

Space Weather Modeling Framework: A new tool for the space science community

Gábor Tóth, Igor V. Sokolov, Tamas I. Gombosi, David R. Chesney, C. Robert Clauer, Darren L. De Zeeuw, Kenneth C. Hansen, Kevin J. Kane, Ward B. Manchester, Robert C. Oehmke, Kenneth G. Powell, Aaron J. Ridley, Ilia I. Roussev, Quentin F. Stout, and Ovsei Volberg

Center for Space Environment Modeling, University of Michigan, Ann Arbor, Michigan, USA

Richard A. Wolf, Stanislav Sazykin, Anthony Chan, and Bin Yu

Rice University, Houston, Texas, USA

József Kóta

University of Arizona, Tucson, Arizona, USA

Received 8 March 2004; revised 2 October 2005; accepted 10 October 2005; published 30 December 2005.

[1] The Space Weather Modeling Framework (SWMF) provides a high-performance flexible framework for physics-based space weather simulations, as well as for various space physics applications. The SWMF integrates numerical models of the Solar Corona, Eruptive Event Generator, Inner Heliosphere, Solar Energetic Particles, Global Magnetosphere, Inner Magnetosphere, Radiation Belt, Ionosphere Electrodynamics, and Upper Atmosphere into a high-performance coupled model. The components can be represented with alternative physics models, and any physically meaningful subset of the components can be used. The components are coupled to the control module via standardized interfaces, and an efficient parallel coupling toolkit is used for the pairwise coupling of the components. The execution and parallel layout of the components is controlled by the SWMF. Both sequential and concurrent execution models are supported. The SWMF enables simulations that were not possible with the individual physics models. Using reasonably high spatial and temporal resolutions in all of the coupled components, the SWMF runs significantly faster than real time on massively parallel supercomputers. This paper presents the design and implementation of the SWMF and some demonstrative tests. Future papers will describe validation (comparison of model results with measurements) and applications to challenging space weather events. The SWMF is publicly available to the scientific community for doing geophysical research. We also intend to expand the SWMF in collaboration with other model developers.

Citation: Tóth, G., et al. (2005), Space Weather Modeling Framework: A new tool for the space science community, *J. Geophys. Res.*, 110, A12226, doi:10.1029/2005JA011126.

1. Introduction

[2] The Sun-Earth system is a complex natural system of many different, interconnecting elements. The solar wind transfers significant mass, momentum, and energy to the magnetosphere, ionosphere, and upper atmosphere and dramatically affects the physical processes in each of these physical domains. The ability to simulate and eventually predict space weather phenomena is important for many applications, for instance, the success of spacecraft missions and the reliability of satellite communication equipment. In extreme cases, the magnetic storms

may have significant effects on the power grids used by millions of households.

[3] The various domains of the Sun-Earth system can be simulated with stand-alone models if simplifying assumptions are made about the interaction of a particular domain with the rest of the system. Sometimes the effects of the other domains can be taken into account by the use of satellite and ground-based measurements. In other cases, statistical and/or phenomenological models can be used. For the prediction of space weather events, however, we wish to use first-principles-based physics models for all of the involved domains, and these models must execute and be coupled in an efficient manner so that the simulation can run faster than real time.

[4] As an illustrative example of modeling multiple domains of the Sun-Earth system with a highly integrated numerical code, we describe the evolution of the space plasma simulation program BATS-R-US developed at the University of Michigan. Originally, BATS-R-US was designed as a very efficient, massively parallel MHD code for space physics applications [Powell *et al.*, 1999; Gombosi *et al.*, 2001]. It is based on a block adaptive Cartesian grid with block-based domain decomposition, and it employs the Message Passing Interface (MPI) standard for parallel execution. Later, the code was coupled to an ionosphere model [Ridley *et al.*, 2001], various upper atmosphere models [Ridley *et al.*, 2003], and an inner magnetosphere model [De Zeeuw *et al.*, 2004]. The physics models were coupled in a highly integrated manner resulting in a monolithic code, which makes it rather difficult to select an arbitrary subset of the various models, to replace one model with another one, to change the coupling schedules of the interacting models, and to run these models concurrently on parallel computers. Thus although BATS-R-US is successfully used for the global MHD simulation of space weather [Groth *et al.*, 2000; Gombosi *et al.*, 2004; Manchester *et al.*, 2005], monolithic programs have limitations.

[5] General frameworks are becoming more and more important in the numerical simulation of complex phenomena [see, e.g., Allen *et al.*, 2000; Reynders, 1996]. Just in the areas of geophysics and plasma physics, there are several frameworks under development [e.g., Hill *et al.*, 2004; Gurnis *et al.*, 2003; Luhmann *et al.*, 2004; Buis *et al.*, 2003; Leboeuf *et al.*, 2003]. A framework is a reusable system design, which aims at coupling together multiple, often independently developed, models via standardized interfaces. The framework makes the integration, extension, modification, and use of the coupled system easier than for a monolithic code or a collection of models coupled together in an ad hoc manner. Ideally, a framework can efficiently couple together state-of-the-art models, which are optimal in their respective domains, with minimal changes in the models.

[6] The Center for Space Environment Modeling (CSEM) at the University of Michigan and its collaborators have recently built a Space Weather Modeling Framework (SWMF). The SWMF is designed to couple the models of the various physics domains in a flexible yet efficient manner, which makes the prediction of space weather feasible on massively parallel computers. Each model has its own dependent variables, a mathematical model in the form of equations of evolution, and a numerical scheme with an appropriate grid structure and temporal discretization. The physics domains may overlap with each other or they can interact with each other through a boundary surface. The SWMF is able to incorporate models from the community and couple them with modest changes in the software of an individual model. In this paper we present the design and implementation of the SWMF and tests involving all the components.

[7] The paper is organized as follows: in section 2 we introduce the concept of physics domains and describe the domains used in the SWMF. The software design and architecture are presented in section 3. We explain how to transform a stand-alone physics model into a component of the framework and how the component interacts with the

core of the framework and with other components. Section 4 describes a SWMF simulation at a general level. We discuss the processor layouts of the components, sequential and concurrent execution models, handling of the input parameters, steady state and time-accurate modes of the SWMF, and the inner workings of the component couplings. The implemented components of the SWMF are briefly described in section 5, and we present test simulations involving all the components in section 6. Finally, we close the paper with conclusions and our plans for future applications and development.

2. Physics Domains and Their Couplings

[8] The current version of the SWMF includes nine physics domains ranging from the surface of the Sun to the surface of a planet (usually the Earth). The nine physics domains are the following: (1) Solar Corona (SC), (2) Eruptive Event Generator (EE), (3) Inner Heliosphere (IH), (4) Solar Energetic Particles (SP), (5) Global Magnetosphere (GM), (6) Inner Magnetosphere (IM), (7) Radiation Belt (RB), (8) Ionosphere Electrodynamics (IE), (9) Upper Atmosphere (UA). The physics domains depicted in Figure 1 correspond to the components of the framework. Each component can have multiple versions. A component version is based on a particular physics model, which is implemented by a particular physics module.

[9] We note here that in the current version of the SWMF the EE and SC domains are represented by one component. The reason is that there are a multitude of approaches by which an eruptive event can be generated, and each approach has different but rather tight coupling with the solar corona model. Since the SWMF is able to run with any subset of the components, it is possible to combine a number of domains into one component as long as only one of the combined domains is coupled to the rest of the components. For example, the EE, SC, IH, and SP domains could be represented by a single “super” IH component because the only coupling required from this subset of domains is the coupling with the global magnetosphere. Similarly, the IE and UA domains can be represented with a single “super” IE component, since (at least currently) the UA domain is only coupled directly to the IE domain.

[10] A new domain can be added to the framework in two different ways. One approach is to incorporate the new domain into an existing component. For example, one may add a reconnection model to the GM component to better represent the physics of the magnetic reconnection. This approach will minimize the development work, but it results in a less flexible framework, since it will be difficult to replace the reconnection model with alternatives, and it will not be possible to use the same reconnection model with alternative versions of the GM component. To truly extend the SWMF, one needs to define a new domain and add a new component to the framework. The SWMF has been extended with new physics domains several times during the development process, and the sequence of steps to take is described in the documentation.

