates an angular binning method that determines the angular fluence (or flux density) distribution over a user-defined sphere of radius R centered about the origin. The utility of this feature is that differences in particle penetration through the shield can be probed to find locations where the electric and/or magnetic fields need to be tailored to increase shielding efficacy. Figure 6 displays an example of the output generated by the angular binning method.

The method used to determine the angular fluence distribution is described below. The particle fluence at some arbitrary position in a radiation field, Φ, is commonly defined as the number of particles per unit area crossing a surface. This definition of the particle fluence is readily understood, but a more useful definition of the fluence can be expressed in terms of particles which traverse through a volume. Consider a volume ΔV that is penetrated by I particles. The total path length traversed

27
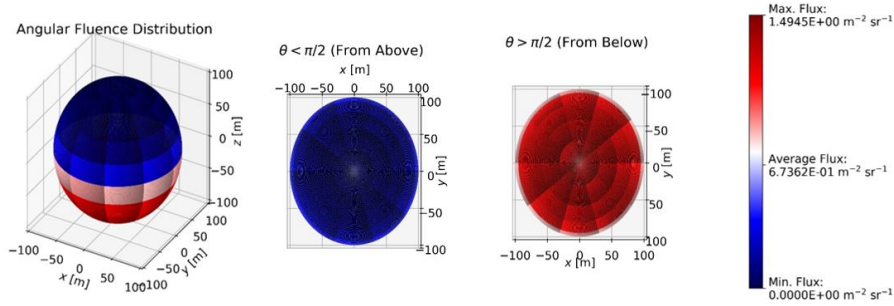
Figure 6. Example angular fluence distribution. For this example, 100 point charges of charge magnitude 1.111×10^{-3} C were approximately evenly distributed over the upper hemisphere ($0 < \theta < \pi/2$) of a sphere of radius $R = 100$ m. The sphere was isotropically irradiated with 100 MeV protons. This image shows that most particles are entering the angular binning sphere from the region which is devoid of shielding ($\pi/2 < \theta < \pi$).

within volume ΔV by the ith particle is expressed by s_i. The particle fluence at a point can be expressed as [64]

$$\Phi = \lim_{\Delta V \to 0} \left[\frac{\sum_i s_i}{\Delta V} \right] \simeq \frac{\sum_i s_i}{\Delta V}. \tag{110}$$

The average fluence over a non-infinitesimal volume, used in ASPP and described below, is found as

$$\Phi_{avg} = \frac{\sum_i s_i}{\Delta V}. \tag{111}$$

ASPP's particle propagation methods currently follow a fixed time-step approach, where the number of propagation steps and the time-step duration, Δt, are specified by the user before the beginning of each run. During each propagation step, each particle's position vector is updated. Express the position vector of particle i during the previous propagation step as $\vec{r}_i(t)$ and call the position vector of particle i for the current propagation step $\vec{r}_i(t + \Delta t)$. The total path length traversed by particle i through the angular binning sphere of radius R can be found via the following method.

1. If $|\vec{r}_i(t)| > R$ and $|\vec{r}_i(t + \Delta t)| \leq R$, particle i enters the angular binning sphere during this propagation step. Determine the entry point of the sphere (via sphere intersection method), $\vec{r}_{i,entry}$. Determine the path length traversed over this step as $\Delta s_i(\Delta t) = |\vec{r}_i(t + \Delta t) - \vec{r}_{i,entry}|$. Note that the solid angle direction in which particle i enters the angular binning sphere, $\vec{\Omega}$, is determined via $\vec{\Omega} = (\vec{r}_{i,entry} - \vec{r}_i(t + \Delta t))/|\vec{r}_{i,entry} - \vec{r}_i(t + \Delta t)|$.

28

2. If $|\vec{r}_i(t)| \leq R$ and $|\vec{r}_i(t + \Delta t)| \leq R$, particle i has been within the angular binning sphere throughout this propagation step. The path length traversed over this step is simply $\Delta s_i(\Delta t) = |\vec{r}_i(t + \Delta t) - \vec{r}_i(t)|$.

3. If $|\vec{r}_i(t)| < R$ and $|\vec{r}_i(t+\Delta t)| \geq R$, particle i exits the sphere during this propagation step. Determine the exit point of the sphere (via sphere intersection method), $\vec{r}_{i,exit}$. Determine the path length traversed over this step as $\Delta s_i(\Delta t) = |\vec{r}_i(t) - \vec{r}_{i,exit}|$.

4. Otherwise, $\Delta s_i(\Delta t) = 0$.

5. Determine the total path length traversed by particle i through the angular binning sphere of radius R via $s_i = \sum_{\Delta t} \Delta s_i(\Delta t)$.

The previous method references a sphere intersection method. Given a sphere of radius R and two points, $\vec{r}_i(t)$ and $\vec{r}_i(t+\Delta t)$, whose associated line segment intersects the surface of the sphere, the point at which the connecting line segment intersects the sphere can be determined. This method is described below.

1. Let $\vec{r}_i(t) = x_0 \hat{i} + y_0 \hat{j} + z_0 \hat{k}$ and let $\vec{r}_i(t + \Delta t) = x_1 \hat{i} + y_1 \hat{j} + z_1 \hat{k}$.

2. The parametric equation for the intersection of a line which passes through (x_0, y_0, z_0) and (x_1, y_1, z_1) and a sphere of radius R is given by:

$$A\lambda^2 + B\lambda + C = 0 \qquad (112)$$
$$A = (x_0 - x_1)^2 + (y_0 - y_1)^2 + (z_0 - z_1)^2,$$
$$B = 2\left(x_0(x_1 - x_0) + y_0(y_0 - y_1) + z_0(z_1 - z_0)\right),$$
$$C = x_0^2 + y_0^2 + z_0^2 + R^2, \text{ and}$$
$$\lambda \in (-\infty, \infty).$$

Solving for λ determines the intersection points.

3. Let the discriminant of Eq. 112 be $\Delta = B^2 - 4AC$. If $\Delta \leq 0$, particle i does not intersect with the sphere. Only the case where $\Delta > 0$ is considered.

4. Solve for both values of λ via the quadratic formula, $\lambda^+ = (-B + \sqrt{\Delta})/(2A)$ and $\lambda^- = (-B - \sqrt{\Delta})/(2A)$.

5. If $\lambda^+ \leq \lambda^-$, then the entry point is associated with λ^+. Thus, let $\lambda_{entry} = \lambda^+$ and $\lambda_{exit} = \lambda^-$. Otherwise, $\lambda_{entry} = \lambda^-$ and $\lambda_{exit} = \lambda^+$.

6. The entry and exit positions can be determined via $\vec{r}_{i,entry} = (1 - \lambda_{entry})\vec{r}_i(t) + \lambda_{entry}\vec{r}_i(t+\Delta t)$ and $\vec{r}_{i,exit} = (1-\lambda_{exit})\vec{r}_i(t) + \lambda_{exit}\vec{r}_i(t+\Delta t)$.

After determining the total path length traversed by each particle through the angular binning sphere, all particles which enter the sphere in the direction $\vec{\Omega}$ about $\Delta\Omega$ are grouped into angular bins. Angular bins are defined by the user, who selects the number of bins in the polar direction θ and azimuthal direction ψ. Specifically, angular bin increments $\Delta\theta$ and $\Delta\psi$ are defined by Eqs. 113 and 114. N_θ and N_ψ are the number of angular bin divisions in the polar and azimuthal directions respectively.

$$\Delta\theta = \frac{\pi}{N_\theta} \tag{113}$$

$$\Delta\psi = \frac{2\pi}{N_\psi} \tag{114}$$

The angular fluence in a single angular bin is given by Eq. 115.

$$\Phi(\vec{\Omega}) = \frac{\Phi}{\Delta\Omega}, \tag{115}$$

where $\Delta\Omega$ is given by

$$\Delta\Omega = (\cos\theta_{low} - \cos\theta_{high})(\psi_{high} - \psi_{low}) \tag{116}$$

The subscripts "high" and "low" simply refer to the upper and lower boundaries of the angular bin, i.e., $\theta_{high} = \theta_{low} + \Delta\theta$. Once each angular bin is assigned an angular fluence value, a plot such as the one in Figure 6 is generated and the angular binning method is complete.

3.4 Sphere Shadowing

Charged conducting spheres are used to generate an electric dipole field in the experimental case. When these spheres are irradiated by a beam of charged particles, a void results. The electric field generated by charging up the conducting spheres is well approximated in ASPP using the electric field due to point charges in place of actual conducting spheres. However, when the physical conducting spheres are irradiated when no voltage is applied to either sphere, a void area (typically two, depending on orientation) still appears in the image. This phenomenon is due to the presence of the physical spheres. The spheres shield charged particles whose trajectories intersect with the sphere (or the rod connecting them to the voltage supply). The team refers to this behavior as "sphere shadowing", as the shadow of each sphere is usually visible in the image data. See Figure 7 for an example of this behavior.

Using a point charge approximation to simulate the electric field produced by these charged spheres is adequate for reproducing the electric field in ASPP, but does not capture the sphere shadowing behavior exhibited during measurement. For this reason, the team added a "sphere shadowing" mode to ASPP. The purpose of this mode is to identify particles whose trajectories intersect one of the conducting spheres at any
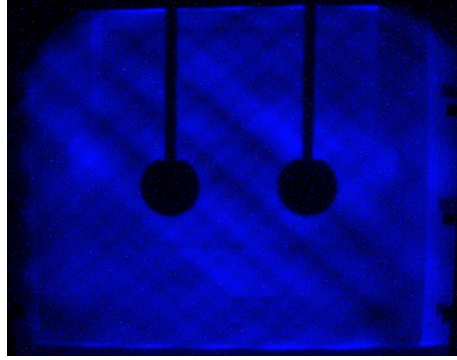
Figure 7. Sphere shadowing behavior for 0 kV spheres of 1 inch diameter. Beam consisted of fully-stripped 6 MeV He ions.

point during particle propagation. The user is able to specify a shadow radius R_{shadow} which represents the conducting sphere radius. If particle i passes within R_{shadow} of any point charge, particle i is flagged and propagation continues as normal. In other words, if $|\vec{r}_i(t) - \vec{r}_p| < R_{shadow}$, particle i is flagged. $\vec{r}_i(t)$ is the position of particle i at time t, and $\vec{r}_p$ is the position of the point charge. After the simulation is complete, the user is able to discriminate particles which intersected the hypothetical charged spheres from those which did not. Figure 8 shows the differences that can be uncovered by implementing sphere shadowing into ASPP.
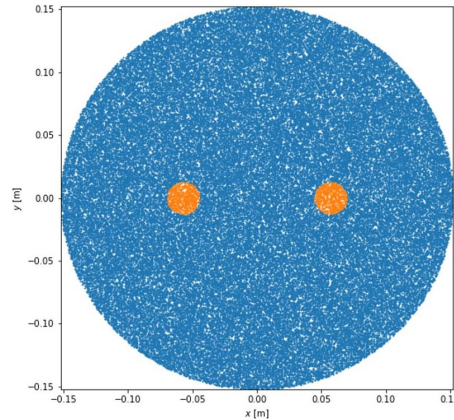
Figure 8. Example of sphere shadowing in ASPP. Orange points represent particles which intersected at least one of the spheres during propagation while blue points represent particles which did not. The charge magnitude of each point charge is zero for this example.

In general, the significance of the shadow decreases as the charge magnitude on each sphere increases. Shadowed particles become more concentrated toward the edge of the void region until the area covered by shadowed particles is no longer significant compared to the area of

the void region. An example of this behavior is shown in Figure 9.
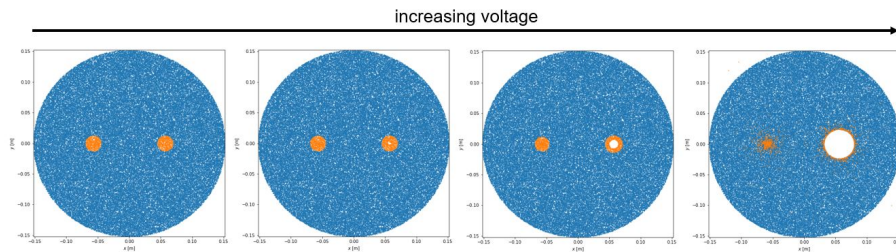
Figure 9. Example of sphere shadowing effects diminishing as the magnitude of the voltage on each sphere increases. Blue points represent particles with trajectories that did not intersect with either sphere; orange points represent particles that did intersect.

3.5 User-Defined Charge States for Projectile Ions

Particle charge state is a user-defined quantity. It is well-known that GCR ions are essentially fully stripped of electrons. However, to accurately simulate beamline measurements, non-fully stripped ions can be represented by making the charge state a user-defined quantity in the input file. The user is able to specify the projectile ion's mass number A, atomic number Z, and number of electrons N_e per ion. The charge of the ion, q_{ion}, is calculated according to

$$q_{ion} = (Z - N_e)\, e, \tag{117}$$

$$m_{ion} = Zm_p + (A - Z)\, m_n, \tag{118}$$

where m_{ion}, is the mass of the ion in kg, e is the electron charge in C, and m_p and m_n are the mass of the proton and neutron in kg, respectively.

4 Platform Dependencies

Source code has been written to be executed in a variety of ways while keeping portability at a maximum. ASPP is written in C++, implementing C++ classes and data structures for easy code readability and adding functions. ASPP integrates Message Passing Interface (MPI) for cross node communication, OpenMP for multi-threading capability, and CUDA for Graphic Processor Units (GPU). The inclusion of OpenMP and MPI allows for simulations to be performed on systems ranging from a single-core processor (laptop and desktop) to multi-core multi-node server systems. CUDA gives the capability to expoit the power of GPU-based systems. CMake is used to configure and build the source code for various platforms and operating languages. ASPP has been tested on Linux, Mac OS-X, and Windows environments successfully.

At code runtime, the main class reads command-line arguments and initializes the run manager. The run manager class initiates the simulation by reading input files, initializing particles, and initializing particle propagation classes. The initialization class creates arrays for all the particles, initiates the particle position based on user-method selected, initiates the particle velocity, and then initiates class objects that handle electromagnetic field calculations selected by the user. The run manager then starts the propagation of particles by calling either GPU or CPU propagation kernels. After all the particles have been propagated, the data is analyzed and output as a comma-separated value (CSV) file.

Each particle occupies a single thread. The ensemble of particles is managed within ASPP by spreading the collection of particles across multiple threads for CPU and GPU propagation. Data sharing between nodes is performed with MPI. OpenMP is used to create multiple threads per node.

4.1 GPU Capability

Compute Unified Device Architecture (CUDA) is a proprietary library by NVIDIA for parallel computing on enabled NVIDIA GPUs for general processing. ASPP implements CUDA to propagate particles at the same time quickly. Multiple GPU cards per node can be supported if OpenMP is enabled with the ability to run on multiple nodes with MPI.

At runtime, ASPP creates the particles, then creates one OpenMP thread per GPU card. Each thread transfers a subset of the particles to the GPU memory and initiates the calculation. After all the threads have completed, the threads then transfer back the particles from the GPU memory to the CPU, where results are combined across nodes and written out to files according to user input. CUDA assigns one particle to each thread on the GPU with 512 threads per block and 64 blocks per multiprocessor; however, these values can be changed to increase performance depending on the GPU architecture. Each thread propagates a single particle through each time step. Since all the threads are propagating particles through the same amount of time steps, the thread block is loaded, increasing performance as all threads are completely busy executing code during the run. After the block has completed analyzing the trajectory of all the particles within the block, the block then analyzes another block of particles. This simple parallelism allows for fast computation of particle trajectories.

4.2 OpenMP

OpenMP implements multi-threading capability through compiler directives and libraries that are easily integrated into C, C++, and Fortran compilers. OpenMP is available on GCC, Intel, and Clang compilers. If configured using CMake, OpenMP is used to make multiple threads

33

for particle creation and propagation. At runtime, OpenMP automatically determines the number of threads available per node, which can be changed by setting environment variables for OpenMP. Upon particle initialization, the particles are split between the number of threads available in the thread pool with each thread creating a subset of particles. At propagation time, each thread individually propagates a single particle for every time step and then proceeds to another particle.

4.3 Message Passing Interface (MPI)

MPI is a specification established to allow for the point-to-point communication between different processes, which may reside on the same processors or across computational nodes in a cluster. Several implementations of MPI have been created, including the OpenMPI and Intel libraries. ASPP has been written based on OpenMPI; however, testing of ASPP shows better performance on processors with the Intel library.

ASPP uses MPI to communicate between nodes, allowing ASPP to quickly and efficiently expand to multiple nodes. If configured with MPI using CMAKE when building the code, the application is built natively with MPI capability. During run initialization, a master MPI thread creates threads across nodes, or the same node, and splits the number of particles between MPI threads. Each node then initializes a set of particles based on a reproducible different random number seed per node, which reduces the amount of data split between nodes during initialization and speeds particle creation. After particles are generated in each node, ASPP then initiates the propagation of particles across all the nodes. Each node completes designated propagation tasks and then waits for all other node propagation to complete. The master node then collects the resulting data for each particle to be written out as a single file.

5 Field Configuration Toolset

To increase flexibility of use a collection of analytic field configurations have been included. Three of these are taken directly from past investigations, two are additions utilizing ideal field constructs, and one is for testing purposes for visualization in 2d. The fields included are:

1. Constant z-aligned magnetic field confined to a ring in the (x,y) plane. Ring offset from the origin, thickness and magnetic field magnitude set by the user. Configuration centered at $(0, 0, 0)$ in Cartesian coordinates.

2. Constant 3d azimuthal magnetic field confined to a spherical shell. Shell offset from the origin, thickness and magnetic field magnitude defined by the user. Configuration centered at $(0, 0, 0)$ in Cartesian coordinates.

3. Ideal magnetic dipole field. Magnetic moment specified by the user. Configuration centered at $(0, 0, 0)$ in Cartesian coordinates.

4. Ideal current loop. Magnetic moment and loop radius specified by the user.

5. 3d Helmholtz coil configuration. Magnetic moment and loop radius specified by the user. Configuration centered at $(0, 0, 0)$ in Cartesian coordinates.

6. 3d Maxwell coil configuration. Magnetic moment and loop radius specified by the user. Configuration centered at $(0, 0, 0)$ in Cartesian coordinates.

7. 3d distribution of ideal magnetic dipoles. Number, position vector and dipole moment and moment orientation vector specified by the user through an input file.

8. 3d distribution of ideal electric dipoles. Number, position and dipole moment specified by the user through an input file.

9. 3d distribution of point charges. Number, position vector and charge specified by the user through an input file.

10. Array of Line Charges. Number, linear charge density and line endpoints specified by the user through an input file.

11. Array of Rectangular Charge Sheets. Number, total surface charge (assumed homogenous distribution), position vector of the center of planar sheet and side length specified by the user through an input file.

Details of each configuration are discussed below. In addition to this set of default configurations, the code includes the ability to read in any magnetic and electric field values on a 3d Cartesian rectangular grid. This capability allows for inclusion of magnetic fields that have been modeled with external static field modeling software.

5.1 Constant z-aligned Magnetic Field

Relativistic motion of a charged particle in a constant magnetic field is one of the few cases where the equations of motion are analytic and the resulting trajectories is well-known. Thus, it serves as a benchmark to using the various integration methods included based on user-defined initial conditions. The field is defined as

$$\vec{B} = B_0 \hat{z}, \tag{119}$$

where B_0 is the field magnitude. The value of B_0 is defined by the user.

5.2 Constant 3d Azimuthal Field

Assuming the magnetic field is not infinite at any point in $3d$ space, the configuration with the greatest deflection power is that of a constant uniform field. This field will serve as a reference point for all analysis presented here. For simplicity, and the greatest spatial coverage, a spherical shell of thickness δ_r and azimuthal field orientation will be used.

An azimuthal field in spherical coordinates is given by,

$$\vec{B} = B_0\hat{\phi} = B_0(-\sin(\phi)\hat{x} + \cos(\phi)\hat{y}), \tag{120}$$

where the angular component of the Cartesian unit vectors can be written as,

$$\cos(\phi) = \frac{x}{\rho}, \tag{121}$$

$$\sin(\phi) = \frac{y}{\rho}, \tag{122}$$

$$\rho^2 = x^2 + y^2. \tag{123}$$

5.3 Ideal Magnetic Dipole Field

There are several reasons to include the field of an ideal dipole. Earth's geomagnetic field is highly effective at shielding energetic particles in LEO. Some of the first investigations aimed at probing the origin of the trapped belts, i.e. Van Allen belts, were carried out by Störmer assuming the geomagnetic field could be represented by that of an ideal magnetic dipole. Ideal dipole fields can also be easily combined in various configurations to produce relatively complex field geometries and can be evaluated numerically with relatively low computational overhead. The vector field $\vec{B}$ of an ideal point dipole with magnetic vector dipole moment $\vec{M}$ is defined as

$$\vec{B}(\vec{r}) = \frac{\mu_0}{4\pi r^3}\left[3(\vec{M}\cdot\hat{r})\hat{r} - \vec{M}\right] - \frac{2}{3}\mu_0\vec{M}\delta^3(\vec{r}), \tag{124}$$

where the last term represents the delta-function moment of a point dipole at the origin. Due to azimuthal symmetry and no *a priori* preferred field orientation with respect to a spacecraft, dipole orientation can be fixed along a single axis. Aligning with the z-axis, $\vec{M} = M_0\hat{z}$. The magnetic field vector can then be written as

$$\vec{B}(\vec{r}) = \frac{\mu_0 M_0}{4\pi r^3}\left[3(\hat{M}\cdot\hat{r})\hat{r} - \hat{M}\right] - \frac{2}{3}\mu_0 M_0\hat{M}\delta^3(\vec{r}). \tag{125}$$

The simulation code integrates the equations of motion in cartesian coordinates. The vector field can be converted to Cartesian coordinates by expressing the radial unit vector as

$$\hat{r} = \sin(\theta)\cos(\phi)\hat{x} + \sin(\theta)\sin(\phi)\hat{y} + \cos(\theta)\hat{z}. \tag{126}$$

Using the Cartesian to spherical coordinate transformation, the Cartesian field components of an ideal point dipole can be expressed as,

$$
\begin{aligned}
B_x(x, y, z) &= \frac{3\mu_0 M_0}{4\pi} \frac{xz}{r^5} \\
B_y(x, y, z) &= \frac{3\mu_0 M_0}{4\pi} \frac{yz}{r^5} \\
B_z(x, y, z) &= \frac{\mu_0 M_0}{4\pi} \frac{(3z^2 - r^2)}{r^5}
\end{aligned}
\tag{127}
$$

These field components can be generalized for any given collection of ideal point dipoles. Consider a point dipole located at position $\vec{r_i}$. Given a particle traveling with velocity vector $\vec{v}$ at time t is located at position $\vec{r}$, the new field position vector is $\vec{r}' = \vec{r} - \vec{r_i}$. Thus the total (B_x, B_y, B_z) field is a sum over a N_d dipoles, i.e.

$$
B_x(x, y, z) = \frac{3\mu_0}{4\pi} \sum_{i=1}^{N_d} M_{i,0} \frac{x_i'}{r'^5}
\tag{128}
$$

$$
B_y(x, y, z) = \frac{3\mu_0}{4\pi} \sum_{i=1}^{N_d} M_{i,0} \frac{y_i' z_i'}{r'^5}
\tag{129}
$$

$$
B_z(x, y, z) = \frac{3\mu_0}{4\pi} \sum_{i=1}^{N_d} M_{i,0} \frac{(3z'^2 - r'^2)}{r'^5}
\tag{130}
$$

The above expressions hold for the special case of all point dipoles have dipole moment oriented along the z-axis. To generalize further, consider that the ith dipole moment unit vector can be written as $\hat{m}_i = (m_{x,i}, m_{y,i}, m_{z,i})$. The Cartesian field components are then given by

$$
B_x = \frac{\mu_0}{4\pi} \sum_{i=1}^{N_d} \frac{1}{r'^5} \left[(x'^2 - \frac{r'^2}{3}) m_{x,i} + x'y'm_{y,i} + x'z'm_{z,i} \right]
\tag{131}
$$

$$
B_y = \frac{\mu_0}{4\pi} \sum_{i=1}^{N_d} \frac{1}{r'^5} \left[x'y'm_{x,i} + (y'^2 - \frac{r'^2}{3}) m_{y,i} + y'z'm_{z,i} \right]
\tag{132}
$$

$$
B_z = \frac{\mu_0}{4\pi} \sum_{i=1}^{N_d} \frac{1}{r'^5} \left[x'z'm_{x,i} + y'z'm_{y,i} + (y'^2 - \frac{r'^2}{3}) m_{z,i} \right]
\tag{133}
$$

For simulations purposes the code has been developed to read in dipole moment orientation and position vector components from a user-defined file and stores the data as a total of six $1d$ arrays. The arrays are dynamically allocated and thus there is no hard set limit to the number of dipoles that can be included.