[11] While using a monolithic model to represent a large subset of the components is a possibility, it is at odds with the purpose of the framework, which aims to provide access and control to the individual domains. Let us reiterate the

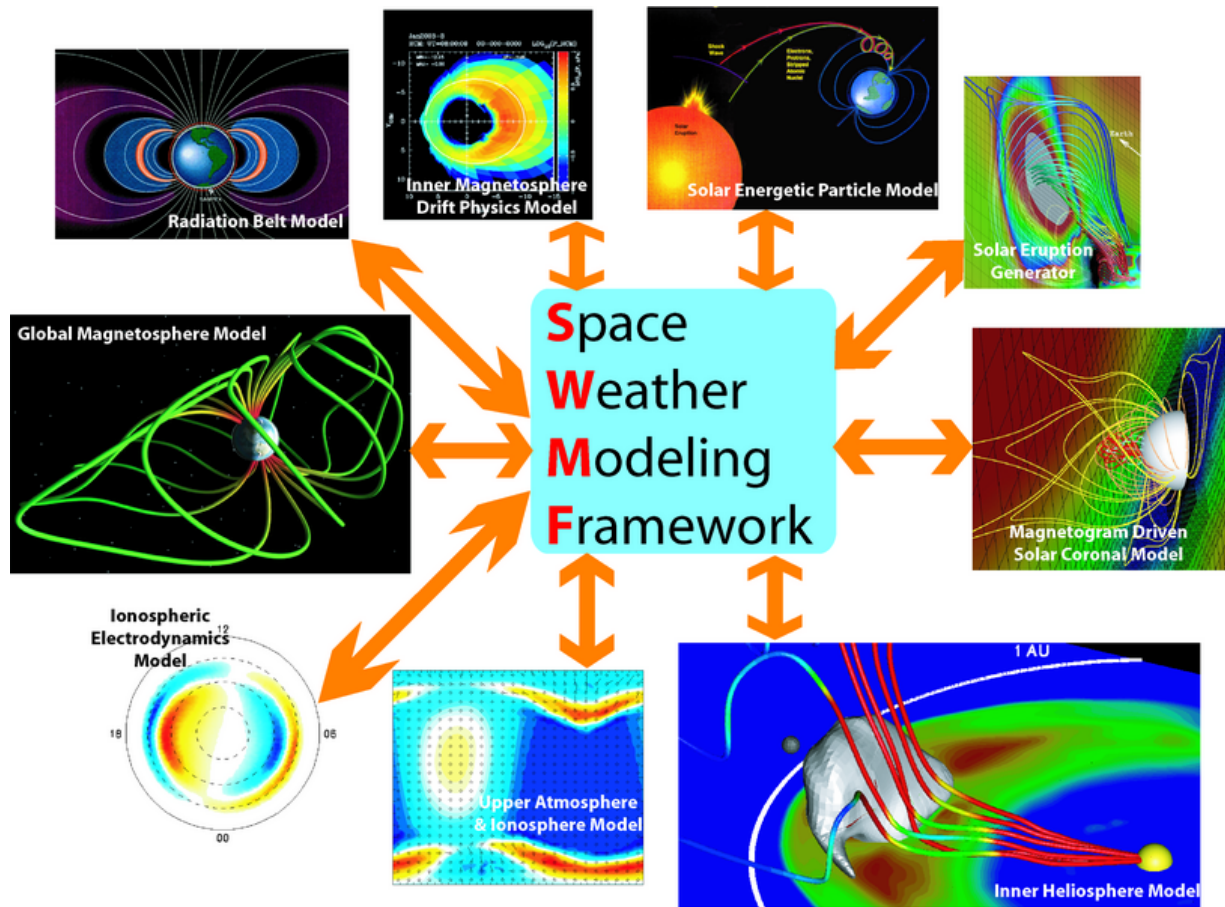


Figure 1. The Space Weather Modeling Framework (SWMF) and its nine physics domains.

advantages of using components instead of a single monolithic code. Although it is possible to approximate the physics from the surface of the Sun to the surface of the Earth with a single MHD code, there are many advantages of dividing up the work into the SC, IH, and GM components. Each component may use slightly different equations, different coordinate systems, different grids, and different numerical schemes, which are optimized for the particular physics domain. The components may have multiple versions or can be replaced with measurements. For example, the Global Magnetosphere may be driven by the measurements of the ACE satellite instead of the IH component. The multiple component approach results in more optimal representation of the physics domains and it allows a large number of combinations. All of these advantages make the framework more accurate, efficient, and versatile than a monolithic code.

[12] Below we briefly describe all nine physics domains, the typical coordinate systems, the equations to be solved, the boundary conditions, and the couplings with the other domains. A component is responsible for solving the dynamical equations in its domain, and it is also responsible for receiving and providing information as needed. The most computationally challenging couplings are described in more detail, since these present additional tasks to be accomplished by the components.

[13] Before going into the description of the individual models, we provide a brief outline of the essential

couplings of the domains when all the components of the SWMF are used: The Eruptive Event generator is coupled with the Solar Corona model only. The Solar Corona model drives the Inner Heliosphere model. Both the SC and the IH models provide input to the Solar Energetic Particles model. The Inner Heliosphere model drives the coupled system of the Global Magnetosphere, Inner Magnetosphere, Ionospheric Electrodynamics, and Upper Atmosphere models through the IH/GM interface. The Radiation Belt model receives information from the GM model only.

2.1. Solar Corona (SC)

[14] The Solar Corona domain extends from the low corona at $\approx 1 R_S$ (solar radius) to approximately $24 R_S$. The physics of this domain is well approximated with the equations of magnetohydrodynamics; however, additional source terms are required to take into account gravity, the heating, and acceleration of the solar wind [Groth *et al.*, 2000; Usmanov *et al.*, 2000]. Alternative models mimic the coronal heating by incorporating a variable adiabatic index [Wu *et al.*, 1999] or solve for one extra equation that describes the energy interchange between the solar wind plasma and the large-scale MHD turbulence [Roussev *et al.*, 2003b]. The SC component can be in an inertial (e.g., Heliographic Inertial (HGI)) frame or in a rotating (e.g., Heliographic Rotating (HGR)) frame. In a rotating frame the inertial forces must be included.

[15] The inner boundary of the SC component is driven by the density, pressure, velocity, and magnetic field defined just above the photosphere. The magnetic field may be obtained from synoptic magnetograms, or a simple dipole (possibly with a few higher-order terms) may be assumed. The boundary conditions for the temperature and mass density at the Sun may vary with longitude and latitude to achieve the most realistic solar wind near the Sun and at 1 AU. The velocity components at the inner boundary should maintain line-tying of the magnetic field. The flow at the outer boundary is usually superfast (faster than the fast magnetosonic speed of the plasma), so no information is propagating inward. Sometimes, however, when a coronal mass ejection (CME) passes the boundary, the solar wind speed may become subfast for short periods of time. During such periods, the SC component needs to receive the outer boundary condition from the Inner Heliosphere.

[16] The Solar Corona provides the plasma variables at the inner boundary of the Inner Heliosphere. The inner boundary of the IH component does not have to coincide with the outer boundary of the Solar Corona, i.e., the two domains are allowed to overlap. Such an overlap is actually numerically advantageous when the flow becomes subfast for a short time. The overlap can reduce reflections or other numerical artifacts at the inner boundary of the IH, which could otherwise arise for subfast flow when the SC and IH components use unaligned grids and different time steps. The Solar Corona also provides information to the Solar Energetic Particle domain: the geometry of one or multiple field lines and the plasma parameters along each field line are provided to the SP component.

2.2. Eruptive Event Generator (EE)

[17] The EE domain is embedded in the Solar Corona, and it is restricted to the region of the eruptive event, which is typically in the form of a coronal mass ejection (CME). To date, we lack good understanding of the actual physical processes by which a CME is initiated, and it is still an active field of research. One group of models [Forbes and Isenberg, 1991; Gibson and Low, 1998; Roussev et al., 2003a] assume that a magnetic flux rope exists prior to the eruption. Flux ropes may suddenly lose mechanical equilibrium and erupt due to foot-point motions [Wu et al., 2000], injection of magnetic helicity [Chen and Garren, 1993], or draining of heavy prominence material [Low, 2001]. Another group of models [Antiochos et al., 1999; Manchester, 2003; Roussev et al., 2004] relies on the existence of sheared magnetic arcades, which become unstable and erupt once some critical state is reached. Here a flux rope is formed by reconnection between the opposite polarity feet of the arcade during the eruption process.

[18] The EE component can be represented as a boundary condition for the SC component, or it can be a (nonlinear) perturbation of the SC solution. In short, the EE component interacts with the SC component only. Owing to the multitude of possibilities, the EE component is integrated into the SC component in the current implementation of the SWMF. Multiple EE versions are possible, but all the EE versions belong to one SC version only.

[19] We note that the eruptive event generator can be regarded as a boundary condition or as a perturbation of the initial conditions for the time accurate evolution of the solar

corona. The time accurate evolution itself follows the governing MHD equations with the source terms.

2.3. Inner Heliosphere (IH)

[20] The IH domain extends from around 20 solar radii all the way to the planet. It does not have to cover a spherical region, it may be rectangular and asymmetric with respect to the center of the Sun. The physics of this domain is well approximated with the equations of ideal MHD. The IH component is usually in an inertial (e.g., HGI) frame.

[21] The inner boundary conditions of the IH component are obtained from the SC component or measurements. The flow at the outer boundary of the IH component is always superfast (the interaction with the interstellar medium is outside of the IH). The Inner Heliosphere provides the same information to the SP component as the Solar Corona. The IH component also provides the outer boundaries for the SC component when the flow at the outer boundary of SC is not superfast. Finally, the Inner Heliosphere provides the upstream boundary conditions for the Global Magnetosphere. The IH and GM domains overlap: the upstream boundary of GM is typically at about 30 R_E (Earth radii) from the Earth toward the Sun, which is inside the IH domain.

2.4. Solar Energetic Particles (SP)

[22] The SP domain consists of one or more one dimensional field lines, which are assumed to advect with the plasma. The physics of this domain is responsible for the acceleration of the solar energetic particles along the field lines. There are various mathematical models that approximate this physical system. They include the effects of acceleration and spatial diffusion and can be averaged [Sokolov et al., 2004] or nonaveraged [Kóta and Jokipii, 1999; J. Kóta et al., Acceleration and Transport of Solar Energetic Particles in a Simulated CME Environment, submitted to *Astrophysical Journal*, 2005, hereinafter referred to as Kóta et al., submitted manuscript, 2005] with respect to pitch angle.

[23] The geometry of the field line and the plasma parameters along the field line are obtained from the SC and IH components. The boundary conditions can be zero particle flux at the ends of the field line(s). The SP component does not currently provide information to other components.