5.4 Ideal Current Loop

The ideal current loop field included in ASPP assumes an infinitesimally thin wire of current I (current density $\vec{J}(\vec{x})$). There is no closed form

solution for the field of a current loop with the exception of the axial field. In general, the field can be determined from the magnetic vector potential, $\vec{A}(\vec{x})$, where

$$\vec{A}(\vec{x}) = \frac{\mu_0}{4\pi} \int \frac{\vec{J}(\vec{x})}{|\vec{x} - \vec{x}'|} d^3\vec{x} \tag{134}$$

$$= \frac{\mu_0 I}{4\pi} \int \frac{\delta(r' - a)\delta(z')\hat{\phi}(\phi')}{|\vec{x} - \vec{x}'|} d^3\vec{x}, \tag{135}$$

where

$$\hat{\phi}(\phi') = -sin\phi'\hat{x} + cos\phi'\hat{y}. \tag{136}$$

The vector magnetic field is then given by the curl of $\vec{A}$, i.e. $\vec{B} = \nabla X \vec{A}$. The resulting solution is in the form of elliptic integrals that have to be evaluated numerically. The Cartesian components of the current loop field can be written as

$$B_x(x, y, z) = \frac{Dxz}{\rho^2}\left[(a^2 + r^2)E(k^2) - \alpha^2 K(k^2)\right] \tag{137}$$

$$B_y(x, y, z) = \frac{Dyz}{\rho^2}\left[(a^2 + r^2)E(k^2) - \alpha^2 K(k^2)\right] \tag{138}$$

$$= \frac{y}{x}B_x(x, y, z) \tag{139}$$

$$B_z(x, y, z) = D\left[(a^2 - r^2)E(k^2) + \alpha^2 K(k^2)\right], \tag{140}$$

where

$$D = \frac{C}{2\alpha^2\beta} \tag{141}$$

$$\rho = x^2 + y^2$$

$$r^2 = x^2 + y^2 + z^2$$

$$\alpha^2 = a^2 + r^2 - 2a\rho$$

$$\beta^2 = a^2 + r^2 + 2a\rho$$

$$k^2 = 1 - \alpha^2/\beta^2$$

$$C = aB_0,$$

with B_0 the magnetic field on axis at the center of the coil $(B_z(0,0,0))$ and a the loop radius. The functions $K(k^2)$ and $E(k^2)$ are the complete error functions of the first and second kind respectively, defined as

$$K(k^2) = \int_0^{\pi/2} \frac{d\alpha}{(1 - k^2 \sin^2(\alpha))^{1/2}} \tag{142}$$

$$K(k^2) = \int_0^{\pi/2} (1 - k^2 \sin^2(\alpha))^{1/2} d\alpha \tag{143}$$

$$k^2 = \frac{4ar}{a^2 + r^2 + 2ar\sin(\theta)} \tag{144}$$

$$K(0) = E(0) = \frac{\pi}{2}. \tag{145}$$

Each of the functions are integrated numerically at each time step. The numerical algorithm has been taken from [63] and has been checked by calculating contours in the $y - z$ plane as well as calculating the on-axis field as a function of distance z. The elliptic integrals are approximated as a power-series expansion and is computationally fast. Series coefficients are taken from [63]. One expects the far-field on-axis value to scale as $B_z(0, 0, z) \propto z^{-3}$. Figure 10 show contours and asymptotic axial field dependency. In general the agreement with the predicted asymptotic far-field value of z^{-3} is quite good. Figures 11 and 12 show field
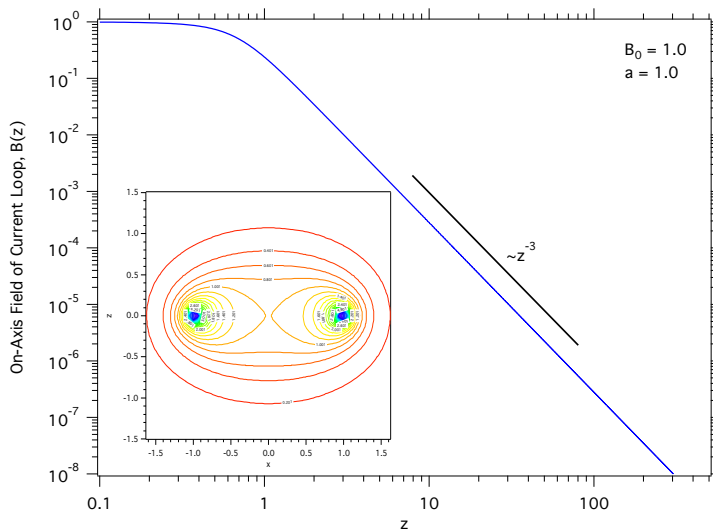
Figure 10. Magnetic field of an ideal current loop (Section 5.4). The inset is a contour plot of the field magnitude in the x-z plane. The figure shows that for $z \gg a$ the on-axis field scales as z^{-3} as predicted.

streamlines in $2d$ and $3d$ respectively. As expected the field lines near the center of the loop in the (x, y) plane are nearly parallel to the loop axis. The field of a current loop has one large disadvantage similar to that of an ideal dipole. The center of the coil has field lines that are parallel to the coil axis. Thus particles with velocity vectors parallel to this axis see no deflection and can easily traverse vehicles placed at the center of the loop.

5.5 3d Helmholtz Coil

The field of an ideal current loop can be used to include other field configurations built from various combinations and orientations of single loops. One such configuration is the Helmholtz coil. The $3d$ Helmholtz configuration is an arrangement of six ideal current loops, where each single-axis oriented pair is spaced a distance apart equal to the coil radius. If the
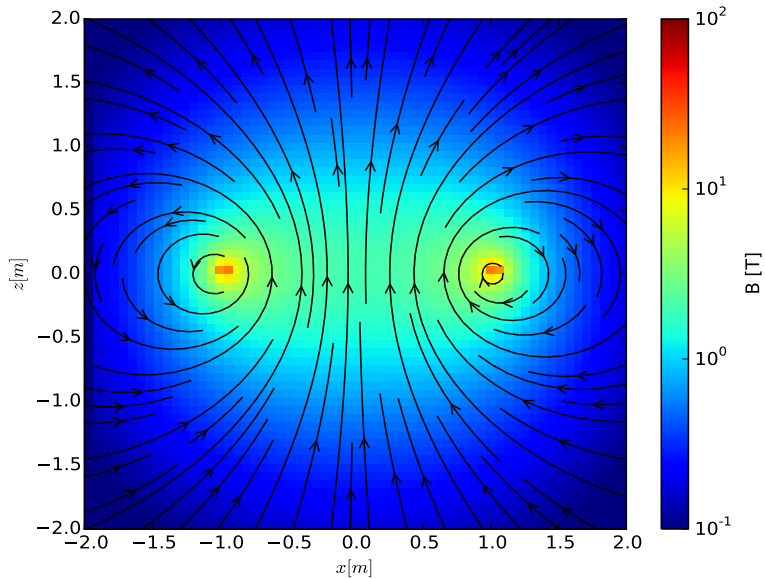
Figure 11. Field streamlines for an ideal current loop (Section 5.4) in the (x,z) plane. The coil circular cross section is located at positions (-1,0) and (1,0).
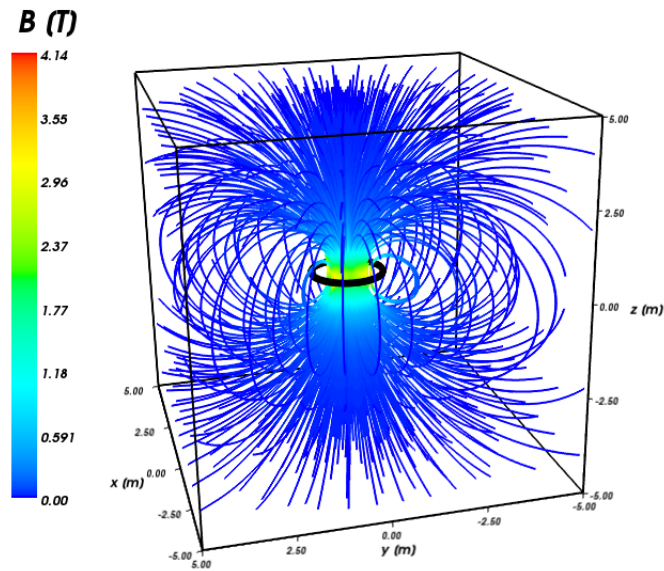
Figure 12. 3*d* field streamlines for an ideal current loop (Section 5.4). The coil, shown as a black solid ring, is located in the (x, y) plane and has a radius of 1 m.

currents in each loop comprising a pair is in the same direction the resulting field at the center point on axis of the coils is non-zero and relatively uniform. Three such pairs, each with its axis aligned along one of the three Cartesian axes are frequently employed in situations where a constant uniform magnetic field is desired.

If the current in each of a pair of loops flows in opposite directions the resulting configuration is called an 'anti-Helmholtz' coil and the resulting midpoint field is zero. This particular configuration would be preferred for active shielding applications where a vehicle is placed at the center of the coils. The total field at any position $\vec{r}$ is then a sum over the contributions from the 6 individual current loops described in the previous section.

Consider first the case of two sets of anti-Helmholtz coils, one aligned along the z-axis and the along the x-axis. Field streamlines in the (x, z) plane are shown in Figure 13. The field magnitude at $(0, 0, 0)$ is $|\vec{B}| = 0$. Notice that there are four lobes with streamlines that resemble the equatorial streamlines of a dipole field. Each of these is created by the currents in neighboring loops flowing in opposite directions analogous to different poles in a common bar magnet or a pair of opposite charges of a simple electric dipole.
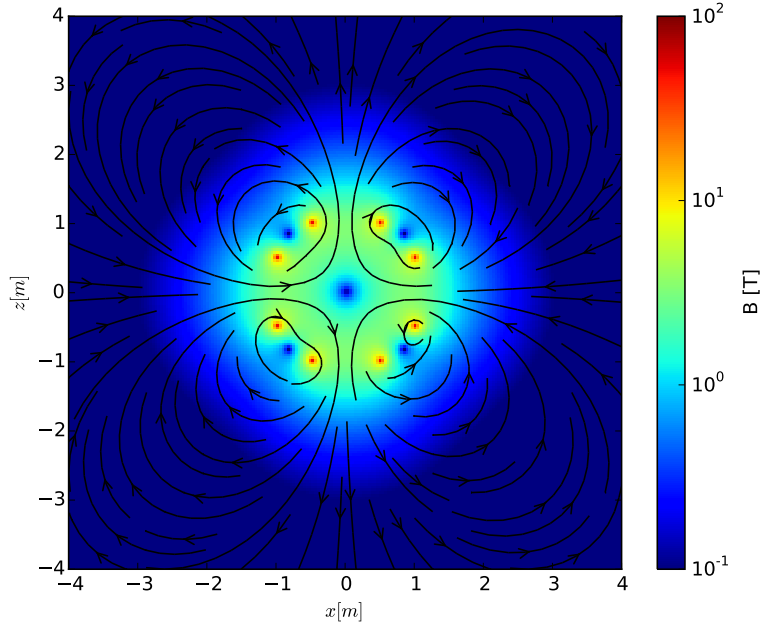
Figure 13.   Axial field for a set of 3-axis Helmholtz coils. For simulation purposes each coil is represented by an ideal current loop (Section 5.4).

41
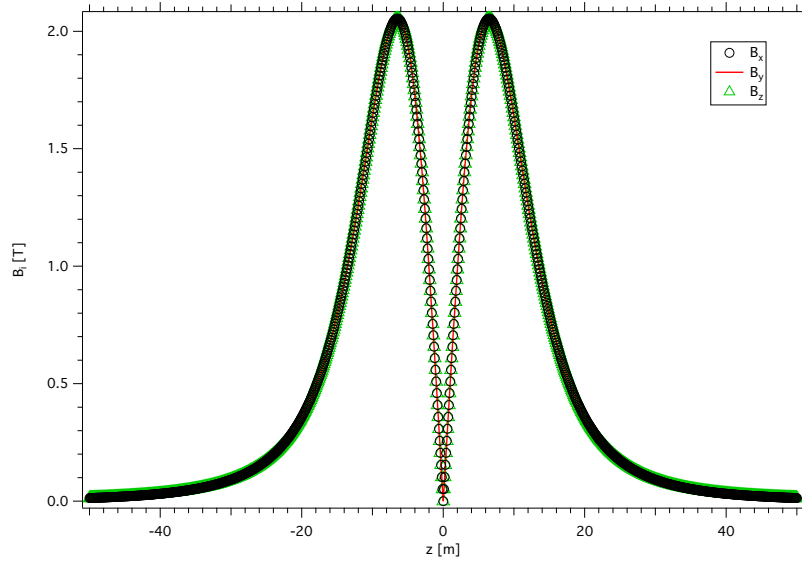
Figure 14.   Axial field for a set of 3-axis Helmholtz coils. For simulation purposes each coil is represented by an ideal current loop (Section 5.4).
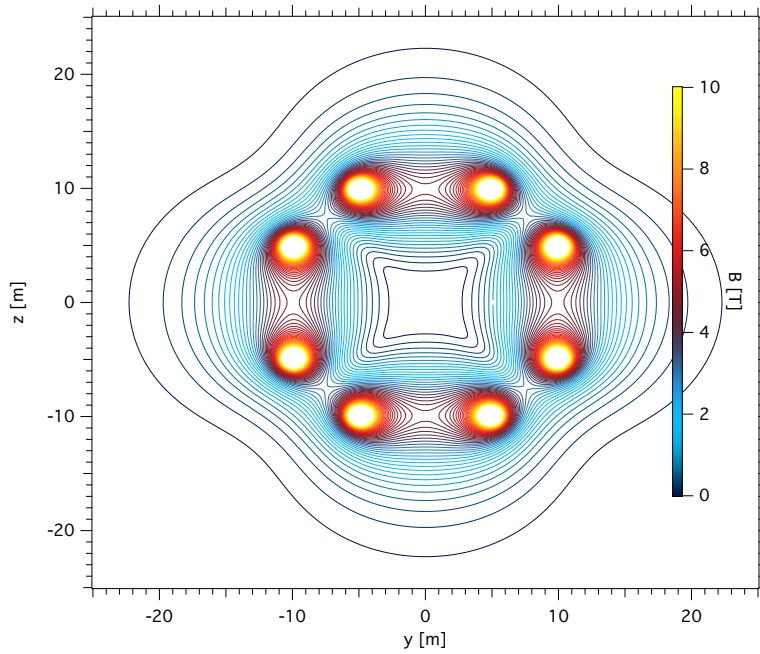
Figure 15.   Field contours for a set of 3-axis Helmholtz coils.   For simulation purposes each coil is represented by an ideal current loop (Section 5.4).
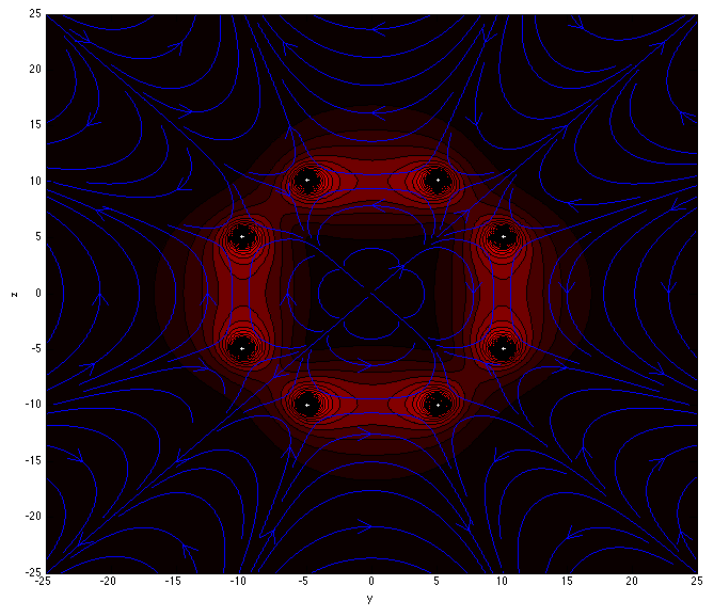
Figure 16. Overlay of field streamlines and contours of field magnitude for a set of 3-axis Helmholtz coils. For simulation purposes each coil is represented by an ideal current loop (Section 5.4).

5.6 3d Maxwell Coil

Maxwell coils are utilized to produce relatively large volumes of nearly constant magnetic field. With respect to spacecraft shielding this coil configuration would have to be placed away from the spacecraft to limit field effects on the vehicle and crew. This configuration also uses the field of an ideal current loop (Section 5.4), with three individual loops configured as shown in Figure 17.
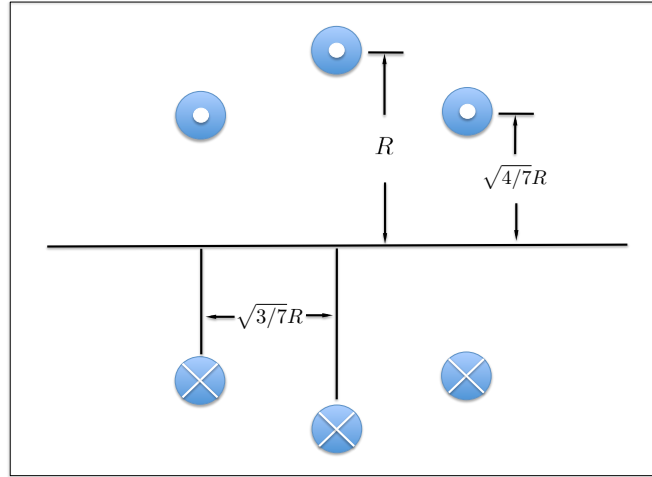
Figure 17.   Maxwell coil layout. For simulation purposes each coil is represented by an ideal current loop of Section 5.4.

5.7 Ideal Electric Point Dipole

The ideal electric dipole is defined by the electric dipole moment vector $\vec{M}_e$. It is functionally the same as the ideal magnetic dipole and is expressed as,

$$\vec{E}(\vec{r}) = \frac{1}{4\pi\epsilon r^3}\left[3(\vec{M}_e \cdot \hat{r})\hat{r} - \vec{M}_e\right] - \frac{2}{3}\mu_0\vec{M}_e\delta^3(\vec{r}), \qquad (146)$$

where the last term represents the delta-function moment of a point dipole at the origin. The field for an array of ideal electric dipoles follows

directly from the magnetic dipole analogue as,

$$E_x = \frac{1}{4\pi\epsilon_0} \sum_{i=1}^{N_d} \frac{1}{r'^5} \left[(x'^2 - \frac{r'^2}{3}) m_{ex,i} + x'y'm_{ey,i} + x'z'm_{ez,i} \right] \tag{147}$$

$$E_y = \frac{1}{4\pi\epsilon_0} \sum_{i=1}^{N_d} \frac{1}{r'^5} \left[x'y'm_{ex,i} + (y'^2 - \frac{r'^2}{3}) m_{ey,i} + y'z'm_{ez,i} \right] \tag{148}$$

$$E_z = \frac{1}{4\pi\epsilon_0} \sum_{i=1}^{N_d} \frac{1}{r'^5} \left[x'z'm_{ex,i} + y'z'm_{ey,i} + (y'^2 - \frac{r'^2}{3}) m_{ez,i} \right] \tag{149}$$

where $(m_{ex,i}, m_{ey,i}, m_{ez,i})$ are the Cartesian components of the ith electric dipole moment vector.