2.5. Global Magnetosphere (GM)

[24] The GM domain contains the bow shock, magnetopause, and magnetotail of the planet. The GM domain typically extends to about 30 R_E on the dayside, hundreds of R_E on the nightside, and 50 to 100 R_E in the directions orthogonal to the Sun-Earth line. The physics of this domain is approximated with the resistive MHD equations except near the planet, where it overlaps with the Inner Magnetosphere. The GM component typically uses Geocentric Solar Magnetic (GSM), Geocentric Solar Ecliptic (GSE), or possibly Solar Magnetic (SM) coordinate system.

[25] The upstream boundary conditions are obtained from the IH component or from satellite measurements. At the other outer boundaries one can usually assume zero gradient for the plasma variables, since these boundaries are far enough from the planet to have no significant effect on the dynamics near the planet. The inner boundary of the Global

Magnetosphere is at some distance from the center of the planet, usually at 1 to 3 planet radii. The inner boundary conditions are partially determined by the Ionosphere Electrodynamics, which provides the electric potential at the inner boundary of the GM. The potential is used to calculate the electric field and the corresponding plasma velocities, which are used as the inner boundary condition for the GM. The GM component also receives pressure and possibly density corrections from the Inner Magnetosphere along the closed magnetic field lines (field lines connected to the planet at both ends). These are used to “nudge” the MHD solution toward the more accurate inner magnetosphere values [De Zeeuw *et al.*, 2004].

[26] The GM component provides the field-aligned currents to the IE component. These currents are mapped from the GM down to the ionosphere along the magnetic field lines. The Global Magnetosphere provides the Inner Magnetosphere with the field line volume, average density, and pressure along closed field lines. Depending on the needs of the IM component, the GM could also provide the geometry of the closed field lines and the distribution of plasma parameters along field lines.

2.6. Inner Magnetosphere (IM)

[27] The IM domain consists of the closed field line region around the planet. This component solves equations describing the motion of keV-energy ions and electrons. Kinetic effects are important for these particles, and several types of theoretical models have been developed to describe them. At least five different groups have developed codes that calculate the distribution function of the ring current ions and associated electrons given an inputted electric and magnetic field distribution (see review by *Ebihara and Ejiri* [2002]). The Rice Convection Model (RCM) [e.g., *Wolf et al.*, 1982; *Toffoletto et al.*, 2003] computes field-aligned currents and ionospheric potentials self-consistently but still requires an inputted magnetic field and assumes that the particles have an isotropic pitch-angle distribution (consistent with MHD). Models developed by *Fok et al.* [2001], *Liemohn et al.* [2004], and *Ridley and Liemohn* [2002] compute full pitch-angle distributions as well as field-aligned currents and ionospheric equipotentials. A different approach uses test-particle Monte Carlo models [*Chen et al.*, 2003; *Ebihara et al.*, 2004]. The IM component typically uses Solar Magnetic (SM) coordinates.

[28] The Inner Magnetosphere obtains the geometrical and plasma information about the closed field lines from the Global Magnetosphere. It also obtains the electric potential solution from the Ionosphere Electrodynamics. The IM component provides the density and pressure corrections along the closed field lines to the GM component. The IM may also provide field-aligned currents along the closed magnetic field lines to the IE component (but this is not done in the current implementation of the SWMF).

2.7. Radiation Belt (RB)

[29] The RB spatial domain is coincident with that of the Inner Magnetosphere component. This component solves equations for the relativistic electron distribution near the Earth, which are responsible for some of the most detrimental space weather effects. Gradient and curvature drift

dominate the motion of these particles around the Earth, and the kinetic equation is sometimes drift-shell averaged as well as gyration and bounce averaged. Diffusion is the primary transport mechanism left in the equation. The physics of this domain can be solved with the same two techniques mentioned for the Inner Magnetosphere, that is, numerical discretization of the kinetic equation [*Beutier and Boscher*, 1995; *Shprits and Thorne*, 2004] or test particle tracking [*Elkington et al.*, 1999]. The RB component typically uses Solar Magnetic (SM) coordinates or simply equatorial plane radial distance.

[30] The Radiation Belt receives similar information from the Global Magnetosphere, as does the Inner Magnetosphere. The RB component does not provide information to the other components.

2.8. Ionosphere Electrodynamics (IE)

[31] The IE domain is a two-dimensional height-integrated spherical surface at a nominal ionospheric altitude (at around 110 km for the Earth). There are several mathematical models that can describe this domain: empirical models such as the *Weimer* [1996] electric potential maps and the *Fuller-Rowell and Evans* [1987] particle precipitation and auroral conductance maps; the assimilative mapping of ionospheric electrodynamics [*Richmond and Kamide*, 1988]; and the height averaged electric potential solver, which uses the field-aligned currents to calculate particle precipitation and conductances [*Ridley et al.*, 2004; *Ridley and Liemohn*, 2002]. In the current version of the SWMF, the IE component is a potential solver, but there is nothing in the design that would exclude the incorporation of other types of IE models. Usually, the IE component uses the Solar Magnetic (SM) coordinates.

[32] The Ionosphere Electrodynamics obtains the field-aligned currents from the Global Magnetosphere and Upper Atmosphere, which is used to generate an auroral precipitation pattern. The UA component also provides IE with the Hall and Pedersen conductivities. In case the UA component is not used, the auroral pattern and the solar illumination are used to generate Hall and Pedersen conductances. This is done through the use of the *Robinson et al.* [1987] and *Moen and Brekke* [1993] formulation, which takes the average and total electron energy flux and converts them to Hall and Pedersen conductances based on a simple formula. The IE component provides the electric potential to the GM, IM, and UA components. In addition, it provides the particle precipitation to the UA component.

2.9. Upper Atmosphere (UA)

[33] The UA domain includes the thermosphere and the ionosphere and it extends from around 90 km to about 600 km altitude for the Earth. The physics of the Upper Atmosphere is rather complicated. It can be approximated with the equations of multispecies hydrodynamics including viscosity, thermal conduction, chemical reactions, ion-neutral friction, coupling of the ions to the electric field, source terms due to solar radiation, etc. In such a complex system there are many possible choices even at the level of the mathematical model. For example, one can approximate the system with the assumption of hydrostatic equi-

librium [Richmond *et al.*, 1992] or use a compressible hydrodynamic description [Ridley *et al.*, 2005]. The UA component is typically in a planet-centric rotating frame, i.e., the Geocentric (GEO) coordinate system for the Earth.

[34] The lower and upper boundaries of the UA domain are approximated with physically motivated boundary conditions. The Upper Atmosphere obtains the electric potential along the magnetic field lines and the particle precipitation from the Ionosphere Electrodynamics. The gradient of the potential provides the electric field, which is used to drive the ion motion, while the auroral precipitation is used to calculate ionization rates. The UA component provides field-aligned currents and the Hall and Pedersen conductivities to the IE component. The conductivities are calculated from the electron density and integrated along field lines.

2.10. Coupling the Inner Magnetosphere and the Global Magnetosphere

[35] The IM to GM coupling is the most challenging computationally. The GM component needs to know where each of its three-dimensional (3-D) grid points are mapped onto the IM grid along the closed magnetic field lines in order to apply the pressure and density corrections. This means that field lines must be traced from possibly millions of grid points. In addition, the magnetic field information is typically distributed over many processors of the GM component. Since the GM grid structure and the magnetic field is inherently known by the GM component, it is the responsibility of the GM component to find the mapping of its 3-D grid along the closed field lines. For our implementation of the GM component, we have developed a highly parallel field line tracing method, which uses interpolations at the block boundaries of the grid. This algorithm can accomplish this task in a few seconds (G. Tóth *et al.*, Parallel field line and streamline tracing algorithms for space physics applications, manuscript in preparation, 2005, hereinafter referred to as Tóth *et al.*, manuscript in preparation, 2005).

[36] The GM to IM coupling is also challenging computationally. The IM needs the magnetic field line flux tube volumes and the average density and pressure in the flux tubes connected to its 2-D spherical grid points. This requires an accurate integration along many (thousands of) magnetic field lines on the distributed grid of the GM component. We have developed an efficient parallel algorithm (Tóth *et al.*, manuscript in preparation, 2005), which can accurately (with no extra interpolations) trace and integrate along the thousands of field lines in a fraction of a second of CPU time. The framework provides a library, which takes care of the information exchange and the collection of data among the processors of GM, but the GM component is responsible for the tracing and integration along field lines within the subdomain corresponding to one GM processor.

[37] Currently, the IM component is implemented by the Rice Convection Model, which assumes that the closed field lines map to symmetric points on the Northern and Southern Hemispheres in the SM coordinate system. This is approximately true in most cases. Given this approximation, the mapping points on the Northern Hemisphere are used in the GM-IM coupling, although

the field line tracing algorithms calculate the southern mapping points as well.

2.11. Coupling the Solar Corona and the Inner Heliosphere to the Solar Energetic Particles

[38] The SP component needs the geometry of one or more magnetic field lines, and it also needs the plasma parameters along these lines. This is not a computationally intensive procedure due to the small number of field lines. On the other hand it is an algorithmically nontrivial problem, especially when the SP component uses a Lagrangian grid. In our implementation of the SWMF, the field line is traced through the SC and IH components by the core of the framework, and the components only need to provide the plasma variables for the moving grid points when requested.

2.12. Coupling of the Inner Magnetosphere, Ionosphere Electrodynamics, and Upper Atmosphere Components

[39] The coupling of the IM, IE, and UA components is challenging physically rather than computationally. This subsection provides a brief description of the current coupling of these components in the SWMF.