5.8 Array of Point Charges

In practice, dipole fields are nearly ideal at best. Physical dipoles can be generated by using two spherical conductors held at a constant potential difference ΔV. These are easy to manufacture and can be placed in various configurations. If the sphere diameter is small compared to their separation each can be approximated as a point charge. The field of an array of N such spheres can be approximated then by

$$\vec{E}(\vec{r}) = \frac{1}{4\pi\epsilon_0} \sum_{i=1}^{N} q_i \frac{\vec{r} - \vec{r_i}}{|\vec{r} - \vec{r_i}|^3} \tag{150}$$

For a spherical conductor of radius a in vacuum, the charge can be calculated directly from the potential. Inside of the conductor the $V = \Delta V$ and $|\vec{E}(\vec{r})| = 0$ for all $r < a$. At the surface of the conductor $V(a) = \Delta V = q/4\pi\epsilon_0 a$. The charge is then

$$q_i = 4\pi\epsilon_0 a \Delta V. \tag{151}$$

Deflection power of the field is governed by the charge on each conducting sphere. Since the charge is determined by the product of the sphere radius and potential difference between the two spheres one can possibly minimize power requirements by simply increasing the sphere diameter. Note that this linear expression between sphere surface charge, q_i and voltage ΔV ignores the field effects on a single sphere due to neighboring spheres. However, for two spheres the charge can be scaled based on the sphere separation distance. ASPP does not automatically perform this scaling. It is left to the user to enter scaled charge values via the input file.

5.9 Array of Line Charges

ASPP is capable of calculating the electric field at any point $\vec{r}$ due to an array of N finite line charges of linear charge density λ_i and endpoints
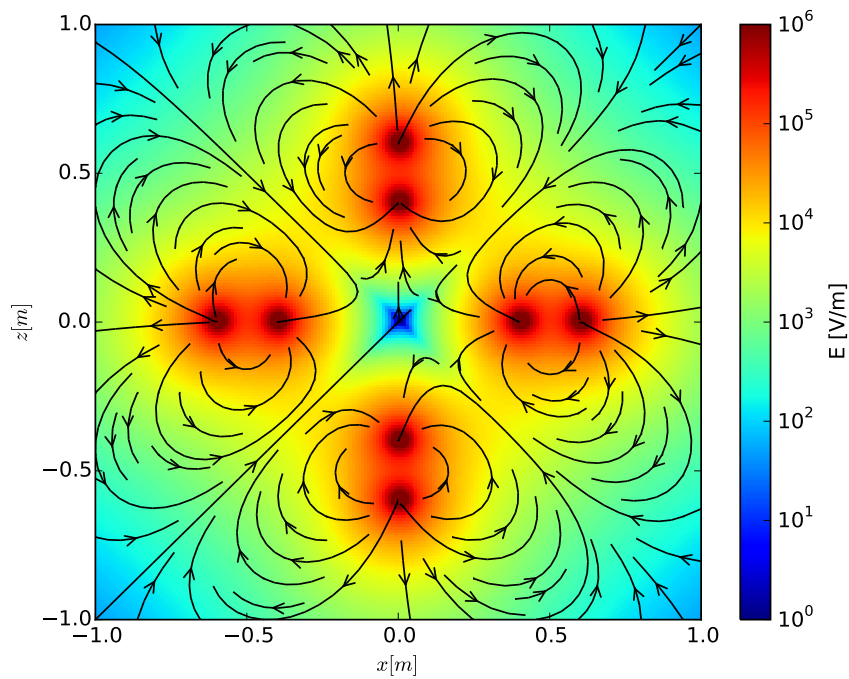
Figure 18. Electric field contour plot with field streamlines in the (x, z) plane. Each pair of 12 charges comprising the dipoles is configured with all negative charges located closest to the origin. The midpoint between each pair is located at ± 0.5 m along each of the three Cartesian axes. Each point charge has charge $q = 8.34486E - 8$ C calculated from Eq. (151) assuming each is a 1 cm sphere and $\Delta V = 75$ kV.
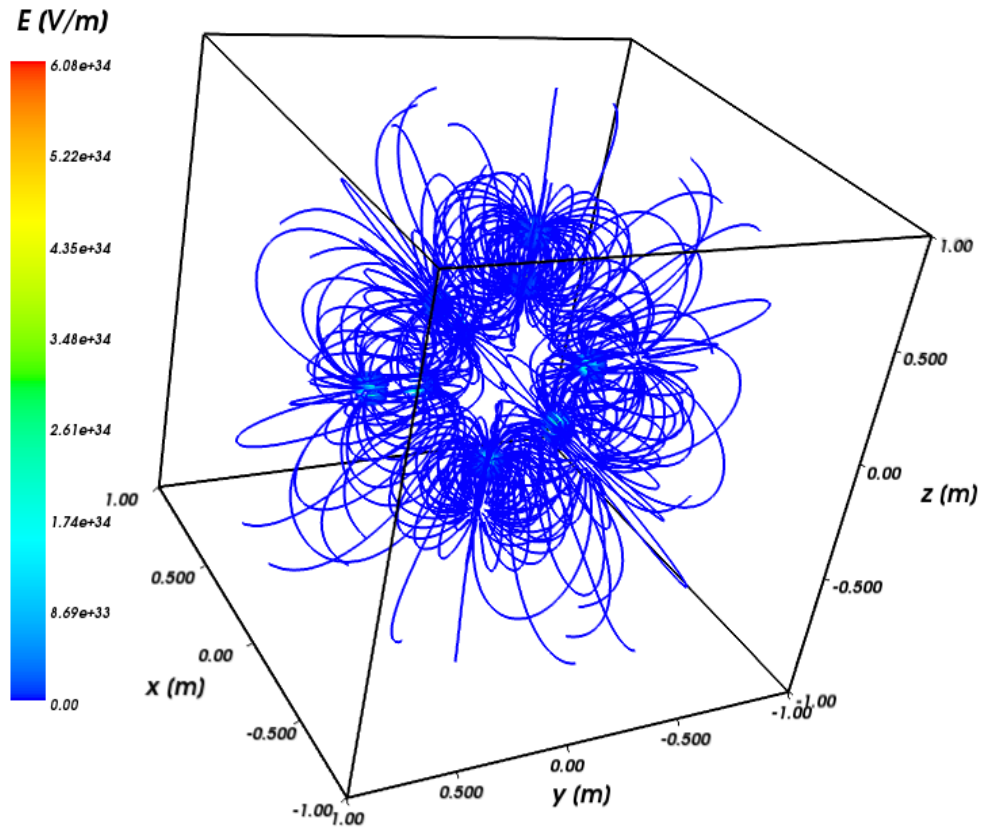
Figure 19. Electric field streamlines for six electric dipoles. Each pair of 12 charges comprising the dipoles is configured with all negative charges located closest to the origin. The midpoint between each pair is located at ±0.5 m along each of the three cartesian axes. Each point charge has charge $q = 8.34486E - 8$ C calculated from Eq. (151) assuming each is a 1 cm sphere and $\Delta V = 75$ kV.

$\vec{r}_{1,i}$ and $\vec{r}_{2,i}$. The field of an array of N finite line charges is given by,

$$\vec{E}(\vec{r}) = \frac{1}{2\pi\epsilon_0} \sum_i^N \frac{\lambda_i |\vec{r}_{2,i} - \vec{r}_{1,i}|}{(|\vec{r} - \vec{r}_{1,i}| + |\vec{r} - \vec{r}_{2,i}|)^2 - |\vec{r}_{2,i} - \vec{r}_{1,i}|^2} \tag{152}$$
$$\times \left(\frac{\vec{r} - \vec{r_{1,i}}}{|\vec{r} - \vec{r}_{1,i}|} + \frac{\vec{r} - \vec{r_{2,i}}}{|\vec{r} - \vec{r}_{2,i}|} \right)$$

5.9.1 Array of Rectangular Charges

ASPP is also able to determine the electric field at any arbitrary point in space $\vec{r} = x\hat{\imath} + y\hat{\jmath} + z\hat{k}$ due to a finite charged plane of user-defined surface charge density σ, width a, height b, midpoint $\vec{r}_m = x_m\hat{\imath} + y_m\hat{\jmath} + z_m\hat{k}$, normal vector $\vec{w} = w_x\hat{\imath} + w_y\hat{\jmath} + w_z\hat{k}$, and rectangular orientation vector $\vec{u} = u_x\hat{\imath} + u_y\hat{\jmath} + u_z\hat{k}$, which specifies the direction in which the length-a edge of the rectangular charge sheet points, was added to the code. Note that $\vec{u}$ must be orthogonal to $\vec{w}$, i.e., $\vec{u} \cdot \vec{w} = 0$. Also note that the user may specify an arbitrary number N of finite charged plane structures within ASPP.

The following steps are taken to determine $\vec{E}(\vec{r})$ due to N independent, charged planes. Let n be an index representing the nth plane in the planar charge array. First, the normal vector $\vec{w}_n$ is normalized to the unit vector $\hat{w}_n$ via

$$\hat{w}_n = \frac{\vec{w}_n}{|\vec{w}_n|}. \tag{153}$$

The normalized rectangular orientation vector $\hat{u}_n$ is found in a similar manner. $\hat{w}_n$ defines the vector normal to the nth rectangular charged sheet. The vector $\hat{v}_n$ may be found via the cross product $\hat{v}_n = \hat{w}_n \times \hat{u}_n$. The unit vectors $\hat{u}_n$, $\hat{v}_n$, and $\hat{w}_n$ define the planar reference frame whose origin is located at the midpoint $(x_{m,n}, y_{m,n}, z_{m,n})$ of the nth rectangular charged sheet, as shown in Figure 20.

Now that the planar reference frame is known, the coordinates of the point of interest are transformed from the absolute reference frame into the planar reference frame from $((x, y, z) \mapsto (u, v, w))$,

$$u_n = (x - x_{m,n})u_{x,n} + (y - y_{m,n})u_{y,n} + (z - z_{m,n})u_{z,n} \tag{154}$$
$$v_n = (x - x_{m,n})v_{x,n} + (y - y_{m,n})v_{y,n} + (z - z_{m,n})v_{z,n} \tag{155}$$
$$w_n = (x - x_{m,n})w_{x,n} + (y - y_{m,n})w_{y,n} + (z - z_{m,n})w_{z,n}. \tag{156}$$

The notation $u_{x,n}$ refers to the x-component of the normalized vector $\hat{u}_n$. Knowledge of u_n, v_n, and w_n allows for the calculation of the electric field vector components in the planar reference frame, $E_{u,n}$, $E_{v,n}$, and $E_{w,n}$. For ease of reading, consider the placeholder variables provided in Table 2. The components of the field in the frame of reference of the
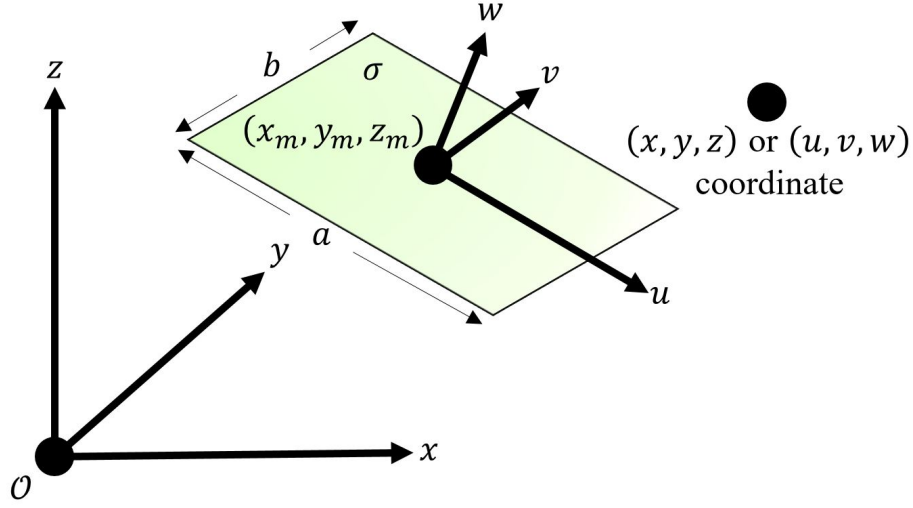
Figure 20. Absolute (x, y, z) and planar (u, v, w) reference frames depicted with example rectangular charged sheet of dimensions $[a \times b]$ and surface charge density σ.

planar sheet, $(E_{u,n}, E_{v,n}, E_{w,n})$ are calculated using

$$E_{u,n} = \frac{\sigma_n}{4\pi\epsilon_0} \left[\ln(\eta_1 - \beta_-) - \ln(\eta_2 - \beta_-) \right. \tag{157}$$

$$\left. - \ln(\eta_3 - \beta_+) + \ln(\eta_4 - \beta_+) \right]$$

$$E_{v,n} = \frac{\sigma_n}{4\pi\epsilon_0} \left[\ln(\eta_1 - \alpha_-) - \ln(\eta_2 - \alpha_+) \right. \tag{158}$$

$$\left. - \ln(\eta_3 - \alpha_-) + \ln(\eta_4 - \alpha_+) \right]$$

$$E_{w,n} = \frac{\sigma_n}{4\pi\epsilon_0} \left[\tan^{-1}\left(\frac{\alpha_- \beta_-}{2 w_n \eta_1} \right) + \tan^{-1}\left(\frac{-\alpha_+ \beta_-}{2 w_n \eta_2} \right) \right. \tag{159}$$

$$\left. - \tan^{-1}\left(\frac{\alpha_- \beta_+}{2 w_n \eta_3} \right) - \tan^{-1}\left(\frac{-\alpha_+ \beta_+}{2 w_n \eta_4} \right) \right].$$

The electric field components due to the nth rectangular charged sheet in the planar reference frame $E_{u,n}$, $E_{v,n}$, and $E_{w,n}$ are then used to calculate the $E_{x,n}$, $E_{y,n}$, and $E_{z,n}$, the electric field components due to the nth rectangular charged sheet in the absolute reference frame. $E_{x,n}$, $E_{y,n}$, and $E_{z,n}$ are then given by

$$E_{x,n} = E_{u,n} u_{x,n} + E_{v,n} v_{x,n} + E_{w,n} w_{x,n} \tag{160}$$

$$E_{y,n} = E_{u,n} u_{y,n} + E_{v,n} v_{y,n} + E_{w,n} w_{y,n} \tag{161}$$

$$E_{z,n} = E_{u,n} u_{z,n} + E_{v,n} v_{z,n} + E_{w,n} w_{z,n}. \tag{162}$$

To find the electric field due to all N rectangular charged planes, $\vec{E}(\vec{r})$,

Parameter	Value
α_-	$2u_n - a_n$
α_+	$2u_n + a_n$
β_-	$2v_n - b_n$
β_+	$2v_n + b_n$
η_1	$\sqrt{\alpha_-^2 + \beta_-^2 + 4w_n^2}$
η_2	$\sqrt{\alpha_+^2 + \beta_-^2 + 4w_n^2}$
η_3	$\sqrt{\alpha_-^2 + \beta_+^2 + 4w_n^2}$
η_4	$\sqrt{\alpha_+^2 + \beta_+^2 + 4w_n^2}$

Table 2. Placeholder parameter definitions used to calculate $E_{u,n}$, $E_{v,n}$, and $E_{w,n}$.

we sum over all planar sheets of charge as,

$$\vec{E}(\vec{r}) = \sum_n^N \vec{E}_n(\vec{r}) \tag{163}$$

6   Conclusions and Future Work

We have presented here the development of a particle tracking code aimed at enabling rapid prototyping of active shielding configurations. The code is portable across the Linux, Mac OS and Windows platforms as long as each is configured to support OpenMP, MPI and CUDA architectures.

One of the future additions is the incorporation of fields generated with finite-element methods. Work is in progress to generate accurate field maps on un-structured grids of realistic conductor shapes and configurations. This will add a large degree of flexibility to the utility of ASPP, even beyond the use for active shielding studies. Unstructured grids will allow for fast propagation as the time step will be adapted in real-time to minimize total propagation time while retaining precision.

Another avenue of development is to add the OpenCL libraries to support non-NVIDIA GPU architecture. This will however, take time as it requires a large code re-write.

Appendix A

Installation and Code Execution

A.1 Dependencies

Required

- CMake version 3.7 or later (`http://www.cmake.org`). CMake configures ASPP for the platform, links to dependencies, and turns options on or off depending on the computer platform.

Optional

- For GPU installation, at least one NVIDIA GPU with Cuda Capable GPUs and Driver (`https://developer.nvidia.com/cuda-downloads`) are required.

- For multithread installations, an OpenMP Compatible Compiler (Intel is required. GCC and Intel compilers support OpenMP natively; however, Clang compilers require building with OpenMP. Linux distributions GCC natively supports OpenMP. Mac users may either download the Intel compiler, use MACPorts to download GCC, or recompile Clang with OpenMP (`https://clang-omp.github.io`).

- For multi-node computing, an MPI compatible compiler is required, such as OpenMPI (`https://www.open-mpi.org`, MPICH (`https://www.mpich.org`), Intel (`https://software.intel.com/`), etc.

A.2 Installation Instructions

Download source from github.

```
git clone https://github.com/mll36/ASPP.git
```

Create a build directory.

```
mkdir build
```

Change directory to build folder.

```
cd build
```

Run CMake to configure project.

```
cmake ../ASPP
```

CMake automatically recognizes if CUDA, OpenMP, and MPI are available, if not ASPP will compile without them. CUDA support may be turned on or off using:

```
-DBUILD_CUDA=OFF
```

MPI support may be turned on or off using:

```
-DBUILD_MPI=OFF
```

OpenMP support may be turned on or off using:

```
-DBUILD_OpenMP=OFF
```

To use another compiler use the DCMAKE_CXX_COMPILER option. For example:

```
cmake ../ASPP -DCMAKE_CC_COMPILER=...path-to-compiler../gcc
↪    -DCMAKE_CXX_COMPILER=...path-to-compiler../g++
```

or you can compile using Intel Compilers by typing:

```
CC=icc CXX=icpc cmake ../ASPP
```

Run make. For example, to make with 4 build threads, type:

```
make -j4
```

A.3   Code Execution

To run the application, type:

```
./aspp
```

To run OpenMP with multiple threads, verify the application has been compiled with OpenMP by a supported compiler. The application will automatically grab the maximum number of threads available. To specify the number of threads used by ASPP, set the environmental variable OMP_NUM_THREADS to your desired number of threads. For example, in bash:

```
export OMP_NUM_THREADS=16
```

To run the MPI version on PBS job scheduler with 10 nodes from the command line, type:

```
mpirun -np 10 -machinefile $PBS_NODEFILE -bynode ./aspp -i
↪    inputfile.txt
```

This command can vary by MPI implementation and scheduler. The option -machinefile gives MPI the list of nodes to run on, -bynode specifies one MPI version per node. Optimal loading is one simulation per node since the application will thread out to maximize threads on each node.

The program defaults to using Earth's magnetic field, Velocity Verlet method, and a launch radius of 10 Earth radii. Simulation parameters are best altered using an input file, but some options may be altered via command line. Command Line arguments are in the form of (-option) (argument). Valid command line arguments are:

- -i Input Filename

- -p Number of Particles

- -s Number of Steps

- -d Time Interval of Each Step

- -r Random Number Seed (8 digits)

- -m Propagation Method: Euler, Leap, Verlet, Gear, Runge, Boris

- -h Displays help screen

A.4 Sample Input Files

Input files consist of commands and options separated by spaces. The first unbroken string of characters in any line is the *command* and all subsequent character strings are *options*. Here, options are given descriptive names and enclosed by angle brackets <...>. Angle brackets are not used within the input files themselves. Blank lines and lines which begin with the pound (#) symbol are ignored. Some commands have default parameters that are used when left unspecified. These will be noted.