[40] In the magnetospheric system, the inner magnetosphere (IM) and the ionosphere (IE and UA) are linked together in three main ways: (1) coupling of electric fields and currents, (2) energetic particle precipitation from the IM to the UA, and (3) outflow of low energy particles from the UA to the IM. In the SWMF, all electrodynamic coupling occurs through the Ionosphere Electrodynamics component; thus the IM coupling to UA occurs indirectly through the IE. The IE component uses field-aligned currents from the GM and the conductances from the UA to calculate the ionospheric potential, which is then provided to the IM. The conductances are also passed on from the UA to the IM by the IE component. Since the RCM model is symmetric, the IM is either coupled with the Northern or the Southern Hemisphere (or possibly an average of these) of the IE component.

[41] The magnetospheric currents that drive the ionospheric potential are mapped from the Global Magnetospheric domain only, instead of from the GM and IM components. This is somewhat problematic in that the IM component could generate current inside of the GM's inner boundary, which would therefore not be mapped down to the ionosphere. This causes some physics of the low-latitude dynamics to be simplified. This simplification is done because it is nontrivial to automatically blend together the GM region-1 currents and the region-2 currents. The diffusion is quite different in the IM and GM components, and there could therefore be large differences in the strength of the mapped ionospheric currents between the two models. Taking only the GM currents insures that the region-1 and region-2 currents are calculated in the same way. The problem of missing currents can be rectified by moving the inner boundary of the model closer to the Earth (it is typically set to $2.5 R_E$).

[42] The particle coupling is also a difficult issue. The RCM does not resolve pitch angles, so it can not truly specify which particles are in the loss cone and should precipitate into the ionosphere, although it can estimate the precipitation within the closed field line region with the

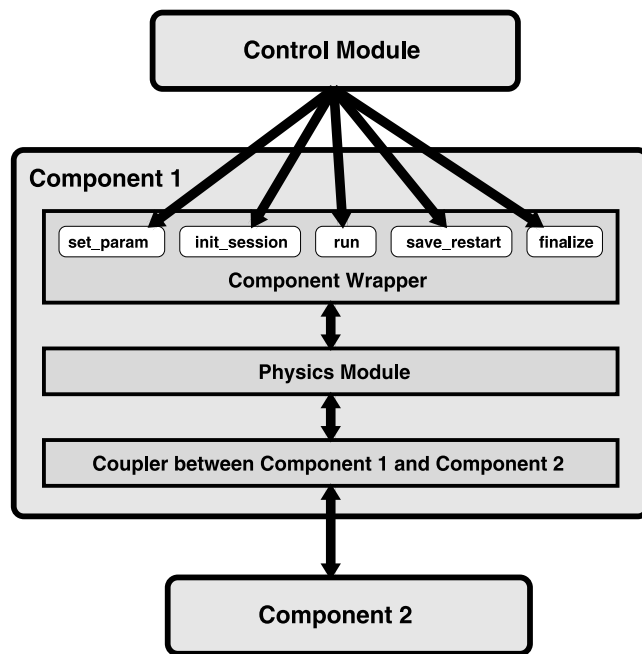


Figure 2. The structure of a physics component.

assumption of strong pitch-angle scattering. The SWMF currently uses an empirical relationship in the IE component to estimate the precipitation based on the field-aligned currents obtained from the GM. This approach has the advantage that the precipitation is calculated the same way inside and outside the closed field line region. The ionospheric outflow is not included in the framework at this time.

[43] The IE-GM and the IE-UA couplings all use mappings along magnetic field lines. The IE-GM mapping uses an analytic formula assuming a tilted dipole field. The UA-IE coupling uses numerical integration along field lines either based on a dipole or a more realistic field. The transformations between the coordinate systems (GSM for GM, SM for IE, and GEO for UA) and the mapping along magnetic dipole field lines are done with the library functions of the SWMF Infrastructure Layer (see the next section).

3. Architecture of the SWMF

[44] The SWMF provides a flexible and extensible software architecture for multicomponent physics-based space-weather simulations, as well as for various space physics applications. The main SWMF design goals are to (1) incorporate computational physics modules with only modest modification, (2) achieve good parallel performance, and (3) make the SWMF as versatile as possible. These design goals are orthogonal to each other, and the actual design represents a tradeoff. One can minimize the changes in the physics modules at the expense of performance and flexibility, or one may maximize software reuse and integration at the expense of the other design goals. The SWMF design focuses on achieving good performance, while the versatility of the SWMF and the minimization of code change are taken into consideration as much as possible.

The initial investment into the integration of the physics modules pays off in the flexibility, usability, and the performance of the SWMF. In our experience, it takes about 2 weeks of work by one or two people to integrate a new physics module into the framework. Typically, a few hundred lines of new code need to be written, but this strongly depends on the number and complexity of the couplings. The integration includes verification that the component still functions correctly, and it sends and receives data from other components as designed. Fine tuning the couplings and improving the physics of the combined framework may take more time, but this is related to the complexity of the physics rather than the software development.

[45] One of the most important features of the SWMF is that it can incorporate different computational physics modules to model different domains of the Sun-Earth system. Each module for a particular domain can be replaced with alternatives, and one can use only a subset of the modules if desired.

3.1. Design Challenges

[46] There are several problems known a priori, which need to be solved so that the heterogeneous computational models of the different domains of the Sun-Earth system can properly interoperate. An incomplete list of these problems is as follows: (1) There are serial and parallel models; (2) An individual model is usually developed for stand-alone execution; (3) Input/output operations do not take into account potential conflicts with other models; (4) The majority of models are not written in object oriented style, which means that data and procedure name conflicts can easily occur; (5) Models often do not have checkpoint and restart capabilities; (6) The efficient coupling of any arbitrary pair of parallel applications, each of them having its own grid structure and data decomposition, is not easily achieved. Some of the problems, like name space and I/O unit conflicts, can be avoided if the physics components are compiled into individual executable programs. However, coupling multiple executables via the Message Passing Interface (MPI) is not generally supported on our target platforms. For this reason the SWMF has been designed to compile and run as a single executable, and we have developed automated solutions to resolve procedure name and I/O unit number conflicts. The design does not exclude multiple executables, in fact the SWMF can be compiled to include only a subset of the components. In principle one could run different configurations of the SWMF together, but this would be more complicated than using a single executable.

[47] The interaction of the (replaceable) physics modules can be achieved by introducing standardized interfaces between them. The physics module and its interface forms a component of the framework. The component is created from a physics module by making some minimal changes in the module and by adding two relatively small units of code: (1) a wrapper, which provides the standard interface to control the physics module, and (2) a coupling interface, to perform the data exchange other components. Both the wrapper and the coupling interface are component interfaces: the wrapper is an interface with the high-level Control Module (CON) of the framework, and the coupling interface is an interface with the other components. As shown in Figure 2, the wrapper interface functions have standard

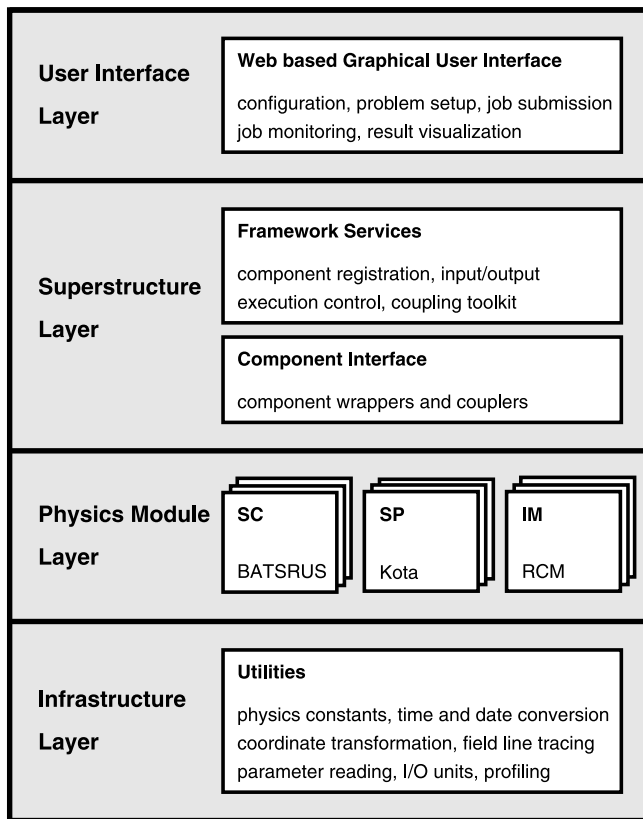


Figure 3. The layered hierarchy of the SWMF.

names, which makes swapping between various versions of a component possible. The data exchange between the components includes all the necessary unit conversions, mapping, coordinate transformation, and interpolations. Both the wrapper and the coupling interface are constructed from the building blocks provided by the framework.

3.2. Requirements for Physics Modules

[48] The physics modules must comply with a minimum set of requirements before they are transformed into a component. (1) The parallelization mechanism (if any) should employ the MPI standard and the physics module should be able to use an arbitrary MPI communicator; (2) The module needs to be compiled as a library that could be linked to another executable; (3) The module should read input from and write output to files that are in a subdirectory unique for the component; (4) A module should be implemented in Fortran 77 and/or Fortran 90; (5) The module should be portable to a specific combination of platforms and compilers, which include Linux clusters and NASA supercomputers; (6) The stand-alone module must successfully run a model test suite provided by the model developer on all the required platform/compiler combinations; (7) A module should be supplied with appropriate documentation. The first three requirements directly address the problems listed in the previous subsection, while the rest make the integration work into the SWMF possible and ensure the portability of the SWMF. Since the core of the SWMF is written in Fortran 90, there is no need to solve the issues that arise when a software is written in multiple languages.