A.4.1 Propagation Options

```
# Run_Type -- Select architecture: CPU (default) or GPU
Run_Type <architecture>
# Method -- Select propagation method: Euler, Leap, Verlet
↪  (default), Gear, Runge, Boris.
Method  <propagationMethod>
# Particles -- Specify number of particles.
Particles <numberParticles>
# Steps -- Specify number of time steps.
Steps <numberSteps>
# Time_Step -- Specify time step duration [seconds].
Time_Step <stepDuration>
# Random_Seed -- Specify random seed (8 digit integer).
Random_Seed <8DigitInteger>
# Launch_Radius -- Specify radius of source sphere (3d mode);
↪  specify position of disk source along chosen Beam_Axis (1d
↪  mode).
Launch_Radius <launchRadius>
# Crew_Volume_Radius -- Specify radius of crew volume [meters]
↪  (3d mode).
Crew_Volume_Radius <crewRadius>
```

A.4.2 Particle Properties

```
# Ion -- Select fully-stripped ion using abbreviation: e-, H, He,
↪  Li ... As.
Ion <elementSymbol>
```

```
# IonAZ -- Select ion via mass number (A), atomic number (Z), and
↪   number of electrons.
IonAZ <Z> <A> <numberElectrons>
# Energy -- Initial kinetic energy of projectile ion [MeV].
Energy <energy>
```

A.4.3 Beam Simulation Options

```
### Note that if Beam_Simulation or Beam_Nonlaminar are not
↪   specified, ASPP generates an isotropic radiation field by
↪   default (3d mode).
# Beam_Simulation -- Generates laminar, monodirectional disk
↪   source parallel to Beam_Axis (1d mode).
Beam_Simulation
# Beam_Nonlaminar -- Generates non-laminar beam.
Beam_Nonlaminar
# Beam_Axis -- Axis on which beam is centered: x, -x, y, -y, z,
↪   or -z (e.g., if x is specified, particles are born on disk
↪   centered somewhere along positive x-axis and travel in
↪   negative x-direction).
Beam_Axis <axis>
# Beam_Width -- Radius of beam.
Beam_Width <beamRadius>
# Detector_Position -- (x, y, z) point which exists in the front
↪   plane of the detector slab [meters]; also serves as a normal
↪   vector for the detector slab.
Detector_Position <x> <y> <z>
# Detector_Width -- Thickness of the detector slab [meters].
Detector_Width <thickness>
# Oblique_Detector -- Allows for detector slabs whose normal
↪   vectors are not aligned or anti-aligned with any of the three
↪   Cartesian axes; if not specified, the front face of the slab
↪   detector is set at the <x> specified for Detector_Position
↪   and its normal vector is aligned with the x-axis.
Oblique_Detector
```

A.4.4 Output Options

```
# Output_Prefix -- Prefix attached to all output files.
Output_Prefix <prefix>
# No_Initial_Positions -- Initial positions and velocities are
↪   not written to <prefix>-initposvelocities.csv.
No_Initial_Positions
# No_Final_Positions -- <prefix>-closest.csv file is not
↪   generated.
No_Final_Positions
```

A.4.5 Magnetic Field Options

```
# Magnetic_Field_Type -- Select magnetic field type: Constant,
↪   Spherical, Dipole (default), Loop, Helmholtz, Maxwell, User.
Magnetic_Field_Type <magFieldType>
```

Constant Magnetic Field Options

```
# Bx_By_Bz -- Magnetic field strength components [Tesla].
Bx_By_Bz <Bx> <By> <Bz>
```

Spherical Azimuthal Magnetic Field Options

```
# Magnetic_Bnot -- Magnetic field strength [Tesla].
Magnetic_Bnot <B0>
```

Ideal Magnetic Dipole Field Options

```
# Magnetic_Field_Coefficient -- Magnetic field strength [Tesla].
Magnetic_Field_Coefficient <C>
```

Current Loop Field Options

```
# Magnetic_Bnot -- Magnetic field strength [Tesla].
Magnetic_Bnot <B0>
# Magnetic_Loop_Radius -- Current loop radius [meters].
Magnetic_Loop_Radius <loopRadius>
```

Helmholtz Field Options

```
# Magnetic_Bnot -- Magnetic field strength [Tesla].
Magnetic_Bnot <B0>
# Magnetic_Loop_Radius -- Current loop radius [meters].
Magnetic_Loop_Radius <loopRadius>
# Magnetic_Coil_1 -- Coil 1 (x, y, z) position [meters].
Magnetic_Coil_1 <x> <y> <z>
# Magnetic_Coil_2 -- Coil 2 (x, y, z) position [meters].
Magnetic_Coil_2 <x> <y> <z>
# Magnetic_Coil_3 -- Coil 3 (x, y, z) position [meters].
Magnetic_Coil_3 <x> <y> <z>
# Magnetic_Coil_4 -- Coil 4 (x, y, z) position [meters].
Magnetic_Coil_4 <x> <y> <z>
# Magnetic_Coil_5 -- Coil 5 (x, y, z) position [meters].
Magnetic_Coil_5 <x> <y> <z>
# Magnetic_Coil_6 -- Coil 6 (x, y, z) position [meters].
Magnetic_Coil_6 <x> <y> <z>
```

Maxwell Magnetic Field Options

```
# Magnetic_Bnot -- Magnetic field strength [Tesla].
Magnetic_Bnot <B0>
# Magnetic_Loop_Radius -- Current loop radius [meters].
Magnetic_Loop_Radius <loopRadius>
# Magnetic_Maxwell_Large_Coil -- Large coil radius [meters].
Magnetic_Maxwell_Large_Coil <largeCoilRadius>
```

User Defined Magnetic Field Options

```
# Magnetic_Field_Filename -- Path to TSV data file for magnetic
↪   field strengths.
Magnetic_Field_Filename <filePath>
# Magnetic_Min -- (x, y, z) position of the minimum magnetic
↪   field values [meters].
Magnetic_Min <x> <y> <z>
# Magnetic_Max -- (x, y, z) position of the maximum magnetic
↪   field values [meters].
Magnetic_Max <x> <y> <z>
# Magnetic_Delta -- Linear increments for magnetic field grid
↪   [meters].
Magnetic_Delta <dx> <dy> <dz>
# Magnetic_Number_Points -- Number of grid points for each axis.
Magnetic_Number_Points <Nx> <Ny> <Nz>


### Data file structure:
### The user-defined magnetic field data file must be a TSV (tab
↪   separated value) file with the following structure:
Bx1     By1     Bz1
Bx2     By2     Bz2
### The positions of the magnetic field strengths must conform to
↪   the defined grid. Assume a 10 by 10 by 10 grid with linear
↪   spacing was constructed. The following structure shows which
↪   magnetic field vectors correspond to each position in the
↪   grid.
(Bx1, By1, Bz1) at (xMin, yMin, zMin)
(Bx2, By2, Bz2) at (xMin, yMin, zMin + 1*dz)
...
(Bx10, By10, Bz10) at (xMin, yMin, zMax)
(Bx11, By11, Bz11) at (xMin, yMin + 1*dy, zMin)
...
(Bx20, By20, Bz20) at (xMin, yMin + 1*dy, zMax)
(Bx21, By21, Bz21) at (xMin, yMin + 2*dy, zMin)
...
(Bx100, By100, Bz100) at (xMin, yMax, zMax)
(Bx101, By101, Bz101) at (xMin + 1*dx, yMin, zMin)
...
(Bx1000, By1000, Bz1000) at (xMax, yMax, zMax)
```

A.4.6   Electric Field Options

```
# Electric_Field_Type -- Select electric field type: Constant,
↪   Dipole, Dipole_Array, Point_Charge, Line_Charge,
↪   Plane_Charge, LineAndPoint_Charge, PlaneAndPoint_Charge,
↪   PlaneAndLine_Charge, PlaneAndLineAndPoint_Charge, User.
```

Constant Electric Field Options

```
# Ex_Ey_Ez -- Electric field strength components [volts per
↪   meter].
Ex_Ey_Ez <Ex> <Ey> <Ez>
```

Ideal Electric Dipole Field Options

```
# Electric_Field_Coefficient -- Electric field coefficient for
↪ ideal dipole field [volts per meter].
Electric_Field_Coefficient <C>
```

Ideal Electric Dipole Array Field Options

```
# Electric_Dipole_Array -- Dipole positions [meters] and
↪ strengths [volts per meter].
Electric_Dipole_Array <x> <y> <z> <Ex> <Ey> <Ez>
### Arrays may also be specified using a separate data file.
# Electric_Field_Filename -- path to TSV data file for ideal
↪ electric dipole parameters.
Electric_Field_Filename <filePath>
### Data file structure (tab separated values):
x1    y1    z1    Ex1    Ey1    Ez1
x2    y2    z2    Ex2    Ey2    Ez2
...
```

Electric Point Charge Array Field Options

```
# Electric_Point_Array -- Point charge strengths [Coulombs] and
↪ positions [meters].
Electric_Point_Array <Q> <x> <y> <z>
### Arrays may also be specified using a separate data file.
# Electric_Field_Filename -- path to TSV data file for point
↪ charge parameters.
Electric_Field_Filename <filePath>
### Data file structure (tab separated values):
Q1    x1    y1    z1
Q2    x2    y2    z2
...
```

Electric Line Charge Array Field Options

```
# Electric_Line_Array -- Linear charge density [Coulombs per
↪ meter], line endpoint positions 1 and 2 [meters].
Electric_Line_Array <L> <xStart> <yStart> <zStart> <xEnd> <yEnd>
↪ <zEnd>
### Arrays may also be specified using a separate data file.
# Electric_Field_Filename -- path to TSV data file for point
↪ charge parameters.
Electric_Field_Filename <filePath>
### Data file structure (tab separated values):
L1    xStart1    yStart1    zStart1    xEnd1    yEnd1    zEnd1
L2    xStart2    yStart2    zStart2    xEnd2    yEnd2    zEnd2
...
```

Electric Plane Charge Array Field Options

```
# Electric_Plane_Array -- Surface charge density [Coulombs per
↪   square meter], uWidth [meters], vWidth [meters], rectangle
↪   midpoint [meters], rectangle normal vector [unitless], vector
↪   which points parallel to side of length uWidth [unitless].
Electric_Plane_Array <S> <uWidth> <vWidth> <xm> <ym> <zm> <nx>
↪   <ny> <nz> <ux> <uy> <uz>
### Arrays may also be specified using a separate data file.
# Electric_Field_Filename -- path to TSV data file for point
↪   charge parameters.
Electric_Field_Filename <filePath>
### Data file structure (tab separated values):
S1    uWidth1    vWidth1    xm1    ym1    zm1    nx1    ny1
↪   nz1    ux1    uy1    uz1
S2    uWidth2    vWidth2    xm2    ym2    zm2    nx2    ny2
↪   nz2    ux2    uy2    uz2
...
```

Electric Line and Point Charge Array Field Options

```
# Electric_LineAndPoint_Array -- Combination of line and point
↪   charges.
Electric_LineAndPoint_Array <L> <xStart> <yStart> <zStart> <xEnd>
↪   <yEnd> <zEnd> <Q> <x> <y> <z>
### Arrays may also be specified using a separate data file.
# Electric_Field_Filename -- path to TSV data file for point
↪   charge parameters.
Electric_Field_Filename <filePath>
### Data file structure (tab separated values):
L1    xStart1    yStart1    zStart1    xEnd1    yEnd1    zEnd1
↪   Q1    x1    y1    z1
L2    xStart2    yStart2    zStart2    xEnd2    yEnd2    zEnd2
↪   Q2    x2    y2    z2
...
```

Electric Plane and Point Charge Array Field Options

```
# Electric_PlaneAndPoint_Array -- Combination of plane and point
↪   charges.
Electric_PlaneAndPoint_Array <S> <uWidth> <vWidth> <xm> <ym> <zm>
↪   <nx> <ny> <nz> <ux> <uy> <uz> <Q> <x> <y> <z>
### Arrays may also be specified using a separate data file.
# Electric_Field_Filename -- path to TSV data file for point
↪   charge parameters.
Electric_Field_Filename <filePath>
### Data file structure (tab separated values):
S1    uWidth1    vWidth1    xm1    ym1    zm1    nx1    ny1
↪   nz1    ux1    uy1    uz1    Q1    x1    y1    z1
S2    uWidth2    vWidth2    xm2    ym2    zm2    nx2    ny2
↪   nz2    ux2    uy2    uz2    Q2    x2    y2    z2
...
```

Electric Plane and Line Charge Array Field Options

```
# Electric_PlaneAndLine_Array -- Combination of plane and line
↪   charges.
Electric_PlaneAndLine_Array <S> <uWidth> <vWidth> <xm> <ym> <zm>
↪   <nx> <ny> <nz> <ux> <uy> <uz> <L> <xStart> <yStart> <zStart>
↪   <xEnd> <yEnd> <zEnd>
### Arrays may also be specified using a separate data file.
# Electric_Field_Filename -- path to TSV data file for point
↪   charge parameters.
Electric_Field_Filename <filePath>
### Data file structure (tab separated values):
S1    uWidth1    vWidth1    xm1    ym1    zm1    nx1    ny1
↪   nz1    ux1    uy1    uz1    L1    xStart1    yStart1
↪   zStart1    xEnd1    yEnd1    zEnd1
S2    uWidth2    vWidth2    xm2    ym2    zm2    nx2    ny2
↪   nz2    ux2    uy2    uz2    L2    xStart2    yStart2
↪   zStart2    xEnd2    yEnd2    zEnd2
...
```

Electric Plane and Line and Point Charge Array Field Options

```
# Electric_PlaneAndLineAndPoint_Array -- Combination of plane,
↪   line, and point charges.
Electric_PlaneAndLineAndPoint_Array <S> <uWidth> <vWidth> <xm>
↪   <ym> <zm> <nx> <ny> <nz> <ux> <uy> <uz> <L> <xStart> <yStart>
↪   <zStart> <xEnd> <yEnd> <zEnd> <Q> <x> <y> <z>
### Arrays may also be specified using a separate data file.
# Electric_Field_Filename -- path to TSV data file for point
↪   charge parameters.
Electric_Field_Filename <filePath>
### Data file structure (tab separated values):
S1    uWidth1    vWidth1    xm1    ym1    zm1    nx1    ny1
↪   nz1    ux1    uy1    uz1    L1    xStart1    yStart1
↪   zStart1    xEnd1    yEnd1    zEnd1    Q1    x1    y1    z1
S2    uWidth2    vWidth2    xm2    ym2    zm2    nx2    ny2
↪   nz2    ux2    uy2    uz2    L2    xStart2    yStart2
↪   zStart2    xEnd2    yEnd2    zEnd2    Q2    x2    y2    z2
...
```

User Defined Electric Field Options

```
# Electric_Field_Filename -- path to TSV data file for electric
↪   field strengths.
Electric_Field_Filename <filePath>
# Electric_Min -- (x, y, z) position of the minimum electric
↪   field values [meters].
Electric_Min <x> <y> <z>
# Electric_Max -- (x, y, z) position of the maximum electric
↪   field values [meters].
Electric_Max <x> <y> <z>
```

```
# Electric_Delta -- Linear increments for electric field grid
↪   [meters].
Electric_Delta <dx> <dy> <dz>
# Electric_Number_Points -- Number of grid points for each axis.
Electric_Number_Points <Nx> <Ny> <Nz>


### Data file structure:
### The user defined electric field data file must be a TSV (tab
↪   separated value) file with the following structure:
Ex1     Ey1     Ez1
Ex2     Ey2     Ez2
### The positions of the electric field strengths must conform to
↪   the defined grid. Assume a 10 by 10 by 10 grid with linear
↪   spacing was constructed. The following structure shows which
↪   electric field vectors correspond to each position in the
↪   grid.
(Ex1, Ey1, Ez1) at (xMin, yMin, zMin)
(Ex2, Ey2, Ez2) at (xMin, yMin, zMin + 1*dz)
...
(Ex10, Ey10, Ez10) at (xMin, yMin, zMax)
(Ex11, Ey11, Ez11) at (xMin, yMin + 1*dy, zMin)
...
(Ex20, Ey20, Ez20) at (xMin, yMin + 1*dy, zMax)
(Ex21, Ey21, Ez21) at (xMin, yMin + 2*dy, zMin)
...
(Ex100, Ey100, Ez100) at (xMin, yMax, zMax)
(Ex101, Ey101, Ez101) at (xMin + 1*dx, yMin, zMin)
...
(Ex1000, Ey1000, Ez1000) at (xMax, yMax, zMax)
```

A.4.7   Additional Options

```
# Shadow_Mode -- Marks particles which intersect with a sphere of
↪   radius <shadowRadius> [meters] surrounding any point charge.
Shadow_Mode <shadowRadius>
# Angular_Binning -- Records the direction at which any particle
↪   enters the sphere of radius <angularBinningRadius> [meters]
↪   as well as the total distance [meters] said particle traveled
↪   within said sphere; allows for the estimation of the incident
↪   angular flux distribution (3d mode). Note that the "1" option
↪   is currently static.
Angular_Binning 1 <angularBinningRadius>
# Track_Trajectory -- Generates a <prefix>-trajectory.csv file
↪   where each line consists of a single particle's position
↪   information as a function of time; <stepGap> tells ASPP how
↪   many time steps to take before recording the position of each
↪   particle.
Track_Trajectory <stepGap>
# Radial_Binning -- Segments space into concentric spheres and
↪   counts the number of particles which enter each radial bin
↪   (3d mode).
Radial_Binning
```

```
# Inner_Bin -- Radius of innermost bin for radial binning scheme
↪   [meters].
Inner_Bin <innerBinRadius>
# Bin_Width -- Radial distance between subsequent radial bins
↪   [meters].
Bin_Width <binWidth>
# Number_Bins -- Number of radial bins.
Number_Bins <numberBins>
```

A.4.8 1d Beam Simulation Input File Example

```
# PROPAGATION OPTIONS
Run_Type CPU
Method Verlet
Particles 400000
Steps 150000
Time_Step 5E-12
Random_Seed 28706043
Launch_Radius 2.0

# PARTICLE PROPERTIES
IonAZ 1 1 0
Energy 2.0

# BEAM SIMULATION OPTIONS
Beam_Simulation
Beam_Axis x
Beam_Width 0.1
Detector_Position -2.0 0 0
Detector_Width 0.0005

# OUTPUT OPTIONS
Output_Prefix output

# MAGNETIC FIELD OPTIONS
Magnetic_Field_Type Constant
Bx_By_Bz 0 0 0

# ELECTRIC FIELD OPTIONS
Electric_Field_Type LineAndPoint_Charge
Electric_LineAndPoint_Array 6.4E-07 0.0 5.715E-02 0.0 0.0
↪   5.715E-02 3.048E-01 8.1E-08 0.0E+00 5.715E-02 0.0
Electric_LineAndPoint_Array -6.4E-07 0.0 -5.715E-02 0.0 0.0
↪   -5.715E-02 3.048E-01 -8.1E-08 0.0 -5.715E-02 0.0

# ADDITIONAL OPTIONS
Shadow_Mode 0.0127
```

A.5 3d Simulation Input File Example

```
# PROPAGATION OPTIONS
Run_Type GPU
```

```
Method Verlet
Particles 2500000
Steps 12500
Time_Step 5.00E-10
Random_Seed 49070871
Launch_Radius 250.0
Crew_Volume_Radius 5

# PARTICLE PROPERTIES
Ion H
Energy 30

# OUTPUT OPTIONS
Output_Prefix dakota
No_Initial_Positions
No_Final_Positions

# MAGNETIC FIELD OPTIONS
Magnetic_Field_Type Constant
Bx_By_Bz 0 0 0

# ELECTRIC FIELD OPTIONS
Electric_Field_Type PlaneAndLine_Charge
Electric_PlaneAndLine_Array 1.4e-06 30 30 35 0.0 0.0 1.0 0.0 0.0
↪   0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Electric_PlaneAndLine_Array 1.4e-06 30 30 -35 0.0 0.0 -1.0 0.0
↪   0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Electric_PlaneAndLine_Array 1.4e-06 30 30 35 7.5 0.0 0.9778
↪   0.2095 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Electric_PlaneAndLine_Array 1.4e-06 30 30 35 -7.5 0.0 0.9778
↪   -0.2095 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
Electric_PlaneAndLine_Array 0.0 30 30 20 0.0 0.0 1.0 0.0 0.0 0.0
↪   1.0 0.0 -2.3E-04 40 40 57 50 50 71
Electric_PlaneAndLine_Array 0.0 30 30 20 0.0 0.0 1.0 0.0 0.0 0.0
↪   1.0 0.0 -2.3E-04 40 -40 57 50 50 71

# ADDITIONAL OPTIONS
Radial_Binning
Inner_Bin 1
Bin_Width 1
Number_Bins 100
Angular_Binning 1 100
```

Appendix B

Integration with DAKOTA

ASPP is integrated with DAKOTA to quickly simulate the effects of variable changes on shielding effectiveness by analyzing the exclusion area. DAKOTA provides interfaces to simulations, using iterative algorithms, quickly completing uncertainty quantification, sensitivity analysis, variance analysis, and parametric studies of design spaces. Figure B1 shows a flow chart of the DAKOTA/ASPP interface, starting with input parameters through ASPP and python analysis.

As shown in Figure B1, DAKOTA required four components in input to run a simulation: variables, methods, interface, and a response block. The variable block defined the simulation variables, which for this study include continuous and discrete variables. The study used one continuous design variable of the dipole-to-detector distance, two discrete variables of ion species and ion energy, and three constant state variables of sphere voltage, sphere diameter, and sphere surface to surface diameter. The method block defined the method of analysis as a vector parameter study to quickly examine the design space. The interface block described the DAKOTA/ASPP interface driver, which used a simulator script to control the execution of the simulations.

The script altered the ASPP input file with the simulation parameters, using DAKOTA's DPREPRO utility that copied input variables for the individual simulation into a sample input file. The script next executed the ASPP simulation with ASPP outputting the detector hits as a comma-separated value (CSV) file. After which the simulator script ran a Python analysis script to create an image from the detector and to find the exclusion area using a Distance Regularized Level Set Evolution (DRLSE) method, described in Section **??**. The script then copied the results from Python for the exclusion area back to DAKOTA, which tabulated the results as the response function.

B.1   DAKOTA Input Script

```
    #  This input file performs a list parameter study of charges
   ↪  with ASPP to find void region.

environment
  tabular_graphics_data
    tabular_graphics_file 'magnification.dat'

method,
        vector_parameter_study
        step_vector =           1 0 0 0 0
          num_steps = 400
```
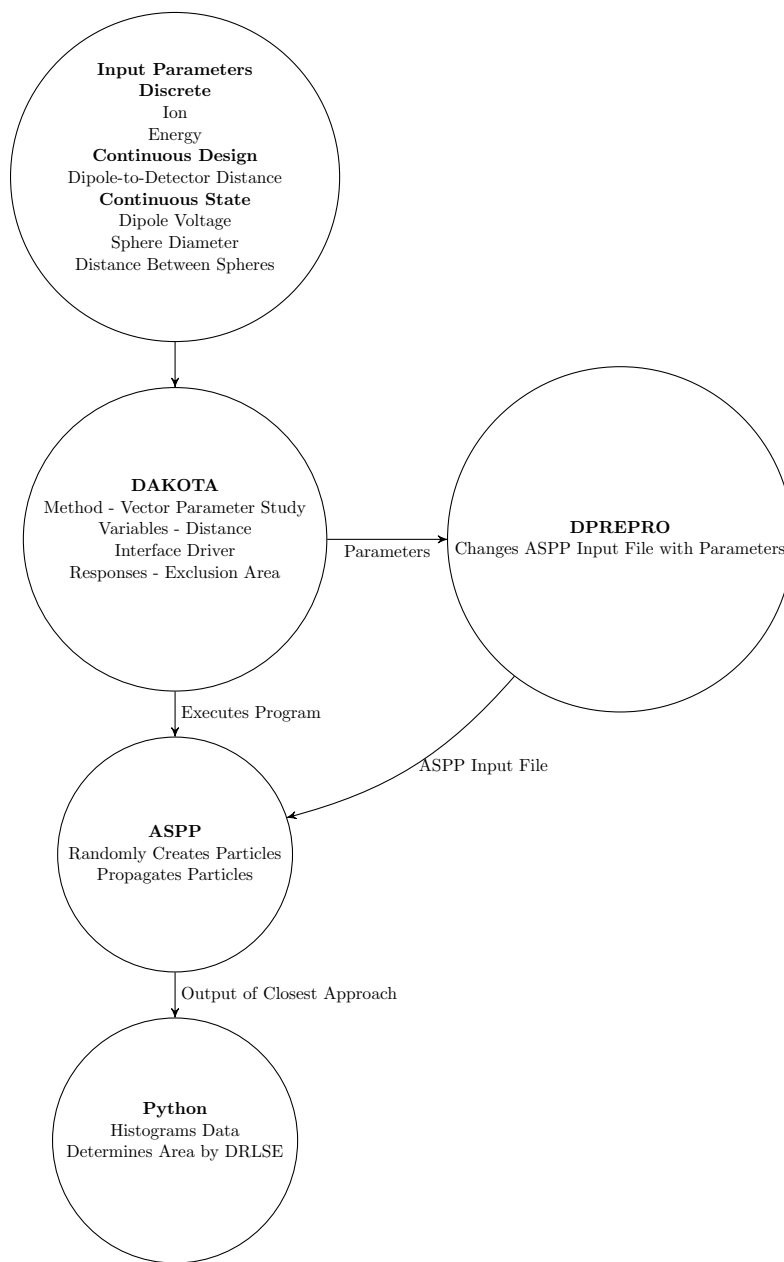
Figure B1. Flow chart of DAKOTA/ASPP interface with Python based data analysis.

```
variables,
        continuous_design = 1
        initial_point = 1
        lower_bounds    1.0
        descriptors         'distance'

        continuous_state 4
                initial_state       500         1000        0.2         0.1
                descriptor          'energy' 'voltage'
                ↪  'sphere_diameter' 'S_to_S'

interface,
        fork
          analysis_driver = 'simulator_script_aspp'
          parameters_file = 'params.in'
          results_file    = 'simulation_results.out'
                    work_directory directory_tag
#          copy_files = 'aspp.in'
# uncomment to leave params.in and results.out files in work_dir
↪   subdirectories
          named 'workdir' file_save  directory_save
          aprepro
           deactivate active_set_vector

responses,
        response_functions = 1  # Exclusion zone from python
          ↪  script
        no_gradients
        no_hessians
```

B.2 DAKOTA Simulator Script

```
#!/bin/sh
# Sample simulator to Dakota system call script
# See Advanced Simulation Code Interfaces chapter in Users Manual

# $1 is params.in FROM Dakota
# $2 is results.out returned to Dakota

# --------------
# PRE-PROCESSING
# --------------
# Incorporate the parameters from DAKOTA into the template,
↪   writing ros.in
# Use the following line if SNL's APREPRO utility is used instead
↪   of DPrePro.
# ../aprepro -c '*' -q --nowarning ros.template ros.in

dprepro $1 H_2MeV.template H_2MeV.in    # COPIES THE TEMPLATE
↪   FILE AND REPLACES VARIABLES WITH SPECIFIC VALUES, WRITES TO
↪   test.txt
```

```
# --------
# ANALYSIS
# --------

../aspp -i H_2MeV.in      # RUN ASPP WITH NORMAL COMMANDS

# ---------------
# POST-PROCESSING
# ---------------
# Extract
#awk -F "\"*,\"*" 'FNR == 1 {next}{print $1,$2}'
↪  ASPPtest-closest.csv >> results.out
python ../image_area.py -i dakota-closest.csv -o results.out
mv results.out $2
```

Appendix C

Code Benchmarks

C.1   Comparison of Integration Methods

In order to assess how well each of the integrators performs one has to pick a suitable problem. We have chose here two extremes in terms of computational complexity. The simplest case is a constant homogenous magnetic field confined to a circular region about the origin of a Cartesian coordinate system. Here, assuming no radiation of the charge, a particle undergoes constant uniform circular motion in a plane normal to the magnetic field with constant cyclotron radius and frequency. Since the field is uniform there is no need to ensure the step size is modified at each time step in order to sufficiently resolve the motion.