3.3. Component Wrapper and Coupler

[49] The SWMF requirements for a component are defined in terms of a set of methods implemented in the component interface, i.e., the wrapper and the couplers. The methods enable the component to perform the following tasks: (1) provide version name and number to the Control Module; (2) accept parameters for parallel configuration; (3) accept and check input parameters obtained from the Control Module; (4) provide grid description to the Control Module; (5) initialize for session execution and read restart files if necessary; (6) execute one time step that cannot exceed a specified simulation time; (7) receive and provide data to another component via an appropriate coupler; (8) write its state into a restart file when requested; (9) finalize at the end of the execution. The structure of a component and its interaction with the Control Module (CON) and another component are illustrated in Figure 2.

3.4. Layered Architecture

[50] The framework's layered architecture is shown in Figure 3. The Superstructure Layer, Physics Module Layer, and Infrastructure Layer constitute the "sandwich-like" architecture similar to that of the Earth System Modeling Framework (ESMF) [Hill *et al.*, 2004].

[51] The Superstructure Layer contains the Component Interface (wrappers and couplers) and the Framework Services. The latter consists of software units (classes) that implement component registration, setting up the parallel layout, reading and distributing the input parameters, control of component execution and coupling, and the SWMF parallel coupling toolkit, which can be used by the component couplers.

[52] The Physics Module Layer contains the physics modules, which have been integrated into the SWMF, and provide appropriate entry points for the Component Interface. Each physics module may have multiple versions. The physics module may contain additional code for stand alone execution.

[53] The Infrastructure consists of utilities, which define physics constants, transformation between different coordinate systems, time conversion routines, time profiling routines, and other lower level routines. The Infrastructure can be used by the physics modules as well as by the Superstructure.

[54] The SWMF will also contain a web-based Graphical User Interface (GUI). The GUI will provide a user-friendly interface via a web browser, which will enable the user to select components, compile the SWMF, define the layout and the input parameters, execute runs on parallel machines, monitor the progress of the runs, post process the output data, view plots and retrieve results from runs. The GUI will implement various levels of access for beginner, intermediate, and expert users. The implementation of the GUI is in progress, and the initial prototype is near completion.

[55] The Superstructure, the Physics Module, and the Infrastructure layers are fully implemented and functional.

4. Execution of the SWMF

[56] In this section we outline the main steps in the execution of the SWMF. First, the components are registered and the components are distributed among the pro-

ID	first	last	stride
#COMPONENTMAP			
UA	0	31	1
SP	32	32	1
RB	33	33	1
IM	34	34	1
IE	35	36	1
SC	37	999	2
IH	37	999	2
GM	38	999	2
#END			

Figure 4. An example of the LAYOUT.in file.

processors according to the requested layout. Next, the input parameters are read and transferred to the components. The execution of the components starts after the components are initialized and coupled for the first time. During the execution the components are coupled and restart information is saved according to a predetermined schedule. At the end of the run the components are finalized. The following subsections provide more detail about these steps.

4.1. Registration and Layout of Components

[57] At the beginning of a run, the SWMF “registers” the components. The registration simply means that the control module reads the LAYOUT.in file and checks the list of components that participate in the run. All physically meaningful subsets of the components can be selected for the run.

[58] Next, the control module assigns an MPI communication group to the component based on the processor layout defined in the LAYOUT.in file. An example of this file is shown in Figure 4. The first column identifies the components by their abbreviated names, while the rest of the columns define the ranks of the first and last processing elements (PE) and the PE stride. In the example shown in Figure 4, the UA component runs on the first 32 PEs, the SP, RB, and IM components run on a single PE, the IE component uses two PEs, the SC and IH components use the odd PEs starting from rank 37 up to the last available PE, and the GM component runs on the even PEs starting from rank 38.

[59] If the processor layouts of two components overlap, then they can advance in a sequential time-shared manner only. In our example LAYOUT.in file the SC and IH components use the same processor set, so they can run sequentially only. If the processor layouts of the two components do not overlap, they can advance concurrently. In the example, the IH and GM components do not share any processors, so they can run concurrently. As the example layout shows, the SWMF allows both sequential and concurrent execution or any combination of them.

[60] The possibility of deadlocks is carefully avoided by a temporal and predefined ordering of tasks. Tasks for a component consist of advancing the solution in time by one discrete time step, and performing the next coupling

with other components. Each task is associated with a simulation time, which is known by the component(s) executing the task. If two tasks have the same simulation time, a predefined ordering is used, e.g., component 1 is advanced before component 2, or coupling of the IE and GM components is done before the coupling of the IE and IM components. It can be proved that this ordering of tasks excludes the possibility of deadlocks (G. Tóth, Flexible, efficient and robust algorithm for parallel execution and coupling of components in a framework, submitted to *Computer Physics Communications*, 2005).

4.2. Input Parameters

[61] Stand-alone physics models read their input parameters in various ways. Some models read parameters from the standard input, others read it from some simple text file, and some have all their parameters hard coded in the source code. Some of these options, such as using the standard input or editing the source code, are clearly incompatible with a software framework. Having multiple input parameter files with component specific syntax is a possibility, but it would make the use of the SWMF rather cumbersome. It is also difficult to build a GUI that can handle such a heterogeneous set of input files.

[62] In our design all the input parameters of the SWMF are read from a single file, the PARAM.in file, which may include further parameter files. The component specific parameters are marked by the #BEGIN_COMP and #END_COMP commands (see Figure 5). The text between these markers is sent to the components for reading and checking.

[63] There is no restriction on the format of the component specific parameters. However, it is very beneficial if the component uses the same parameter syntax as the SWMF. First, the parameters given in the SWMF syntax can be easily read with the library provided in the SWMF infrastructure. This library takes care of echoing the input parameters and it does error checking and error reporting. Second, the input parameters can and should be described with an XML file. This XML file contains the commands and the type, valid range, default value, and verbal description of the input parameters. The XML description is used for extensive parameter checking (before the SWMF is run), and it is also used to generate documentation in PDF and HTML formats. Finally, the GUI will be able to provide a user-friendly input parameter editor for all components that use this syntax. All the components currently integrated into the SWMF use the same input parameter format.

[64] Figure 5 shows an example PARAM.in file. The file consists of a list of commands starting with a hash mark (for example, #DESCRIPTION), and the commands are followed by the parameters corresponding to the command. The parameters can be integer or real numbers, logicals, or strings. The #INCLUDE command, for example, has the file name as its string parameter, and it serves to include another parameter file. Comments can be freely placed anywhere between commands or after the commands and parameters.

4.3. Execution Control

[65] The execution of the SWMF is completed in sessions. The parameters of the control module and the

```

#DESCRIPTION
This is a demonstrative example only

Set up a steady state without the IE component

#TIMEACCURATE
F                               DoTimeAccurate

#COMPONENT
IE                               NameComp
F                               UseComp

#BEGIN_COMP GM -----

Include the parameter file with the GM parameters

#INCLUDE
PARAM_GM.in

#END_COMP GM -----

#STOP
100                             MaxIteration
-1.0                            tSimulationMax [sec]

#RUN #####

Switch to time accurate mode for 1 hour

#TIMEACCURATE
T                               DoTimeAccurate

#COMPONENT
IE                               NameComp
T                               UseComp

Couple GM and IE every 10 seconds

#COUPLE2
GM                               NameComp1
IE                               NameComp2
-1                               DnCouple
10.0                             DtCouple

#BEGIN_COMP IE -----

#SAVEPLOT
1                               nFile
min idl                          StringPlot
-1                               DnOutput
10.0                             DtOutput [sec]

#END_COMP IE -----

#STOP
-1                               MaxIteration
3600.0                           tSimulationMax [sec]

```

Figure 5. An example of the PARAM.in file.

components can be changed in each session. This provides more flexibility than using a single set of parameters for the whole run, which is the typical case for stand alone physics models. For example the parameter file in Figure 5 contains

two sessions separated by the #RUN command. The SWMF contains two different session execution modes: time accurate and steady-state.

[66] The steady-state mode can improve the convergence toward a steady state by allowing the different components to run at different speeds in terms of the physical time. This is achieved by executing the components at different frequencies; for example, one component completes one time step at every iteration of the SWMF, while another component is only called every 10th iteration, which we refer to as component subcycling. Most of the components will advance their internal time with various time steps, but the whole SWMF is not synchronized by time. The coupling frequencies and the end of the session are determined by the number of iterations. In the parameter file in Figure 5 the first session is in steady state mode and stops after 100 iterations. In steady-state mode the components are allowed to use various algorithms to accelerate their progress. For example, the BATS-R-US code [Powell *et al.*, 1999], which is used for the GM, IH, and SC components in the SWMF, can use local time stepping. This algorithm allows each grid cell to converge toward steady state as fast as possible, limited only by the local numerical stability limit. The steady-state mode of the SWMF is analogous to the local time stepping but at a much larger scale.

[67] The time accurate mode advances the components in simulation time. The coupling frequencies and the end of the session are determined in terms of the simulation time. In the parameter file in Figure 5 the second session is in time accurate mode and it stops after 3600 s (final #STOP command), while the two-way coupling between the GM and IE components is done at every 10 s (see the #COUPLE2 command).