The second field chosen is that of an ideal magnetic dipole. There have been numerous published investigations regarding the trajectory of charged particles in Earth's magnetic field, particularly Earth-like ideal dipole fields [21–23]. Because trajectory characteristics are relatively well-understood this particular field configuration serves as a good benchmark for the developed toolset presented here. This field is inherently non-uniform and singular at the origin. Field strength increases as the particle moves towards the dipole origin and thus the cyclotron frequency and radius will change from one integration step to the next. Moreover, it is infinite in extent.

For the case of an ideal dipole field, one expects kinetic energy, E_k and hence γ to be a constant of the motion. Trajectories of a charged particle in an ideal dipole field have been extensively studied since the time of Störmer's work. For a z-aligned dipole, $x(t)$ should be periodic with high frequency 'wiggles' about each minimum and maximum. The presence of the two characteristic frequencies translates to motion in the (x, y) plane showing as a circular shell of criss-crossing trajectories.

Figures C1 and C2 show results for γ and $x(t)$ as a function of time using four different integration methods. Position units are in terms of Earth radius, R_e. For all methods except the Boris-Buneman integrator, γ increases with time. Since $\vec{E}(\vec{r}, t) = 0$, γ must be a constant of the motion since energy is conserved according to Eq. 17. $x(t)$ should show global period time-dependence with no drift and higher-frequency micro-motion around each maximum/minimum value as shown in Figure C2 (black line - Boris-Buneman integrator). Figure C3 shows the same data as in Figure C2 except over a shorter time range to highlight the higher frequency micro-motion. The AVV and RK4 integrator results show substantial drift as indicated by the changing periodicity in $x(t)$. The reason for the effects seen in both γ and $x(t)$ is due to the RK4 and

AVV (as well as standard Velocity-Verlet) methods being derived from a Taylor expansion of a strictly position-dependent conservative field. The ideal dipole field is however explicitly dependent on velocity, not position. The BB algorithm handles the velocity-dependent magnetic force in a way that relieves this constraint with the Taylor expansion.
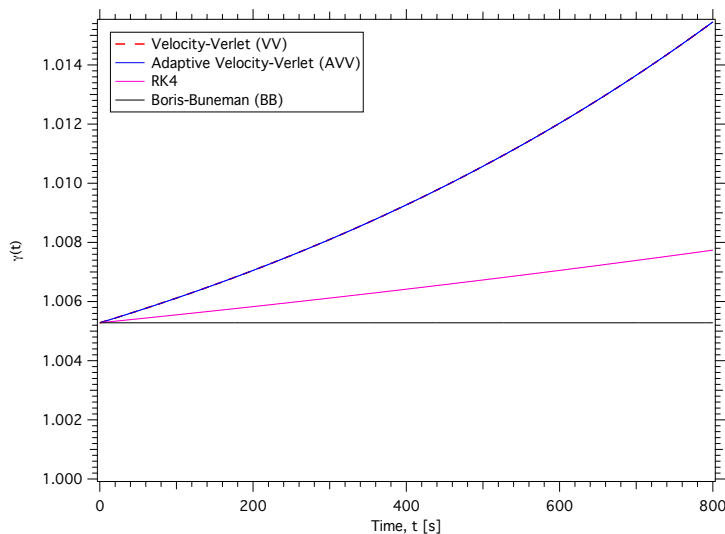
Figure C1. Lorentz factor, γ as a function of time for the Velocity-Verlet (red dash), Adaptive Velocity-Verlet (blue), Runge-Kutta (magenta) and Boris-Buneman (black) integrators using the field of an ideal magnetic dipole (Eq. 125). Only the Boris-Buneman integrator results in a γ value that is constant in time, and thus conserves energy.

The envelope of trajectories in the (x, y) plane is shown in Figures C4-C6 for the AVV, RK4 and BB integrators, respectively. The inset in each figure is an exploded view of one segment of the larger plot. Note that the envelope increases in sharpness from the AVV to BB methods. It is clear here that only the BB algorithm can capture the details of the planar trajectories as expected. The AVV integrator induces an additional precession of the trajectory over the standard drift which causes the envelope to blur. The degree of blurring is less with the RK4 but still present. The same time step Δt was used in each run with different integrators.

The canonical phase space (z, v_z) was also investigated as it should also show a confined region of points between minimum/maximum values of both z and v_z. Again, we observe similar trends as the planar trajectory envelope of Figure C4- C6. Starting with the AVV integration method (Figure C7), the phase space envelope is larger than for the RK4 and BB integrators and relatively 'rough' along the edges. The envelope decreases in amplitude for the RK4 method yet is not nearly as 'clean' as for that produced using the BB method.

68
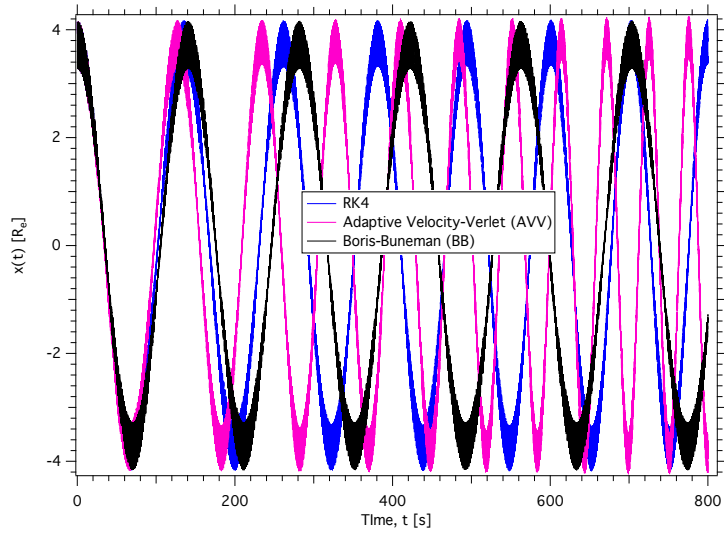
Figure C2.   Particle position along the x-axis as a function of time for Adaptive Velocity-Verlet (magenta), Runge-Kutta (blue) and Boris-Buneman (black) integrators. The field used was that of an ideal magnetic dipole, Eq. 125. Only the Boris-Buneman integrator results show equal spacing between peaks. Both the Adaptive Velocity-Verlet and Runge-Kutta show drift in position x.
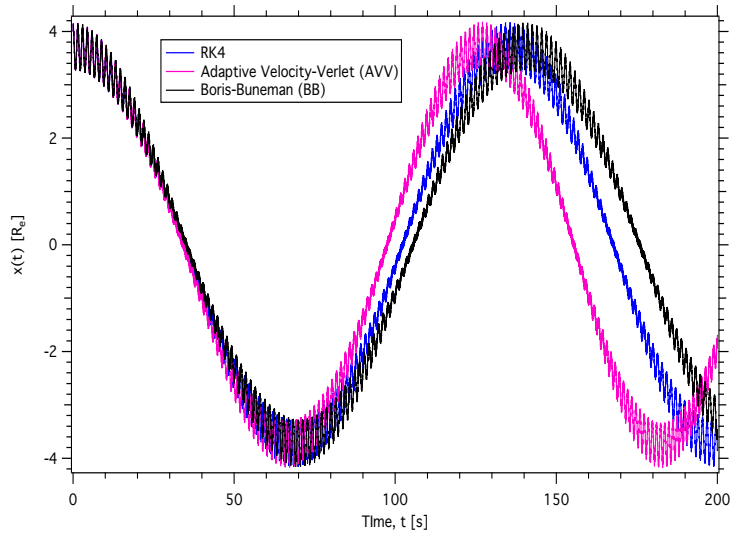
Figure C3.   Particle position along the x-axis as a function of time for Adaptive Velocity-Verlet (magenta), Runge-Kutta (blue) and Boris-Buneman (black) integrators. The field used was that of an ideal magnetic dipole, Eq. 125. Data same as that in Figure 125 except shown on a shorter time range to highlight higher frequency micro-motion.
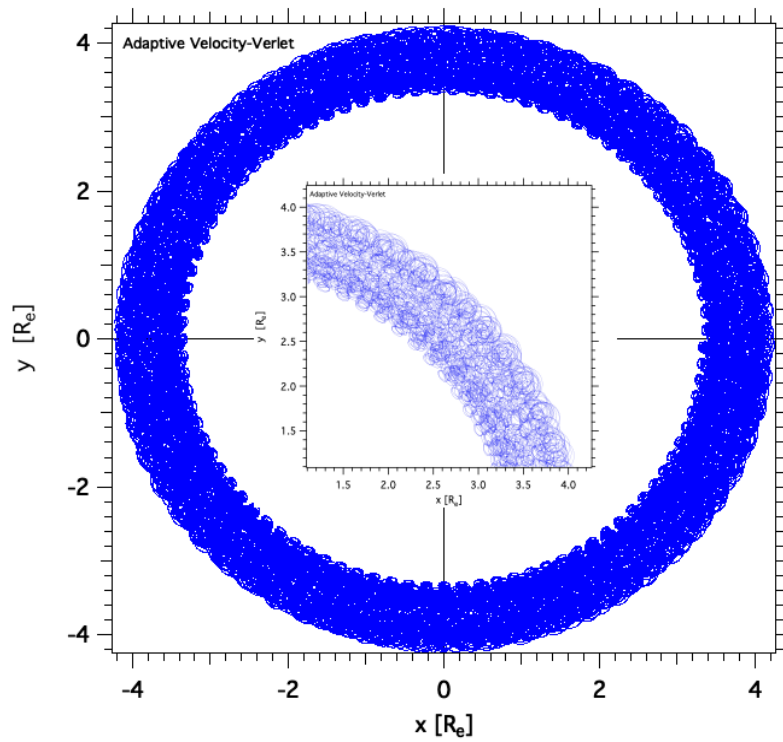
Figure C4. Particle trajectories in the (x,y) plane in the presence of an ideal magnetic dipole field, Eq. 125 generated with the AVV algorithm. The inset shows an enlarged sample of the envelope of trajectories. Note the 'noisy' appearance of the outer and inner trajectory envelope.
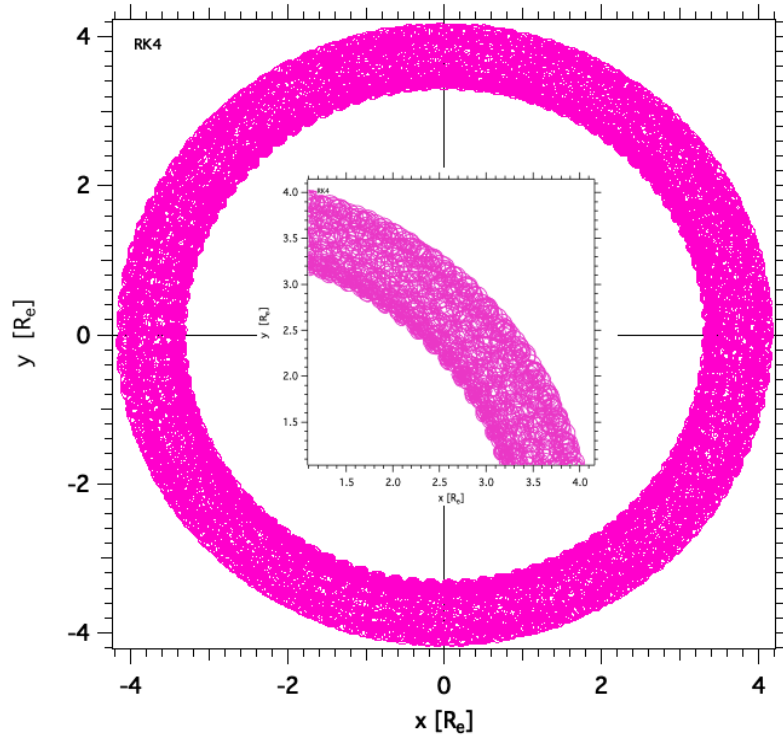
Figure C5. Particle trajectories in the (x,y) plane in the presence of an ideal magnetic dipole field, Eq. 125 generated with the RK4 algorithm. The inset shows an enlarged sample of the envelope of trajectories. Note that the envelope, while still somewhat rough, is cleaner than that produced by the AVV algorithm (see Figure C4).
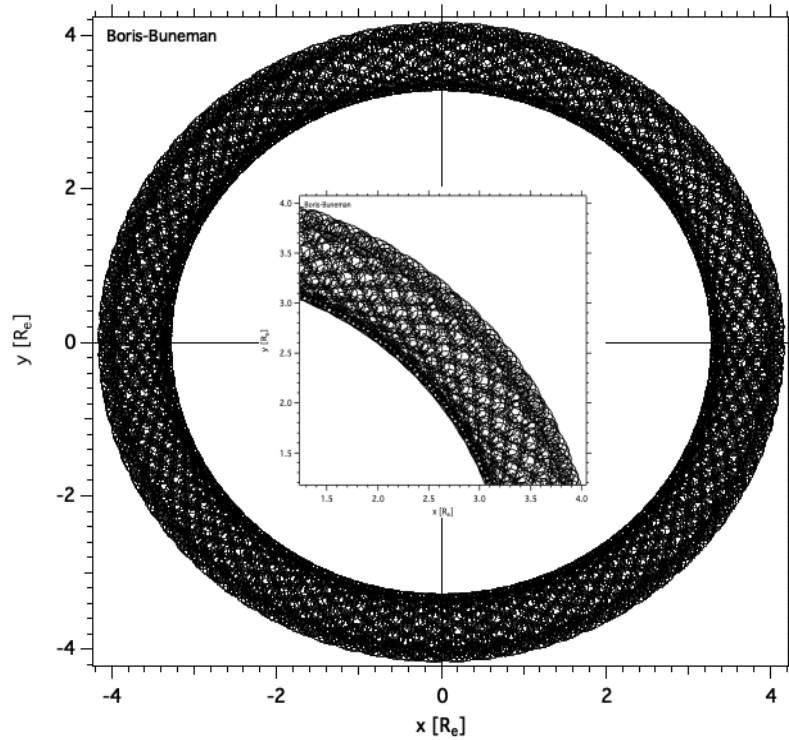
Figure C6. Particle trajectories in the (x,y) plane in the presence of an ideal magnetic dipole field, Eq. 125 generated with the BB algorithm. The inset shows an enlarged sample of the envelope of trajectories. The inner and outer envelope of trajectories is highly even as compared to the other two algorithms (Figures C4 and C5).
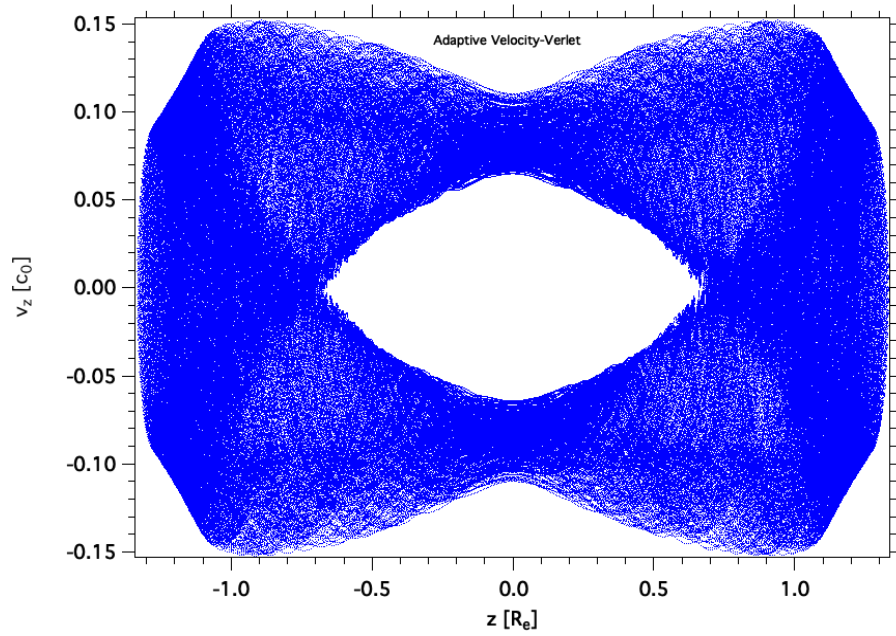
Figure C7.   Single particle canonical phase space map in z for the Adaptive Velocity Verlet algorithm. The field used was that of an ideal magnetic dipole, Eq. 125.
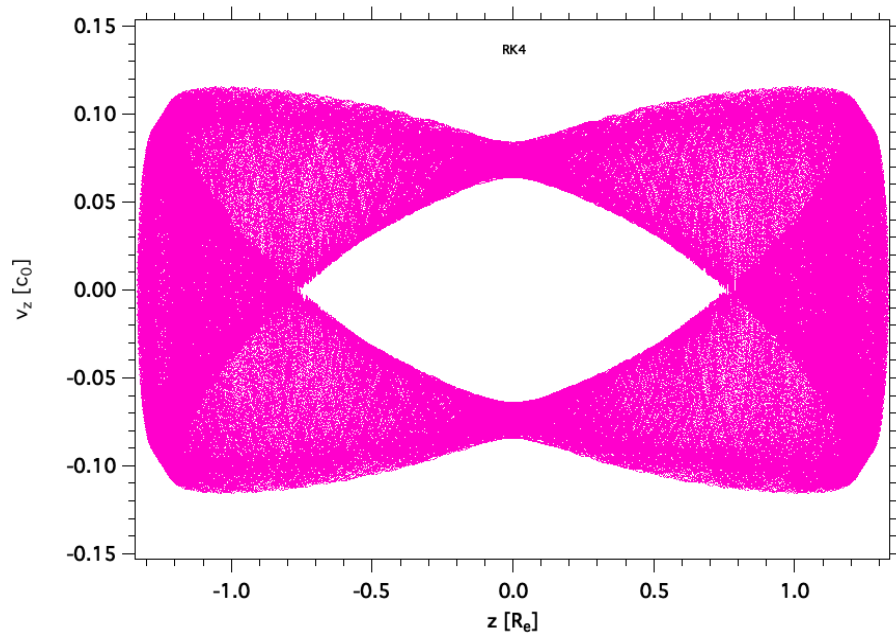
Figure C8.   Single particle canonical phase space map in z for the RK4 algorithm. The field used was that of an ideal magnetic dipole, Eq. 125.
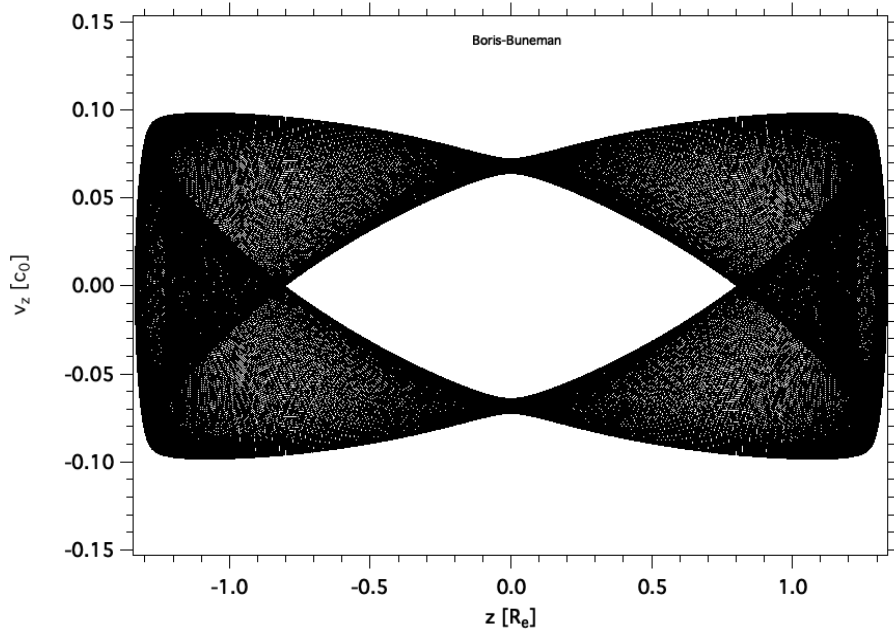
Figure C9. Single particle canonical phase space map in z for the BB algorithm. The field used was that of an ideal magnetic dipole, Eq. 125.