[68] At the beginning of each session the components are initialized and the interacting components are coupled together for the first time. After this initial coupling the components communicate only when necessary both in steady-state and time accurate modes. This is possible because the coupling frequencies are known by CON from the PARAM.in file. The components advance to the coupling time (or iteration) and only the processors involved in the coupling need to communicate with each other. This allows efficient concurrent execution of the components that do not overlap in their processor layout.

[69] As specified in the PARAM.in file, CON instructs the components to save their current state into restart files periodically. This makes the seamless continuation of a run from a given point possible in the simulation. Checkpoint restart is an essential feature of a robust, user-friendly, and fault-tolerant software design.

[70] At the end of the last session each component finalizes its state. This involves writing out final plot files, closing log files, and printing performance and error reports. After the components have finalized, CON also finalizes and stops the execution.

4.4. Component Coupling

[71] The coupling of the components is realized either with plain MPI calls, which are specifically designed for each pair of interacting components, or via the general SWMF coupling toolkit. We found that there is so much

variation between the physics and numerics of the couplings that it is not useful to make the couplings conform with some abstract general coupler. The only general restriction is that the data sent between the two components must be in SI units. The coupler must ensure that the data is correctly transformed between the two components, but this is individually achieved for each pairwise coupling.

[72] Even the plain MPI couplers use some of the SWMF infrastructure. A one-way coupler between two components is built from three pieces: the get method in the interface of the providing component, the put method in the interface of the receiving component, and the data exchange in the control module, which contains the allocation of the data buffers and all the required MPI communication calls. By breaking the coupler into pieces, each piece becomes simpler: the get and put methods are very specific to the physics module but do not contain communication. The data exchange is specific for the components but not too sensitive about which component version is used. The buffer size or the coordinate transformations can use the grid descriptors provided by the components. The get method performs the interpolations to the requested location, while the put method is usually a simple setting of the variables at the requested grid locations. The data exchange performs the MPI communication, which is often between the root CPUs of the two components. This type of coupling is used when the interpolation and the communication patterns are relatively simple, for example when two serial models with simple structured grids are coupled. For more complex cases the use of the SWMF parallel coupling toolkit is preferred.

[73] The couplings using the SWMF parallel coupling toolkit can couple components based on the following types of distributed grids: (1) 2-D or 3-D Cartesian or spherical block adaptive grid or (2) 1-D, 2-D, or 3-D structured grid. Structured grids include uniform and nonuniform spherical and Cartesian grids.

[74] The SWMF coupling toolkit obtains the grid descriptors from the components at the beginning of the run. The grid descriptor defines the geometry and parallel decomposition of the grid. At the time of coupling the receiving component requests a number of data values at specified locations of the receiving grid (for example, all grid points at one of the boundaries). The geometric locations are transformed, sometimes mapped, to the grid of the provider component by the coupling toolkit. On the basis of the grid descriptor of the provider component, the toolkit interpolates the data values to the requested locations and sends them to the requesting component. The toolkit provides second order linear, bilinear, and trilinear interpolation schemes which work for all the supported grid types. If necessary, the model can also use its own interpolation method in form of a subroutine, which is passed as an argument to the toolkit methods. The interpolation weights and the MPI communication patterns are calculated in advance and saved into a “router” for sake of efficiency. The router is reused in subsequent couplings. The routers are updated only if one of the grids has changed (e.g., due to grid adaptation) or when the mapping between the two components has changed (e.g., due to the rotation of one grid relative to the other).

[75] An interesting problem arises when the receiving component uses an adaptive grid. If the receiving component adapts its grid between two couplings, it may be difficult to get the coupling information for the newly created grid points. In this case it is useful to introduce an intermediate grid. The intermediate grid is usually a simple structured grid covering the region where data is transferred, and it is replicated on all the processors of the receiving component. The intermediate grid should have a sufficiently high resolution to keep the coupling accurate, but it should not be huge, otherwise the storage and the data transfer time may become a problem. The coupling with an intermediate grid consists of the following steps: (1) The providing component interpolates data from its own grid to the intermediate grid. (2) The intermediate grid is collected and sent to all PEs of the receiving component. (3) The receiving component interpolates data from the intermediate grid to its adaptive grid as needed. Often the first step is a trivial mapping because the intermediate grid is identical with (or a part of) the grid of the providing component. Both the plain MPI and the toolkit based couplers can use an intermediate grid.

5. Component Versions

[76] The current version of the SWMF includes the following component versions:

[77] 1. The Solar Corona, Inner Heliosphere, and Global Magnetosphere components are based on the University of Michigan’s BATS-R-US code [Powell *et al.*, 1999]. The SC component uses the physical model of Roussev *et al.* [2003b], which incorporates magnetogram measurements to specify the realistic boundary conditions for the potential magnetic field at the Sun. The highly parallel BATS-R-US code uses a 3-D block-adaptive Cartesian grid, high-resolution shock capturing schemes, and explicit and/or implicit time stepping (G. Tóth *et al.*, A parallel explicit/implicit time stepping scheme on block-adaptive grids, submitted to *Journal of Computational Physics*, 2005, hereinafter referred to as Tóth *et al.*, submitted manuscript, 2005).

[78] 2. The Eruptive Event Generator is implemented as part of the SC/BATS-R-US component. There are two versions: One superposes the Gibson and Low [Gibson and Low, 1998] magnetic flux rope to the background solar wind solution. The other version imposes converging and sheared velocity field at the inner boundary [Roussev *et al.*, 2004]. Both versions were developed at the University of Michigan.

[79] 3. The Solar Energetic Particles component has two serial versions. The first is J. Kóta’s SEP model (Kóta *et al.*, submitted manuscript, 2005), which uses an operator split implicit scheme in one spatial, one pitch angle, and one momentum dimension. This model has been developed at the University of Arizona. The second component version is the Field Line Advection Model for Particle Acceleration (FLAMPA) [Sokolov *et al.*, 2004] using an explicit shock capturing scheme in one spatial and one particle momentum dimension. This model has been developed at the University of Michigan.

[80] 4. The Inner Magnetosphere component is the Rice Convection Model (RCM) [Wolf *et al.*, 1982; De Zeeuw *et al.*, 2004] developed at Rice University. This serial module uses an explicit advection scheme on a 2-D nonuniform spherical grid.

Table 1. Spatial and Temporal Resolutions in the Test Run

Component	Version	Number of Cells	Smallest Cell	Variable/Cell	Time Step
IH	BATSRUS	2,500,000	1 R_S	8	60.0 s
SC	BATSRUS	1,400,000	1/40 R_S	9	0.4 s
GM	BATSRUS	1,300,000	1/4 R_E	8	4.0 s
UA	GITM	64,800	$5^\circ \times 5^\circ \times 2$ km	30	10.0 s
IE	Ridley	33,000	$1.4^\circ \times 1.4^\circ$	1	—
SP	Kota	10,000	$0.1 R_S \times 10^\circ$	150	varies
IM	RCM	3,800	$7.5^\circ \times 0.5^\circ$	150	5.0 s
RB	RRBM	3,800	$7.5^\circ \times 0.5^\circ$	1	60.0 s

[81] 5. The Radiation Belt component is the serial Rice Radiation Belt Model (RRBM) recently developed at Rice University. It solves an adiabatic transformation of phase space density on a 2-D nonuniform spherical grid. Although the RRBM model is fully coupled and it runs, it is not yet fully functional.

[82] 6. The Ionosphere Electrodynamics component is a 2-D spherical electric potential solver developed at the University of Michigan [Ridley *et al.*, 2004; Ridley and Liemohn, 2002]. It can run on one or two processors since the Northern and Southern Hemispheres can be solved in parallel.

[83] 7. The Upper Atmosphere component is implemented by two versions of the Global Ionosphere Thermosphere Model, GITM and GITM2 (A. J. Ridley *et al.*, The Global Ionosphere-Thermosphere Model, submitted to *Journal of Atmospheric and Solar-Terrestrial Physics*, 2005). Both versions have been recently developed at the University of Michigan as fully parallel 3-D spherical models. They use different explicit upwind schemes for the advection and GITM2 uses implicit time integration for the stiff source terms.

6. Test Simulations

[84] We present results of two test simulations that involve all the components of the SWMF. The first simulation is used as a comprehensive test for the framework, and it also serves

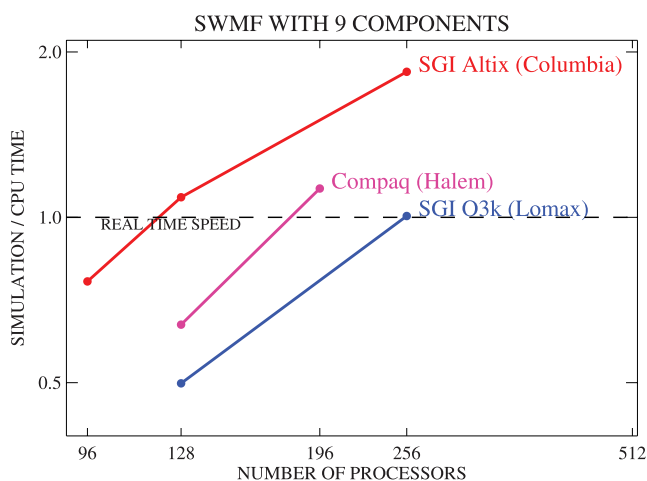


Figure 6. The ratio of simulation and CPU times as a function of the number of processors on various supercomputers. The SWMF can run faster than real time on all three platforms, and it runs almost twice faster than real time on 256 CPUs of the SGI Altix.

as a benchmark for its performance. This test illustrates that the SWMF with all components can run faster than real time with reasonable spatial and temporal resolution. The SWMF could be run in this now-casting mode using observed data.