C.2 Run-time Benchmarks and Memory Usage

In this section, a brief overview is presented for different simulations performed in 3d using GPU compute architecture. The simulation run-time for different input parameters is shown in Tables C1 and C2. Run-time data for Table C1 corresponds to 200 MeV protons incident on a collection of point charges. The number of charges used is represented by column '#Nodes'. Electric field values are calculated using Eq. 150. Figure C10 represents the relationship between the simulation time per GPU and the complexity of the simulation. Computational complexity of the simulation is defined as the product of the number of particles, N_p, number of time steps, $N_{\Delta t}$ and the number of charged nodes present in the configuration, N_{nodes}.

Table C2 run-time data corresponds to protons incident on a collection of charged rods and planes. The proton initial kinetic energy was varied across the runs from 10 MeV to 500 MeV. The field of the charged rods is calculated by assuming an array of linear charge densities according to Section 5.9. The field of the array of planes is calculated according to Section 5.9.1.

Memory usage on a node with 4 RTX-2080Ti GPUs as a function of the number of particles simulated, N_p, is shown in Figure C11 for 100 MeV protons in Earth's magnetic field launched from a distance of ten times Earth's radius. Note the clear linear dependence on N_p. We

74

Table C1. Computation Time for 3d simulations of 200 MeV protons incident on an array of point charges using GPU architecture. The incident particle field directionality is generated isotropically. N_p is the number of particles(protons), $N_{\Delta t}$ the number of time steps, N_{GPU} the number of GPUs used and T the total simulation time (in h).

N_p	$N_{\Delta t}$	N_{nodes}	N_{GPU}	T [h]	GPU-Type
10^7	20000	9150	1	192.25	GTX-1080Ti
10^7	20000	600	1	13.07	GTX-1080Ti
2×10^7	25000	800	4	10.35	GTX-1080Ti
2×10^7	25000	1000	4	13.12	GTX-1080Ti
3.6×10^6	5000	3402	4	1.65	GTX-1080Ti
6.4×10^6	8000	4374	4	5.9	GTX-1080Ti
10^7	10000	3888	4	10.24	GTX-1080Ti
2.5×10^6	5000	1192	2	0.81	GTX-1080Ti

Table C2. Computation Time for 3d simulations of $10\,\mathrm{MeV}$ to $500\,\mathrm{MeV}$ protons incident on an array of line and planar charges using GPU architecture. The incident particle field directionality is generated isotropically. N_p is the number of particles(protons), $N_{\Delta t}$ the number of time steps, N_{GPU} the number of GPUs used, T the total simulation time (in h) and E_k the initial proton kinetic energy. The number of nodes for this configuration is shown as the number of planes (rods). The total nodes in a given run is the sum of the number of planes and number of rods.

N_p	$N_{\Delta t}$	N_{nodes}	N_{GPU}	T [h]	GPU-Type	E_k [MeV]
1.6×10^6	10000	24(8)	2	0.52	GTX-1080Ti	30
1.6×10^6	2400	24(8)	2	0.11	GTX-2080Ti	500
3.6×10^6	15000	24(8)	2	1.7	GTX-1080Ti	10
1.6×10^7	10000	18(8)	2	4.97	GTX-1080Ti	30
1.6×10^7	2400	18(8)	2	0.93	GTX-2080Ti	500
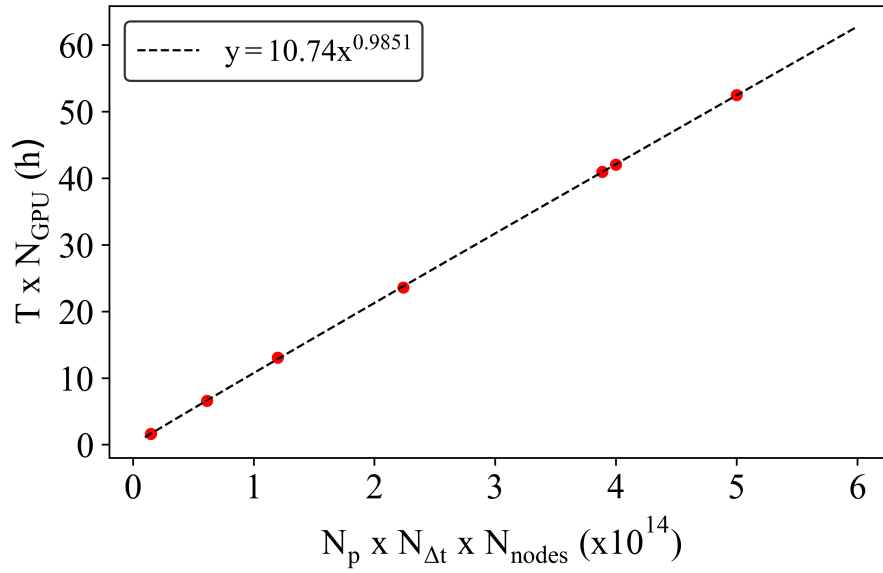
Figure C10. Simulation time as a function of simulation complexity. The black dotted line is a linear regression fit to the data (red solid circles).

have found that memory usage is largely dependent on the number of particles. In addition, large, complex field geometries also have an impact on memory usage. Memory requirements are expected to increase once the finite-element solver with unstructured grid for modeling user-defined electric fields is implemented.
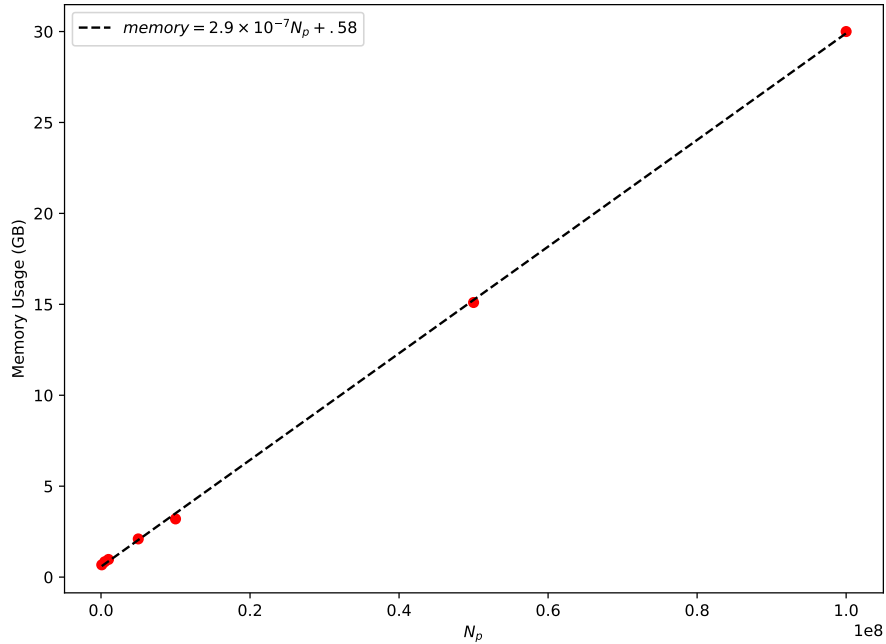
Figure C11. Memory usage in GB as a function of the number of particles, N_p. The black dotted line is a linear regression fit to the data (red solid circles).

Appendix D

Power Irradiated and Radiation Reaction Force

Accelerating charges radiate. Deflecting charged particles inherently involves a change in particle momentum as a function of time and thus imparting an acceleration to the incoming ion. With respect to magnetic deflection the particle changes direction only. The magnitude of momentum is constant in time and thus energy is conserved. For electric deflection both direction and magnitude of the momentum will change. Regardless of the method one expects that the deflected particle will radiate some of its energy away thus lowering its kinetic energy. The emission of photons imparts an impulse to the incoming ion. For instance, electrons beams accelerated in a synchrotron can undergo quite severe energy shift and broadening that must be corrected for [24]. The question here however is does it matter for calculation of particle trajectories in electromagnetic fields that may be used for active shielding studies.

The classical expression for radiated power was first worked out by Larmor [4]. For a particle of charge q, velocity $\vec{v}$, mass m and kinetic

energy E_k, the classical power P is given by

$$P = \frac{q^2}{6\pi\epsilon_0 c^3}|\dot{\vec{v}}|^2, \tag{D1}$$

where $\dot{\vec{v}}$ is the acceleration of the charged particle. Both electric and magnetic fields will result in acceleration (or deceleration) of a charged particle. To understand the relative contribution one can assess each field separately. A commonly used field configuration in previous active shielding studies is that of a constant magnetic field confined to a region around a habitable volume on the orders of meters in depth. Thus, one limiting case to consider is that where a charged particle of kinetic energy E_k undergoes uniform circular motion in a plane perpendicular to a uniform magnetic field of magnitude B_0. For this case the classical expression for the emitted power is

$$P = \frac{q^2}{4\pi\epsilon_0 c^3}|\dot{\vec{v}}|^2 = \frac{q^2}{4\pi\epsilon_0 c^3}a^2 = \frac{q^4 E_k}{3\pi\epsilon_0 (mc)^3}B_0^2. \tag{D2}$$

Due to the fact that charge particles enter the field region uniformly over 4π, depending on the field strength, some trajectories can actually be bent into the direction fo the habitable volume. To overcome this requires relatively high field strengths on the order of a few to 10 T such that the cyclotron radius of the largest particle energy to shield against is $\leq \ell_t$, the 'thickness' of the field region. Whereas other configurations with extended rather than confined magnetic fields, e.g. dipole and current loop fields, have much weaker field strengths extended over large distances, the constant, confined field case can be used to estimate a worst-case scenario for irradiated power.

The relativistic form of the power irradiated can be arrived at by considering the relativistic momentum as a Lorentz invariant. In covariant form the radiated power can be expressed as,

$$P = -\frac{q^2}{6\pi\epsilon_0 c^3}\left[\left(\frac{d\vec{p}}{d\tau}\right)^2 - \beta^2\left(\frac{dp}{d\tau}\right)^2\right] \tag{D3}$$

Assuming a constant magnetic field of magnitude $\vec{B} = B_0\hat{z}$, the power irradiated assuming relativistic particle speeds is then expressed as

$$P = \frac{q^2 E_k}{6\pi\epsilon_0 c^3}B_0^2\gamma^4. \tag{D4}$$

From a computational perspective this effect is somewhat challenging to model. The direction of emission ranges from isotropic at low energy to emission distributed in a narrow cone aligned with the particle velocity vector at higher energy. This cone becomes increasingly narrow with

increasing particle energy. In the relativistic limit ($\gamma \gg 1$) the angular distribution is [1]

$$\frac{dP}{d\Omega} = \frac{q^2}{6\pi\epsilon_0} \frac{|\dot{\vec{u}}|^2}{(1 - \beta cos\theta)^3} \left[1 - \frac{sin^2\theta cos^2\phi}{\gamma^2(1 - \beta cos\theta)^2} \right], \qquad \text{(D5)}$$

where θ is the angle between $\vec{\beta}$ (aligned along the z axis) and the unit vector $\hat{n}$ where $\vec{r} = r\hat{n}$ and ϕ is the polar angle in spherical coordinates. For $\gamma \ll 1$ the distribution is predominantly perpendicular to the direction of particle motion, so-called cyclotron emission. As γ increases the maximum in power irradiated shifts increasingly to the forward direction parallel to $\vec{u}$ (see Figure D1). At each step the magnetic field would need to be evaluated for the given field configuration selected. From this and the particle energy the cyclotron frequency ω_c is calculated. This sets the step size in time. The total energy loss due to radiation emitted by the particle is then calculated as $\Delta E_k = P\Delta t$. This total energy loss must then be translated to a change in particle speed but must be done by repartitioning each component of the velocity vector $\vec{u}$ according to the angular distribution $dP/d\Omega$, and, there is no a priori way to perform the partitioning.

In order to implement radiative energy loss, at each integration step the exact emission direction must be sampled from the angular distribution. This will give the power emitted per unit integration step. However, how this is partitioned into a change in velocity across all three directions can only be treated stochastically. This will undoubtedly increase computation time. Instead of tackling this problem directly, we look at order-of-magnitude estimates of the rate of energy loss in a worst-case scenario.

An estimate of the power radiated as well as the fractional energy loss for both non-relativistic and relativistic forms is shown in Figures D2 and D3. The field configuration and sample trajectory is shown in the inset of Figure D2. To estimate the fractional energy loss, it is assumed that at each magnetic field value, the thickness of the shell is adjusted to be $\ell_t = R_c$, the cyclotron radius. The fractional energy loss is calculated as

$$f_E = \frac{P\Delta t}{E_k}, \qquad \text{(D6)}$$

where P is calculated from either Eq. D1 (non-relativistic) or Eq. D4 (relativistic). Note for the electron, $\gamma \approx 197$ and for the 100 MeV proton $\gamma \approx 1.1$. Thus a 100 MeV electron is considered highly-relativistic whereas a proton at the same kinetic energy is at the lower end of where relativistic effects are important. For the case of 1000 MeV proton $\gamma \approx 2.1$. The number of photons emitted per second is quite high for electrons, and low for protons, as expected. However, the fractional energy loss over this confined field region is quite low for both electrons and protons at $E_k = 100$ MeV. For 1000 MeV protons, using the relativistic expression for irradiated power (Eq. D4), the fractional energy loss
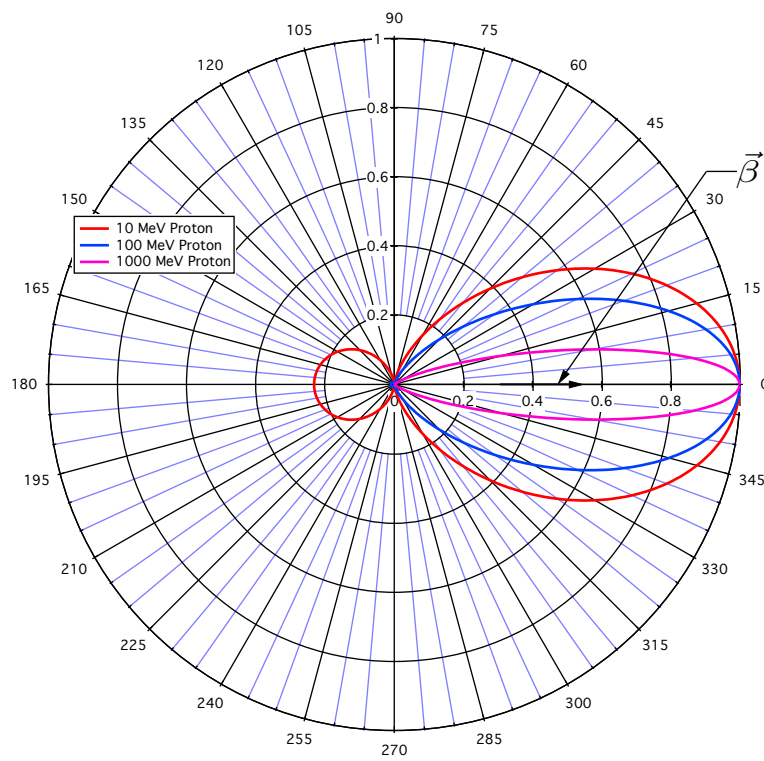
Figure D1. Angular distribution of irradiated power for 10 MeV, 100 MeV and 1000 MeV protons moving in a constant, azimuthally-oriented static magnetic field.

increases but is still much to low to be considered relevant for particle trajectory simulations using intense confined magnetic fields. For these reasons, radiative energy loss of charged particles has not been included into ASPP.
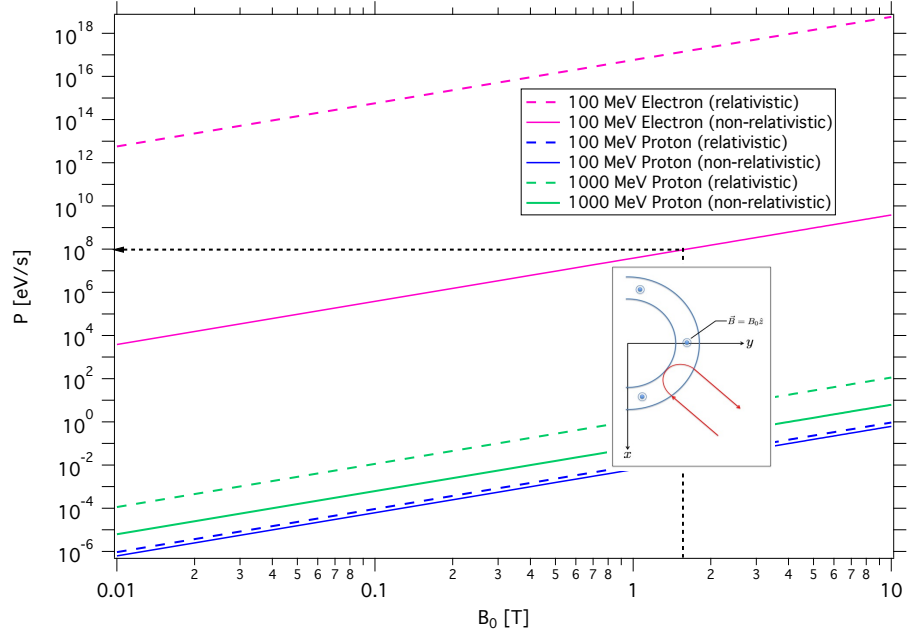
Figure D2. Theoretical power irradiated (Eqs. D2 and D4) from a single 100 MeV electron and proton traversing a uniform azimuthal magnetic field as a function of field strength, B_0. Estimates for both non-relativistic and relativistic effects are shown. The black dotted line is only to guide the eye on the difference between electrons and protons at fixed field strength. The inset shows the field and planar trajectory configuration.
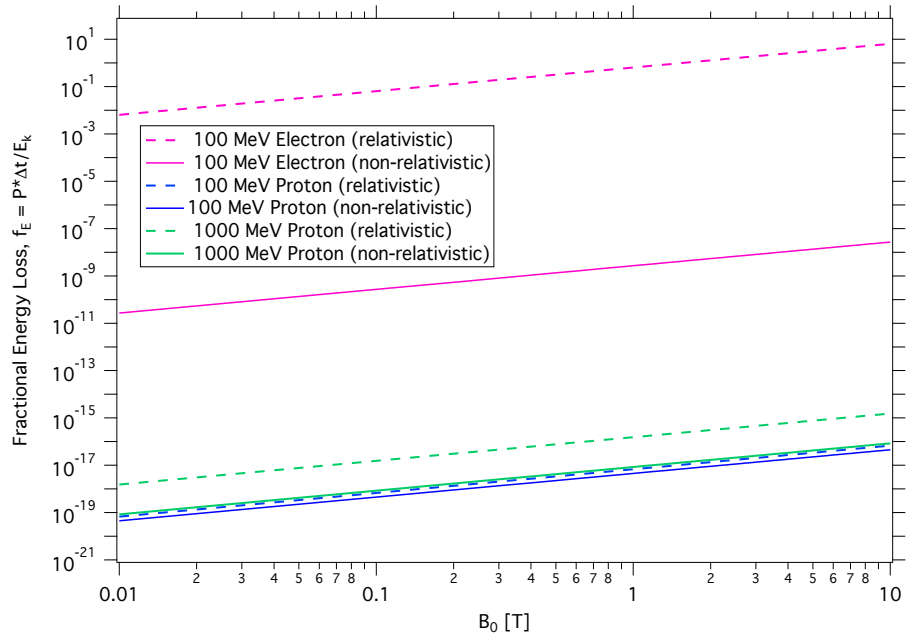
Figure D3.   Fractional energy loss for electrons and protons in a constant azimuthal magnetic field, calculated using Eqs. D2, D4 and D6. Note that for 100 MeV relativistic electrons, although the dotted line extends past a value of $f_E = 1$, this is not physically possible.

Appendix E

Review of Active Shielding Investigations

Radiation exposure represents a relatively large risk for long-duration human spaceflight. On the ground, the radiation protection strategy typically employed is 'time, distance and shielding', i.e. limit the time of exposure, increase the distance from the source and provide an acceptable degree of shielding. For space exploration missions, shielding is the only option.

There are two sources of radiation exposure for exploration, Solar Energetic Particle (SEP) events and Galactic Cosmic Rays (GCR). SEP events are relatively rare, random occurrences where the energetic proton flux can rise up to 4 orders of magnitude in intensity over background levels, extending out to energies on the order of 1 GeV/n. The GCR environment represents a relatively constant background exposure. GCR particle fluxes are relatively constant in time, fluctuating by roughly a factor of 2 over a period of 11.5 years, i.e. the solar cycle. Composed predominantly of protons, but including heavier ions up to iron, GCR ions are fully stripped of electrons and have energies that can extend to > 10 GeV/n. As a result they are highly penetrating through both spacecraft and the human body.

Traditional shielding methodology is to use 'passive' materials such as polyethylene and aluminum to block the incoming charged particles. Fundamentally this happens through the interaction between the charged particle and the material where the particle loses some or all of its energy to the material, and is either slowed down or completely comes to rest. For protons less than a few hundred MeV modest amounts of passive material can be used and in general mass can be reconfigured to minimize exposure without adding any substantial mass to the vehicle. Since the majority of the GCR spectrum is for energies higher than a few hundred MeV, requiring much greater material mass (meters of aluminum up at energies > 5 GeV/n), passive techniques are not effective as a shielding method.

All active methods shield regions in space by producing a force on charge particles that cause the particle trajectory to change. The physics of charged particle deflection by both electric and magnetic fields is well-understood. The difficulty has remained in how to build a shielding system that fits in the mass/power constraints of human space flight. Active shielding itself predates any known studies as our own planet has been providing an active shield for millennia. Earth's geomagnetic field is very effective in shielding both SEP and GCR particles resulting in lower exposures for Low Earth Orbit (LEO) mission such as the International Space Station (ISS).