[85] The second test demonstrates how the eruption and propagation of a single CME can be modeled with the SWMF. At the beginning of the run only the SC and IH components are used. When the CME gets close to the Earth, the SC component can be “switched off” and the GM, IM, IE, and UA components are “switched on.” The GM component is initialized in a steady state mode using the solar wind data obtained from the IH component. Since the SC and GM components are the most expensive, this strategy allows to run the SWMF about two times faster than with all the components. Alternatively, finer grids can be used and still run faster than real time. We did the latter in the second test, i.e., the resolution in the SC and IH components is increased to better follow the CME.

[86] It should be emphasized that neither of these tests try to model an actual space weather event, although we use real magnetometer data to obtain the steady state solar wind conditions. Our intention here is to demonstrate the use of the SWMF. Modeling and analyzing actual events will be done in future papers.

6.1. Performance Test With All Components

[87] We present performance results of a test simulation that involves all components of the SWMF. This simulation

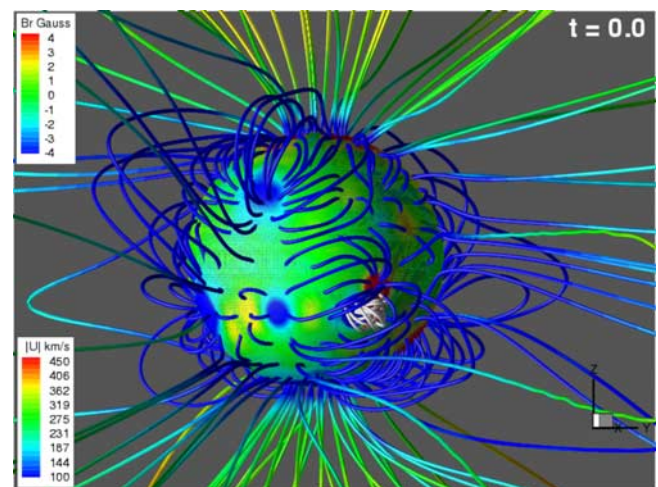


Figure 7. The initial steady state solution obtained by the Solar Corona. The surface of the Sun is colored by the radial magnetic field. The field lines are colored by the velocity. The flux rope added by the Eruptive Event generator is shown with white field lines.

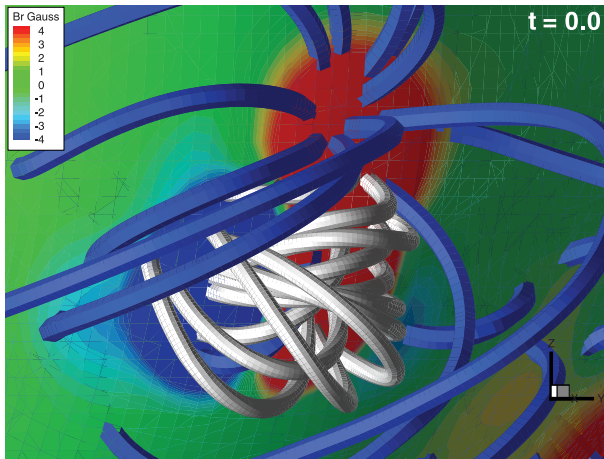


Figure 8. A closeup of the flux rope inserted by the Eruptive Event generator. The surface of the Sun is colored by the radial magnetic field. The field lines of the SC steady state are shown in blue. The flux rope added by the EE component is colored white.

is used as a comprehensive test for the framework, and it also serves as a benchmark for its performance. The simulation grids of all the components have reasonably high resolutions and the time steps and coupling frequencies are all suitable to model space weather. Table 1 contains the grid size, the number of state variables per grid cell, and the time step information about all the components. It is interesting to note that the time step in the SC component

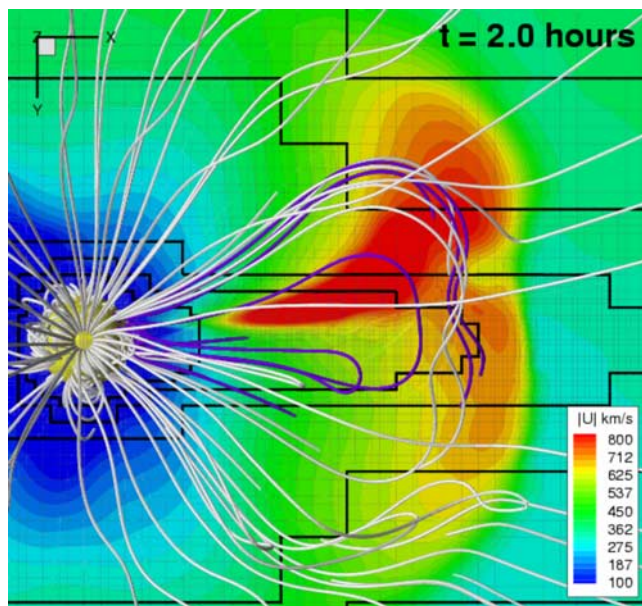


Figure 9. The CME in the Solar Corona 2 hours after the eruption as viewed from the direction of the southern pole ($-Z$ axis). The Sun is shown as a yellow sphere on the left. The $Z = 0$ equatorial plane is color coded with the velocity and the grid resolution changes are indicated by black lines. The magnetic field lines are shown in white and purple. The purple lines emanate from the active region. Sections of the field lines north of the equatorial plane are not visible.

is determined by the stability limit for the explicit scheme, the IH time step is limited by the coupling frequency, while the GM time step is set to be optimal for the explicit/implicit time integration scheme.

[88] The test simulation is built up in several stages: (1) The SC component obtains a magnetogram-driven steady state solution; (2) The IH component obtains a steady state using the SC solution as the inner boundary condition; (3) The GM component is initialized with the solar wind conditions obtained from the IH component; (4) The GM, IM, IE, and UA components start from the GM solution and run a few thousand iterations in the steady state mode to obtain a reasonable initial state for all four components; (5) The EE component initiates a CME by adding a Gibson-Low type flux rope to the SC solution; (6) The SC, IH, and SP components run for 4 hours of simulation time; (7) All components of the SWMF are run in time accurate mode restarted from the previous stages. The final stage serves as our comprehensive test simulation. For code verification, portability checking, and benchmarking, the test is run for 10 min of simulation time only, but it could be continued for much longer time if desired.

[89] The SWMF reaches faster than real-time performance in this test on modern supercomputers. Figure 6 shows the ratio of simulation and CPU times on up to 256 processors of three supercomputers: the SGI Altix system at NASA Ames (Columbia), the SGI Origin 3800 at NASA Ames (Lomax), and the Compaq ES45 at NASA Goddard Space Flight Center (Halem). On 256 CPUs of Columbia, which is the fastest (per CPU) of these super-

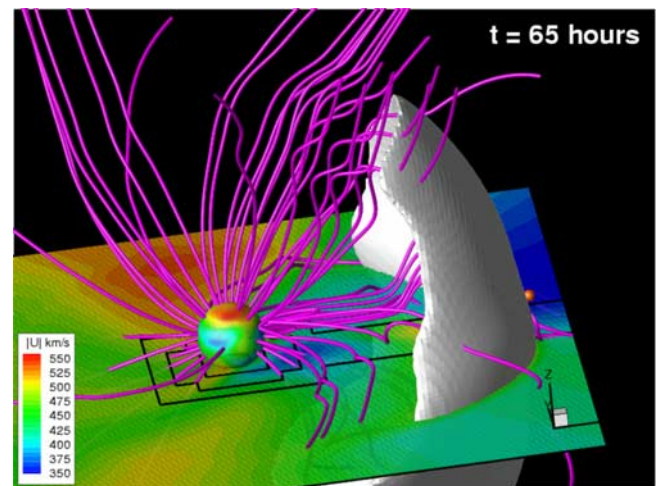


Figure 10. The solution obtained by the Inner Heliosphere after 65 hours of simulation time (1700 9 June). The CME is shown as a white isosurface of density enhancement of a factor of 1.8 relative to the original steady state solution. The inner boundary of the IH component at $20 R_S$ is shown as a sphere colored by the velocity. The velocity (colors) and the grid resolution changes (black lines) are shown in the $Z = -20 R_S$ plane. The magnetic field lines are shown in magenta. The CME is about to reach the Earth which is at the center of the orange sphere on the right. Note that the radius of this sphere is about 400 times larger than the radius of the Earth, and it is much larger than the entire domain of the Global Magnetosphere.