An example of how the geomagnetic field provides shielding for the

ISS is shown in Figure E1. The bottom plot shows the integral proton flux as a function of time measured by the GOES satellite in geosynchronous orbit for the January, 2005 SEP event. Note the quite abrupt increase in flux of roughly 3 orders of magnitude above background over all proton energy channels. The top plot is the vertical magnetic rigidity as a function of time over the ISS trajectory covering the same time frame as the event. For all times where the vertical rigidity is above the horizontal black line (400 MV) ISS crews are effectively shielded from the event. It is only in the regions where the rigidity cutoff falls below 400 MV that there is penetration of energetic protons to ISS altitudes and increased exposure is expected. This translates to regions where ISS is near the maximum upper/lower latitudes and spans roughly 10 to 20 minutes of each orbit. This enables flight operations to manage exposure levels by directing crews on ISS to reside in better shielded locations during high/low-latitude time periods when SEP events take place. For flights beyond LEO, such as en route to Mars, crews and the vehicle will see the full extent of such a solar storm. Figure E2 shows example spectra for both SEP and GCR environments. SEP event data shown here is taken from the National Geophysical Data Center (NGDC) [19] and has been time integrated over the duration of the event. For direct comparison over the same time period, GCR daily proton flux has been integrated over the September 29th SEP duration.

Figure E2 shows the event-integrated differential proton flux for several historically large SEP events in 1989 and 2000. Also shown is the GCR proton flux time -integrated over the October 29, 1989 SEP. It is clear at lower energies (≤ 200 MeV) that SEP flux dominates over GCR proton flux. One might interpret this to mean that SEP exposure is the dominant concern. However, there are several features here that must be taken into account. First, SEP events this large are rare. The events shown, known as the '89' Halloween storms, are some of the largest in the historical record of the modern space age. The probability of having a single storm this large on any long duration mission of several years' duration is low. The probability of having more than one is essentially zero over the same mission duration. Second, the event was roughly 30 days in duration. Although the GCR flux over this 30-day period is substantially lower for energies < 400 MeV/n, integrating over a mission of several years in duration and including tissue weighting factors shows that GCR exposure represents a substantial risk to blood forming organs.

E.1   Summary of Key Publications in Active Shielding

Active spacecraft shielding has been a topic of study for approximately 55 years [25–52]. During this time, there has been over 80 publications (including non-peer reviewed study reports) on the subject, with the largest flurry of activity in the 1960's and early 1970's (see Figure E3).
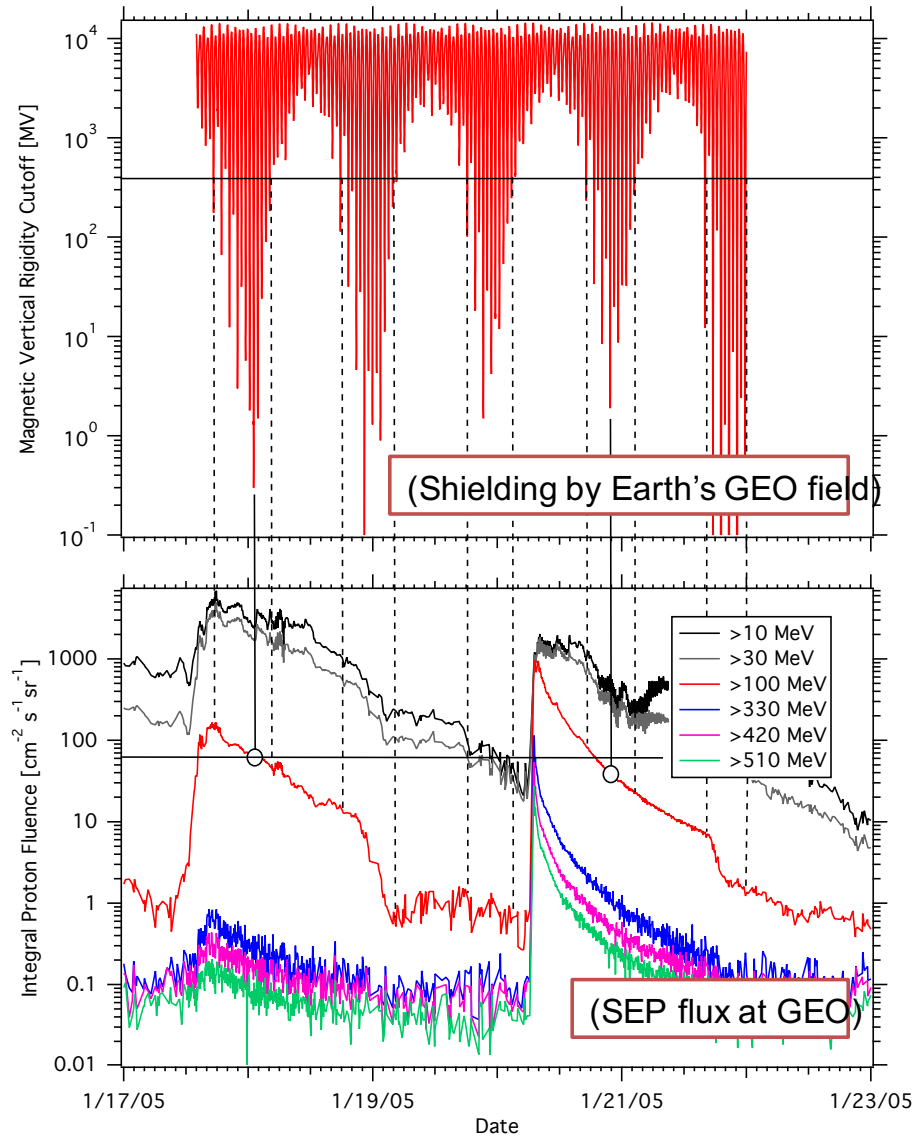
Figure E1. Time profile of January, 2005 SEP event and overlap with periods of high/low shielding by Earth's geomagnetic field characterized by the vertical rigidity cutoff. SEP data has been take from the NGDC [19].
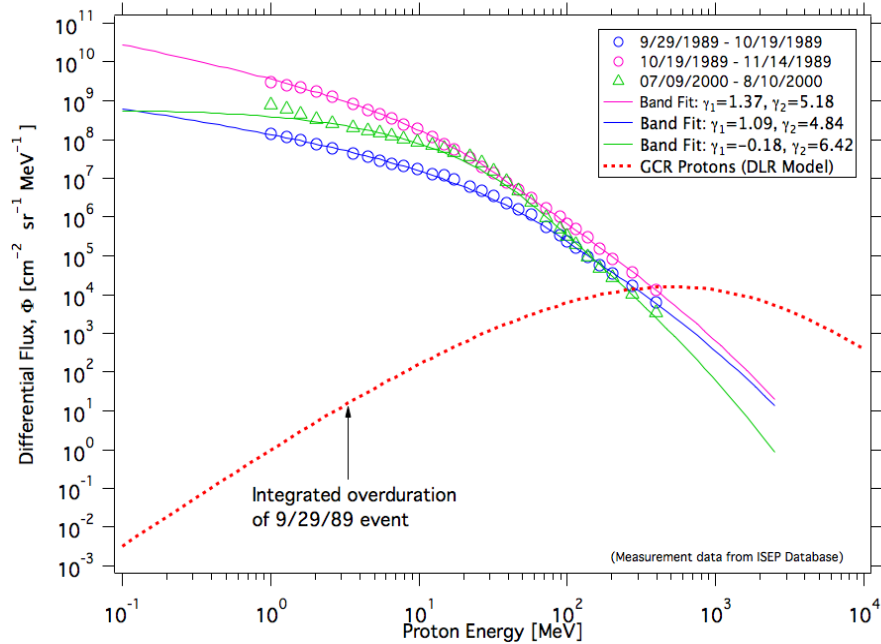
Figure E2. Differential SEP and GCR flux spectra. SEP spectra pulled from the NGDC [19]. The GCR spectrum is generated with the DLR model [20].

There was renewed interest in the early 1990's but has since been relatively stagnant apart from a handful of recent studies. Active shielding is a hard problem to solve and it appears that all efforts to date have come to the same conclusion – it's just too heavy, requires too much power and potentially dangerous for human exploration missions.

All of the work to date on the subject can be grouped into three primary classes:

1. Electrostatic shielding - the use of time-independent electric fields to deflect charged particles away from a spacecraft.

2. Magnetostatic shielding - the use of time-independent magnetic fields to deflect charged particles away from a spacecraft.

3. Plasma shielding - the use of electrostatic and/or magnetostatic fields to trap low-energy plasma at some distance from a spacecraft. The plasma potential contributes to particle deflection.

Electrostatic and magnetostatic methods use relatively large-strength electric or magnetic fields to deflect energetic charged particles away from a space craft. The required degree of dose reduction sets constraints on which part of the particle energy spectrum needs to be shielded against.

Figure E3. Histogram of the number of active shielding studies per year.

This, in turn, along with the isotropic nature of particle flux determines the field strengths and spatial distribution needed.

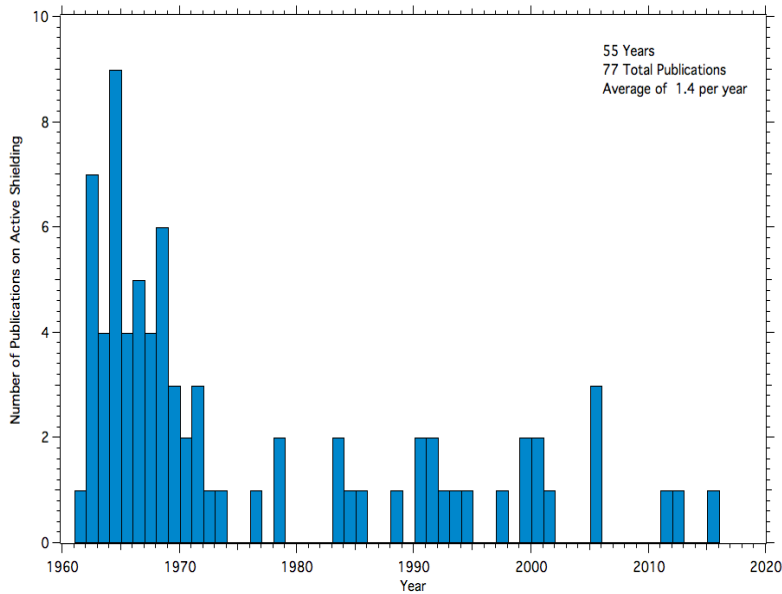The classes can be delineated further by considering the spatial extent of the field itself and how particle trajectories are actually changed. The majority of studies have focused on confined fields, magnetic and electric fields that have a non-zero value within a relatively small region around a spacecraft, and zero field outside of this region. There are only two studies in the literature that look at using unconfined fields, or fields that effectively terminate, in the ideal sense, at infinite distance from a spacecraft.

The difficulty with confined fields is that it inherently drives up power and in some cases mass of the shield support structure. The reason is that the field acts on a charged particle over a short distance. In this case, since the time in the field is short, the only way to effectively produce large changes in a particle's trajectory is by imparting a large impulse, $\Delta \vec{p}$.

Studies have looked at using large electric fields [25–35], intense, confined magnetic fields [36–44], unconfined magnetic fields generated with a current loop [45–49], charged plasmas confined to the near-vicinity of a spacecraft [26,27,50–52], finite solenoidal fields [53] and extended plasmas trapped over relatively large distances [54,55]. The problem has also been looked at through the lens of mathematical physics by application of Liouville's Theorem to understand the limits of particle dynamics in unconfined fields [56]. Several publications reviewing all past avenues

have also been produced [57–60]. Lastly, shielding of the artificial electron radiation belt was proposed using a magnetic dipole [61] . All of these approaches have one primary commonality: deflection and subsequent shielding is governed by the Lorentz force (see Eq. 30).

The environment to shield against defines the particle mass, m, velocity $\vec{v}$ (and γ) and charge state Ze. As a result, finding an effective shield means finding the most effective forms for electric and magnetic field $\vec{E}$ and $\vec{B}$ that have the largest impact on the change in particle ($d\vec{p}/dt$) momentum to keep it from impacting a habitable volume. The solution here is explicitly the integration of both sides of Eq. 30, and at closer inspection two distinct cases can be uncovered. The degree of shielding provided is directly related to how large the transfer of momentum given to a charged particle can be. The momentum is transferred directly from the field and thus the more time a particle is in the field the more time the field can act on it to change its trajectory. In confined fields the time spent is short since the field is confined to a relatively small volume around a space craft. Thus, in order to provide the required momentum transfer, the field has to be intense, requiring large power and likely large support structures. However, with unconfined fields, the time spent in the field is large and thus for an equivalent momentum transfer compared to the confined field case, the field intensity can be lower, requiring less power and less support structure.

E.1.1   Electrostatic Shielding

Early studies proposing electrostatic shielding considered charged concentric spheres. The difficulty with this approach is that the field vector only points in the radial direction emanating from the sphere. There is no tangential field component acting on any particle trajectory. In order to produce deflection voltages have to be on the order of the incoming particle energy (100's of MV).

Another approach is to look at a collection of charged spheres with the goal of having an extended electric field that more gently 'deflects' charged particles [9,10,12,13]. Tripathi looked at using a configuration of non-concentric spheres to provide more of a gradual deflection of SEP and GCR charged particles [9,10]. The arrangement is shown in Figure E4. The outer spheres (solid red circles) are held at a negative potential to keep electrons from penetrating the inner regions close to the habitable volume and neutralizing the positively charged inner spheres (solid green circles). Later, the authors proposed replacing the outer spheres by encasing the habitable volume with three toroidal rings (bottom left of Figure E4) as this provided better shielding. The units of distance on the (x,y,z) axes for this plot are meters. What was novel about this approach was the move away from having strong confined fields provide the shielding. Going to a long-distance field is what we propose may yield a feasible solution. In addition, it was recognized that

a spherical mesh may limit discharge due to ions hitting the conducting spheres/toroids. However, the voltages required are 10's to 100's of MV even with the spheres at 200 meters from the habitable volume. Moreover, the toroidal rings need 100's of MV. It is not clear from the proposal how such high voltages would be generated and there is no estimate of mass or deployment scenarios.
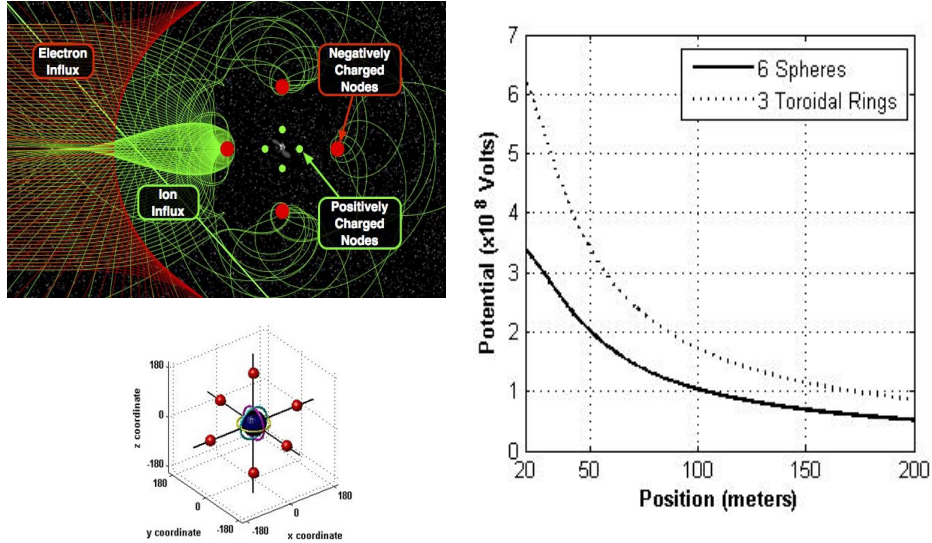

Figure E4.   Electrostatic shielding concept from Tripathi et al. [32]

The use of charged spheres has also been investigated for shielding surface habitats. Results by Buhler [34] showed that using roughly 12 charged spheres arranged in a configuration resembling the branches of a tree, and positioned 10's of meters off of a surface one could shield against energetic particles. However, the potentials required were again more than 100 MV and only showed effective shielding up to 100 MeV particle kinetic energy. Since 9 of the 12 spheres all had the same polarity potential this systems functions as a monopole where the numerous spheres simply increase the field coverage. It is thus not surprising that 100 MV was needed for 100 MeV particles.

Metzger et al. [35] investigated the use of an electrostatic quadrupole. The authors point out that the problem of active shielding possesses "several asymmetries which may be exploited in electrostatics to obtain nearly isotropic protection without radial symmetry in the fields, a concept that has been overlooked in previous studies". This same concept is to be leveraged in what we propose in the following sections. However, there is a key difference in what we propose that appears to be a way forward to a drastic lowering of the voltages required. In the work of Metzger, the arrangement of the spheres mimics an electrostatic linear quadrupole. Although this system acts as a focusing/defocusing

ion-optic element, the habitable volume is forced to be inside of the quadrupole arrangement. Thus the cluster of spheres surrounding this volume has to provide relatively strong deflection, which drives up the voltages required. It's not surprising then that similar to other electrostatic studies the authors find that they need 10's of MV on the spheres.

E.1.2 Magnetostatic Shielding

Active confined magnetic systems have been looked at by numerous authors. One study, funded by NASA through the Nasa Innovative Advanced Concepts (NIAC) program looked at 120 racetrack superconducting coils place around a cylindrical spacecraft [41]. A layout of the concept is shown in Figure E5. In this scenario the magnetic field is toroidal, i.e. wraps around the cylindrical vessel housing the crew. Outside of the coils and inside of the cylinder the field is zero. At first glance this is a relatively simple system. However, it is very heavy and potentially hazardous. The estimated field needed to shield up to 2 GeV/n protons is 9 T. Such high fields can only be produced by employing superconducting coils.
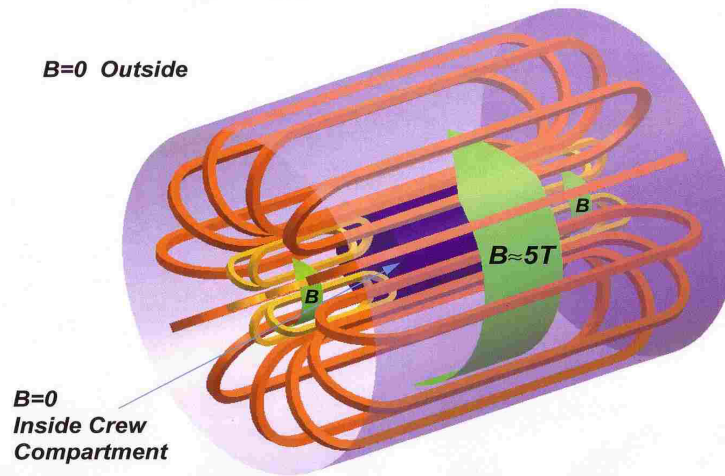
Figure E5. Concept of confined magnetic field proposed by Hoffman et al. [41].]. Estimate needing 9T field strengths to shield up to 2 GeV/n protons, totaling 400 to 1600 tons for a 3-year mission.

Although such fields can be generated, there are other difficulties when working with such high fields. The mechanical stresses produced within coils must be balanced and is proportional to the square of the field intensity, B^2. Support structures then tend to be large to counter the field-induced stress. In addition, the stored magnetic energy is proportional to B^2. With fields as high as 9 T the stored energy is on the order of GJ. Any shielding system of this sort must remain at cryogenic

temperatures while activated if aimed at shielding against GCR since it is the long-duration cumulative exposure of the crew that is of primary concern. This levies strict constraints on thermal system management to avoid a quench of the field. Since the probability of a quench will likely always remain non-zero, the system must also carry with it enough mass to absorb GJ of energy if a quench does occur. If not, the end result could be catastrophic. As noted by Hoffman et al., for the system studied, 5000 tons of additional mass is needed above the 400 to 1600 tons of mass just for the shield, not including the roughly 75 tons for a three-year mission. Thus any vehicle on a long-duration mission would need to carry along up to 6675 tons of additional mass just to shield the crew from energetic particles. For comparison the space shuttle Endeavor had an empty weight of 85.9 tons and a maximum payload weight of 27.6 tons. The entire orbiter, tank and boosters had a liftoff weight of 2204 tons (2000 metric tons).

Fast forward roughly 10 years. The NIAC Phase I and II of Westover et al. [42] took a similar approach to the study by Hoffman in terms of a toroidal magnetic field but looked at generating the field with a new solenoidal coil technique that would minimize coil mass. A concept of the configuration studied is shown in Figure E6. Each of the 6 coils is designed using a proprietary 'double helix' technique that uses higher order multipole terms from a skewed solenoidal field to produce field vectors that are diamagnetically oriented [43]. The superposition amongst the six coils produce a magnetic field vector that wraps azimuthally around a cylindrical crew volume at the center. Because the windings of the coil are able to be embedded in a soft fabric, these coils have the added benefit of being able to be expanded as they are being energized with current. The mass is also lower due to the low-weight fabric and minimized internal support structure. The six coils have a diameter of 8 m, a length of 20 m and a nominal field strength of 1 T. The total stored magnetic energy is 400 MJ, lower than the Hoffman study but still large enough to need a substantial amount of mass in the presence of a quench. The magnetic pressure between the coils was calculated to be 4 atm but could be as high as 25 atm with 2.5 T fields. The higher field would increase the stored energy into the GJ range. Leakage of the magnetic field into the crew volume required an additional coil to reduce the internal field from 0.25 T to < 0.02 T. The study looked at reducing the risk of a magnet quench. One of the issues was the reliability of quench detection which is still in question. No solution exists at this time.

The study also looked at the feasibility of using a solar shield to cool down the coils. Results showed that temperatures required (40 − 60 K) could not be reached with a solar shield alone. Cryogenic cooling will still be needed at the current time and requires 24 kW of power. This is however about a factor of 5 lower than the estimates of power for cooling using racetrack coils [18]. That said, it is estimated to take roughly 75 days (neglecting radiative heating) to cool the coils to a point that they
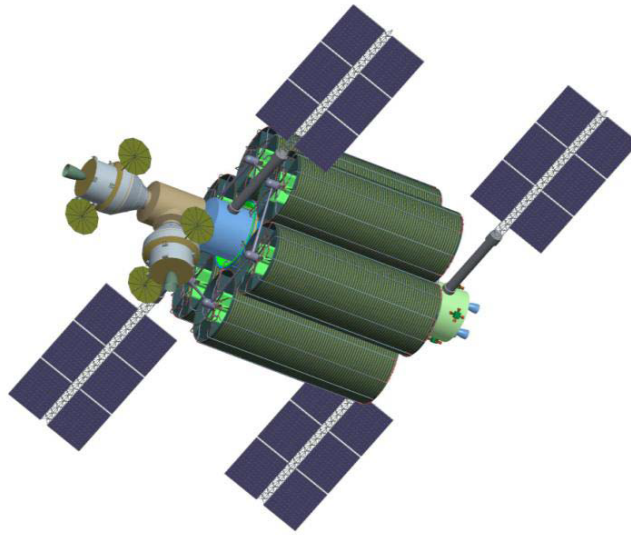
Figure E6. Concept of a confined magnetic shield from Westover et al., NIAC Phase I, 2012 [42]. Estimate needing 9 T field strengths, totaling 25 tons and 26 kW. However, study noted that although some GCR exposure reduction is obtained with this design "it is not significant enough to provide significantly greater mission duration".

will be superconducting.

The estimated mass of the 6 coils plus the compensator coil to reduce the field in the crew volume is about 48 tons (US). Cryocoolers contribute another 1.3 tons (US). The reduction in mass is significant over the Hoffmann study using racetrack coils (400 – 1600 tons). However, there is nothing in the estimates regarding mass needed to protect against a quench. Moreover, the Hoffmann study shielded up to 2 GeV/n protons and modest protection against heavier ions at the same energy per nucleon. Since the energy stored is proportional to the square of the field intensity a 9 T field (Hoffmann) requiring 5000 tons of mass for a quench could be reduced for a 1 T field, to first order, by a factor of 81. Thus, for the six coil configuration an additional 62 tons of mass would be needed, roughly the same mass as the entire six coil configuration. Although improvements in reduced mass over past studies have been found, the authors state that the reduction in GCR exposure "is not significant enough to provide significantly greater mission duration". It appears that the greatest reduction in exposure is in the range of a few hundred MeV/n ion energy with little shielding from 1 GeV/n ions.

The Space Radiation Superconducting Shield (SR2S) is a current European Space Agency (ESA) project developing a similar concept as studied by Hoffman [44]. The shield being investigated contains 120 10 m by 12.8 m racetrack coils with an estimated stored magnetic energy of 953

MJ at a maximum field strength of 4 T. The coil mass is roughly 36 tons, comparable to the study by Westover. Most of the same outstanding questions as past studies remain with the SR2S project. However, the work is ongoing and it is difficult at this time to assess what the outcome will ultimately be at project end.

Thermal shielding, operability, and quench protection still remain open questions. All confined magnetic shielding studies within the last decade arrive at similar conclusions: shield mass is $\geq$ 35 tons, field strengths required are more than several Tesla over several meters, stored magnetic energy is very large ($\geq$ 400 MJ) requiring active quench protection that has yet to be developed and large mass to protect in presence of a quench. Because the fields required for this field configuration are dictated by the physics governing particle deflection, there is no way to relax the high field/high current density issue and subsequent issues such as power, mass and quench risk.

Relatively large unconfined magnetic fields, generated with circular current loops, have been studied by several authors [45–49]. The advantage of using a loop of current is that far from the loop the field is essentially dipole in nature. Because the dipole moment of any given ideal loop can be readily calculated, trades can be done that can be directly compared to shielding by the geomagnetic field of the Earth. There is an additional attractiveness here in that studies of particle motion in a magnetic dipole field, investigated by Stormer to understand the Earth's aurora and now relatively well understood, show a well-defined 'exclusion zone', a region that energetic particles cannot access.

Cocks et al. [47–49], using Störmer theory and by assuming the asymptotic form of the magnetic field far from a current loop (dipole-like field) put forth an engineering analysis of what size coils and field strengths would be needed to shield from GCR ions. For instance, for an exclusion zone 5 m in size and a coil radius of 10 m the on-axis field strength is 2.25 T. Increasing the coil radius to 1 km, the field extent is quite large but its strength lowers by a factor of 1000 (0.02 T). Both cases, according to the authors will shield up to 500 MeV protons but the larger radius coils are at a much lower mass due to the reduced number of turns of wire needed. The major problem with this assessment is that the exclusion zone only shows up in a pure dipole field. The approximation of a dipole field for a current loop only holds for distances from the loop that are large compared with the loop (coil) radius. Shepherd et al. [45, 46] have shown that when considering the exact field of a current loop for large loops, on the order of km in radius, the exclusion zone at the center of the coil disappears. They show that current loops must be drastically smaller (10's of m rather than km) in order to produce an exclusion zone. At radii on the order of 50 m a region devoid of particles is shown but centered about the coil itself. As the radius increases further this region shrinks in size. Since the mass required to produce the same dipole moment increases as the coil radius decreases, this method does

not offer mass savings.

E.1.3   Plasma Shielding

This technique combines both electrostatic methods for generating large electric field gradients from trapped electrons with relatively weak magnetic fields [27] to contain the electron cloud and maintain the electric field. In this scenario, a toroidal shaped spacecraft (see Figure E7) is charged to a large positive potential, attracting electrons in the local environment around the spacecraft. A magnetic field, generated with superconducting coils traps the electron cloud at a distance from the spacecraft to avoid charge neutralization. The magnetic field is relatively weak with the study authors estimating needing a value of about 0.3 T, a value far below what is estimated to be needed for shielding up to several GeV/n protons with only a magnetic field (2-8 T).
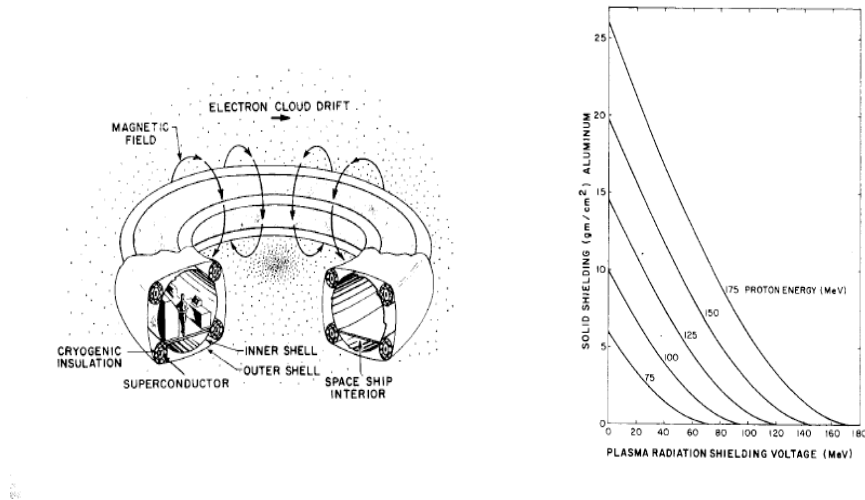

Figure E7.   Concept for using low-energy plasma for shielding along with results of Levy [26, 27].

Although a novel approach that clearly offers mass reduction over pure magnetic shielding, there are inherent difficulties in the approach, recognized by the study authors. For one, the shield functions primarily through effective trapping of the electron cloud, which is dynamic and not static. Electron loss rates are still somewhat poorly understood and thus controlling the dynamic nature of the electron cloud is questionable. Any power supply used to charge the space craft will need to sink this current and will cause the cloud to fluctuate about an equilibrium value on time scales that are still largely unknown. A recent study by Bamford et al. has shown similar concerns.

Another difficulty with this approach is the voltages needed. The right plot in Figure 7 shows a comparison of the voltage required to shield various proton energies and the representative passive (solid) material needed to produce the same shielding. For comparison, the 50th percentile shield thickness for a point at the center of the ISS US LAB module is about 10 g cm^{-2}. According to the study, a voltage of 80 MV is needed to shield up to 175 MeV/n protons. We know that the proton energies that need to be shielded against extend above 1 GeV/n, about a factor of 10 higher than the maximum energy of the study. Since the voltage needed for this method is roughly proportional to the particle energy, 1 GeV/n would require nearly 800 MV. The problem is that there is no currently available technology to generate 80 MV, let alone 800 MV in space.

E.1.4   Difficulty in Implementation

One of the key difficulties that has hindered progress in active shielding is that there is no way to actually ground test full scale spacecraft shields. There is no accelerator facility in the world that can accommodate a test article of that size, nor make ion beams wide enough to fully encompass the entire vehicle.

The focus on magnetic systems is likewise limiting. Most studies have focused on intense fields confined to a few meters and placed near a spacecraft. One would expect then that the most effective way to 'steer' particles away from a particular region in space would be to ensure that the magnetic field vector is always perpendicular to the surface normal of any surface bounding a habitable volume. Assuming a cylindrically-shaped crewed space craft, the field would be directed azimuthally around this cylinder. There aren't many ways to construct a magnetic field with this configuration and one is thus limited to a pure toroidal field or solenoids with diamagnetic oriented internal field vectors. In both cases generation is done via current-carrying coils. The problem however with all confined magnetic field configurations resides in the very nature of the Lorentz force itself. The force vector will always point in the same direction at each point in space about the azimuthal direction. This means that unless the magnetic field is large enough to make the cyclotron radius, $R_c(E_k, B_0) \leq \ell_t$, where (E_k, B_0), and ℓ_t are the particle kinetic energy, magnetic field magnitude and field thickness, respectively, half of the particles will be deflected away from the shielded region but half that may have missed this region will be deflected into it.

Many of the difficulties in implementation for magnetic systems remain today after 55 years of investigation. Although mass has been reduced in some cases, the composite system, along with quench mass is quite heavy. Numerous studies have made the case that active systems can be constructed with less mass than that required using only passive

shielding. This, however, is a faulty argument. The passive mass needed is so exorbitantly high that one could just as well set the mass at infinity and then state that any other system with lower mass is viable. The only viable metric here should be based on capability to get the system to Earth orbit within the launch and cost constraints that currently exist.

Magnetic systems continue to be plagued by high risk both to operation of the system itself and the entire mission. Concepts continue to be highly complex in terms of structural and thermal loading. It is still not clear whether or not each of these can be minimized further within the technology investment strategy that currently exists. Regardless, the ultimate limitation will likely reside with the development timeline of superconductor technology where there is no real driver related to use in human spaceflight.

The key to future development with electrostatic systems is either lowering the voltage required or developing the capability to generate 10's to 100's of MV. Work also needs to be done on understanding how to overcome issues with discharging and neutralization of charge by the background low-energy plasma.

Appendix F

References

1. J. D. Jackson, Classical Electrodynamics, Third Edition, John Wiley & Sons, Inc. (1999).

2. H. A. Lorentz, La theorie elecromagnetique de Maxwell et son application aux corps mouvemants, Arch. Neerl. Sci. Exactes Nat., **25** (1892).

3. H. A. Lorentz, Weiterbildung der Maxwellschen Theorie, Encykl. Mathe. Wiss., **2** (1904).

4. J. Larmor, On the Theory of the Magnetic Influence on Spectra; and On the Radiation from Moving Ions, Philos. Mag., **44** (1897).

5. M. Abraham, Prinzipien der Dynamik des Elektrons, Ann. Phys., **10** (1903).

6. M. Abraham, Theorie der Elektrizitat, Vol. II, (1905).

7. P. A. M. Dirac, Classical Theory of Radiating Electrons, Proc. R. Soc. London A, **167** (1938).

8. I. V. Sokolov, Renormalization of the Lorentz-Abraham-Dirac Equation for Radiation Reaction Force in Classical Electrodynamics, J. of Experimental and Theoretical Phys., **109** (2009).

9. R. F. O'Connell, The Equation of Motion of an Electron, Phys. Lett. A, **313** (2008).

10. F. Rohrlich, The Dynamics of a Charged Sphere and the Electron, Am. J. Phys., **65** (1997).

11. M. M. de Souza, The Lorentz-Dirac Equation and the Structures of Spacetime, Brazilian Journal of Physics, **28** (1998).

12. Cheng, K. S. and Zhang, J. L., General Radiation Formulae for a Relativistic Charged Particle Moving in Curved Magnetic Field Lines: The Synchrocurvature Radiation Mechanism, Astro. Jour., **463** (1996).

13. Verlet, L., Computer Experiments on Classical Fluids I: Thermodynamical. Properties of Lennard-Jones Molecules, Phys. Rev., **159** (1967).

14. Rappaport, D. C., *The Art of Molecular Dynamics Simulations*, Cambridge University Press, ISBN 978-0-521-82568-9 (2004).

15. Allen, M. D. and Tildesly, D. J., *Computer Simulation of Liquids*, Oxford University Press, (2017)

16. Stormer, C. J., *On the Trajectories of Electric Corpuscles in Space Under the Influence of Terrestrial Magnetism Applied to the Aurora Borealls and to Magnetic Disturbances*, A. Cammermeyer (1906).

17. Boris, J. P., Relativistic Plasma Simulation - Optimization of a Hybrid Code, Proceedings of the Fourth Conference on Numerical Simulation of Plasmas, US Government Printing Office (1970).

18. Rajon, D. A. and Bolch, W. E., Computer Methods and Programs in Biomedicine, **70**, 167–177 (2003) .

19. GOES Data Archive, National Geophysical Data Center, https://www.ngdc.noaa.gov/stp/satellite/goes/.

20. Matthiä, D., Berger, T., Mrigakshi, A. I., and Reitz, G., A Ready-to-Use Galactic Cosmic Ray Model, Adv. in Space Research, **51**, 3, ISSN 0273-1177 (2013).

21. McGuire, G. C., Using Computer Algebra to Investigate the Motion of an Electric Charge in Magnetic and Electric Dipole Fields, Am. J. Physics, **71**, 809-812, (2003).

22. Ozturk, M. Kaan, Trajectories of Charged Particles Trapped in Earth's Magnetic Field, Am. J. Physics, **80**, 420-428 (2012).

23. García-Farieta, J. E. and Hurtado, A., Simulation of Charged Particles in Earth's Magnetosphere: An Approach to the Van Allen Belts, Revista Mexicana de Física E, **65** (2019).

24. Synchrotron Radiation: Basics, Methods and Applications, Mobilio, S., Boscherini, F. and Meneghini, C. (Eds.), ISBN 978-3-642-55315-8, Springer-Verlag (2015).

25. Levy, R. H., Radiation Shielding of Space Vehicles by Means of Superconducting Coils, American Rocket Society, **31**, 1568-1570 (1961).

26. Levy, R. H. and French, F. W., Plasma Radiation Shield: Concept and Applications to Space Vehicles, J. Spacecraft, **5** (1967).

27. Levy, R. H. and Janes, G. S., Plasma Radiation Shielding, AIAA Journal, **2** (1964).

28. Tooper, R. F., Electrostatic Shielding Feasibility Study, ASD-TDR-63-194, U. S. Air Force, (1963).

29. Vogler, F. H., Analysis of an Electrostatic Shield for Space Vehicles, AIAA Journal, 2, 872-878, (1964).

30. Felten, J. E., Feasibility of Electrostatic Systems for Space Vehicle Radiation Shielding, J. of the Astro. Sci., 11, 16-22, 1964.

31. Tripathi, R. K., Wilson, J. W., and Youngquist,R. C., Electrostatic Active Radiation Shielding - Revisited, IEEEAC (2005).

32. Tripathi, R. K., Meeting the Grand Challenge of Protecting Astronaut's Health: Electrostatic Active Space Radiation Shielding for Deep Space Missions, NIAC Phase I Final Report (2011).

33. Kovalev, E. E., Molchanov, E. D., Pekhterev, Yu. G., Riabova, T. Y., Tikhomirov, B. I., and Khovanskaya, A. I., An Investigation of the Basic Characteristics of Electrostatic Shielding from Cosmic Radiations on the Artificial Earth Satellite Kosmos 605. I. Measurement Procedure and the Complex Scientific Aparatus, Cosmic Research 13(5), 687-692, (1976) (translated into English from Kosmicheskie Issledovaniya 13(5), 771-777).

34. C. R. Buhler, Analysis of a Lunar Base Electrostatic Radiation Shield Concept, NASA NIAC CP 04-01 Phase I Final Report (2005).

35. Metzger, P. T., Lane, J. E. and Youngquist, R. C., Asymmetric Electrostatic Radiation Shielding for Spacecraft, IEEE Aerospace Conference Proceedings (2004).

36. Bernert, R. E. and Stekly, Z. J. J., Magnetic Radiation Shielding Systems Analysis, Rept. PAM134, Avco- Everett Research Lab., Everett, Mass (1964).

37. Levine, S. H., and Lepper, R., Analog Studies of Magnetic Shields, AIAA Journal, **6** (1968).

38. Levine, S. H., and Lepper, R., An Active Radiation Shield for Cylindrically Shaped Vehicles, J. Spacecraft, **8** (1971).

39. L. W. Townsend, HZE Particle Shielding Using Confined Magnetic Fields, J. Spacecraft, **20** (1983).

40. Spillantini, P., Active Shielding for Long Duration Interplanetary Manned Missions, Adv. in Space Research, **45** (2010).

41. Hoffmann, J. A., Fisher, P. and Batishchev, O., Use of Superconducting Magnet Technology for Astronaut Radiation Protection, NASA NIAC CP 04-01 Phase I Final Report (2005).

42. Westover, S. C., et al., Magnet Architectures and Active Radiation Shielding Study (MAARSS), NIAC Phase I Final Report (2012).

43. Bruce, R. and Baudouy, B., Cryogenic Design of a Large Superconducting Magnet for Asto-Particle Shielding on Deep Space Travel Missions, Physics Procedia, 67, 264-269 (2015).

44. Musenich, R., Superconducting Magnetic Conceptual Design, SR2S Workshop, Turin, Italy (2014).

45. Shepherd, S. G. and Kress, B. T., Stormer Theory Applied to Magnetic Spacecraft Shielding, Space Weather (2007).

46. Shepherd, S. G. and Kress, B. T., Comment on "Applications for Deployed High Temperature Superconducting Coils in Spacecraft Engineering: A Review and Analysis" by J. C. Cocks et al., J. British Int. Soc..

47. Cocks, F. H., A Deployable High Temperature Superconducting Coil (DHTSC): A Novel Concept for Producing Magnetic Shields Against Both Solar Flare and Galactic Radiation During Manned Interplanetary Missions, J. British Int. Soc., **44** (1991).

48. Hilinski, E. J. and Cocks, F. H., Deployed High-Temperature Superconducting Coil Magnetic Shield, J. Spacecraft, **31** (1992).

49. Cocks, F. H. and Watkins, S., Magnetic Shielding of Interplanetary Spacecraft Against Solar Flare Radiation, NASA/USRA Advanced Design Program, Final Report (1993).

50. Bamford, R., et al., The Interaction of a Flowing Plasma with a Dipole Magnetic Field: Measurements and Modeling of a Diamagnetic Cavity Relevant to Spacecraft Protection, Plasma Phys. Control. Fusion, **50** (2008).

51. Gargate, L., et al., Hybrid Simulations of Mini-Magnetospheres in the Laboratory, Plasma Phys. Control. Fusion, **50** (2008).

52. Bamford, R. A. et al., An Exploration of the Effectiveness of Artificial Mini-Magnetospheres as a Potential Solar Storm Shelter for Long-Duration Human Space Missions, Acta Astronautica, 1-13 (2014).

53. Helgesen, J. Ol. and Spagnolo, F. A., The Motion of a Charged Particle in a Magnetic Field Due to a Finite Solenoid with Application to Solar Radiation Protection, AIAA 4th Aerospace Science Meeting (1966).

54. Winglee, R. M., Slough, J., Ziemba, T. and Goodson, A., Mini-Magnetospheric Plasma Propulsion: Tapping the Energy of the Solar Window for Spacecraft Propulsion, J. Geo. Res., 105 (2000).

55. Winglee, R. M., Advances in Magnetized Plasma Propulsion and Radiation Shielding, Proceedings of the 2004 NASA/DoD Conference on Evolution Hardware (2004).

56. Prescott, A. D., Urban, E. W. and Shelton, R. D., The Application of the Liouville Theorem to Magnetic Shielding Problems, Proceedings of the Second Symposium on Protection Against Radiations in Space, 189 (1964).

57. Sussingham, J. C., Watkins, S. A. and Cocks, F. H., Forty Years of Development of Active Systems for Radiation Protection of Spacecraft, J. Astro. Sci., **47**, 165-175 (1999).

58. Townsend, L. W., Overview of Active Methods for Shielding Spacecraft from Energetic Space Radiation, 1st International Workshop on Space Radiation Research and 11th Annual NASA Space Radiation Health Investigators' Workshop (2000).

59. Adams, J. H., Jr., Hathaway, D. H., Grugel, R. N., Watts, J. W., Parnell, T. A., Gregory, J. C. and Winglee, R. M., Revolutionary Concepts of Radiation Shielding for Human Exploration of Space, NASA/TM-2005-213688 (2005).

60. Parker, E. N., Travelers, Scientific American, (2006).

61. A. Bhattacharjie and I. Michael, Mass and Magnetic Dipole Shielding Against Electrons of the Artificial Radiation Belt, AIAA Journal, 2, 2198 (1964).

62. Advanced Magnet Lab, Melbourn, Fl.

63. Handbook of Mathematical Functions, edited by M. Abramowitz and I. A. Stegun, Dover Publishing, 1972.

64. Shultis, J. K. and Faw, R. E., Radiation Shielding, **18**, (2000).

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
01-05-2020	Technical Publication	

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Active Shielding Particle Pusher (ASPP): Charged-Particle Tracking Through Electromagnetic Fields	
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)	5d. PROJECT NUMBER
L. Stegeman	
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
NASA Johnson Space Center	L–

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
National Aeronautics and Space Administration Washington, DC 20546-0001	NASA
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
	NASA/TP–2020–5002408

12. DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified-Unlimited
Subject Category
Availability: NASA STI Program (757) 864-9658

13. SUPPLEMENTARY NOTES

An electronic version can be found at http://ntrs.nasa.gov.

14. ABSTRACT

15. SUBJECT TERMS

radiation shielding, active shielding, modeling, particle tracking, relativistic dynamics, particle tracking simulation

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			STI Information Desk (email: help@sti.nasa.gov)
U	U	U	UU		19b. TELEPHONE NUMBER *(Include area code)* (757) 864-9658