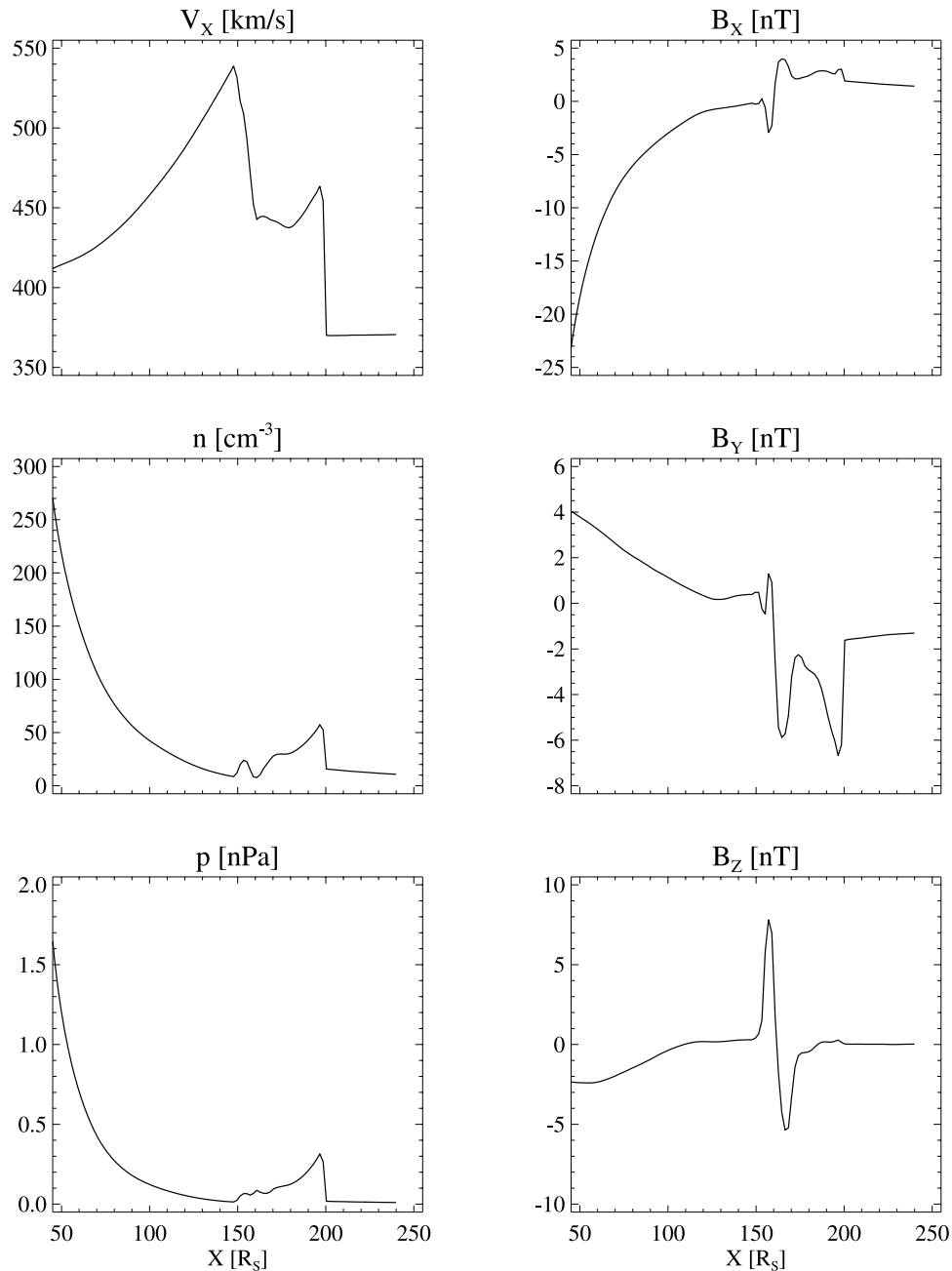


Figure 11. The Inner Heliosphere solution along the X axis at 65 hours of simulation time (1700 9 June).

computers, the SWMF can run almost twice as fast as real time. We emphasize that this speed is achieved with reasonable spatial and temporal resolution: the components use about 47 million state variables in the discretized physics domains!

[90] There are several design and algorithmic choices that make this performance possible. The concurrent execution of the components allows good scaling up to a large number of CPUs. Even if the components do not scale perfectly to hundreds of CPUs, the whole SWMF can because the components can use independent sets of CPUs and they can progress concurrently. It is also important that the SWMF allows the overlap of the

component layouts, since the Inner Heliosphere runs much faster (due to the large time steps) than the other computationally expensive components. Still the IH needs a lot of CPUs for its large grid and the corresponding memory requirements. In the time accurate simulations the SC and IH components use the same set of processors, otherwise all the components run concurrently on an optimal number of CPUs for the given component (see the processor layout in Figure 4).

[91] Another key to achieving this speed is that there are three components to represent the Solar Corona, Inner Heliosphere, and Global Magnetosphere. Each component uses the optimal grid and time stepping scheme in each

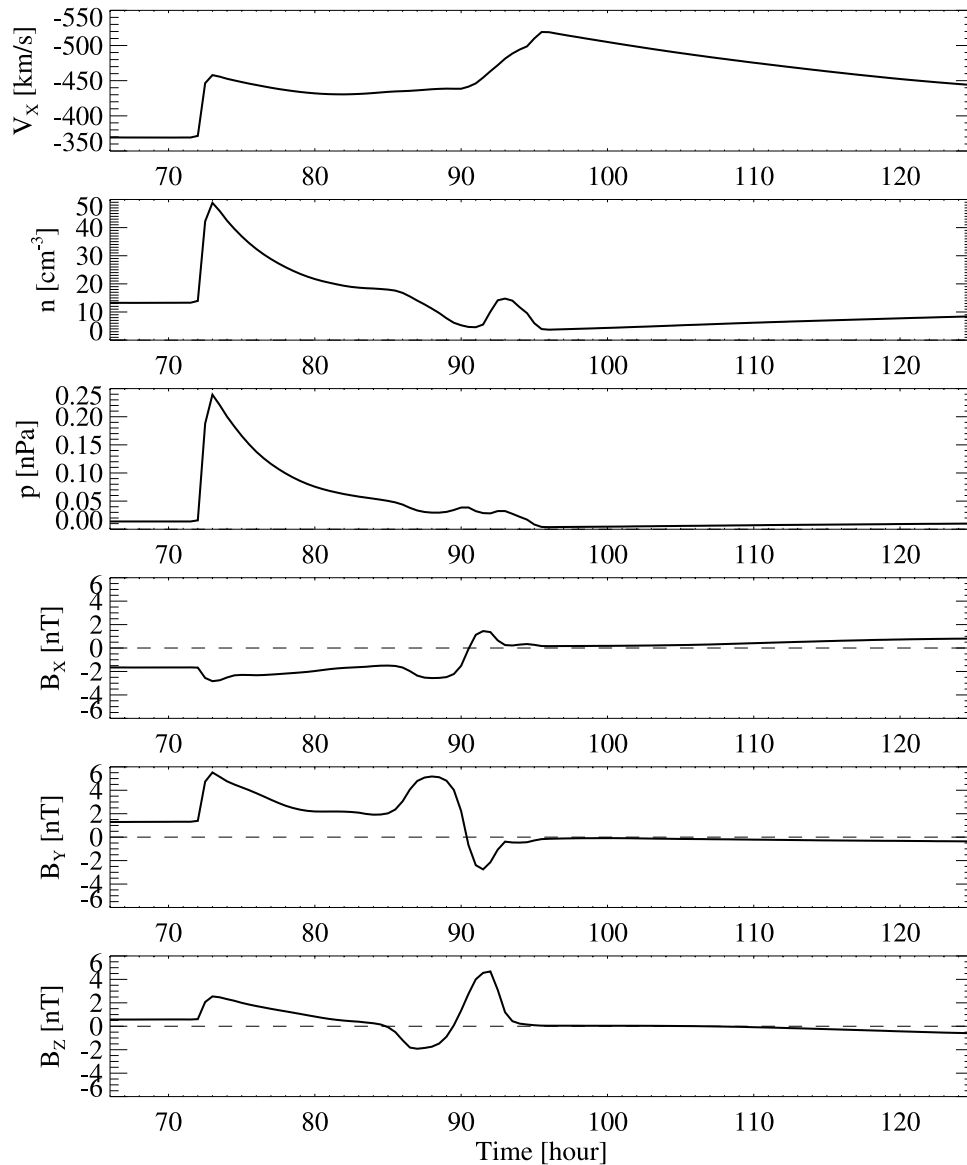


Figure 12. The variation of the solar wind parameters with simulation time between 66 hours (1800 9 June) and 125 hours (0500 12 June). The values are taken at the center of the inflow boundary of the Global Magnetosphere at $X = 32$, $Y = Z = 0$.

domain. The Solar Corona uses explicit time stepping with small 0.4 s time steps on a fine but relatively small grid. The Inner Heliosphere uses much larger 60 s time steps on its large but much coarser grid. Finally, the Global Magnetosphere uses an efficient explicit/implicit time stepping scheme (Tóth et al., submitted manuscript, 2005) on its fine grid, so the time step is not limited by the numerical stability condition. The optimal performance is achieved with 4 s time steps, which speeds up the GM component by about a factor of 30 relative to an explicit scheme.

[92] The efficient coupling of the Global Magnetosphere with the Inner Magnetosphere is also crucial in achieving faster than real-time performance. The two components are coupled every 40 s of simulation time and each coupling involves the mapping of the 1.3 million GM grid cell centers to the IM grid along the dynamically changing magnetic field lines. Owing to the efficient parallel field

line tracing algorithm (Tóth et al., manuscript in preparation, 2005) each coupling takes only a couple of seconds.

6.2. Generation and Propagation of a CME

[93] This simulation is built up in a similar manner as the previous test, but this time the CME is propagated from the solar corona to the magnetosphere of the Earth. The following stages are used: (1) The SC component obtains a magnetogram driven steady state solution; (2) The IH component obtains a steady state using the SC solution as the inner boundary condition; (3) The GM component is initialized with the solar wind conditions obtained from the IH component; (4) The EE component initiates a CME by adding a Gibson-Low type flux rope to the SC solution; (5) The SC and IH components run until the CME gets close to the Earth; (6) The IH, GM, IM, IE, and UA components start from the IH and GM solutions and the CME is

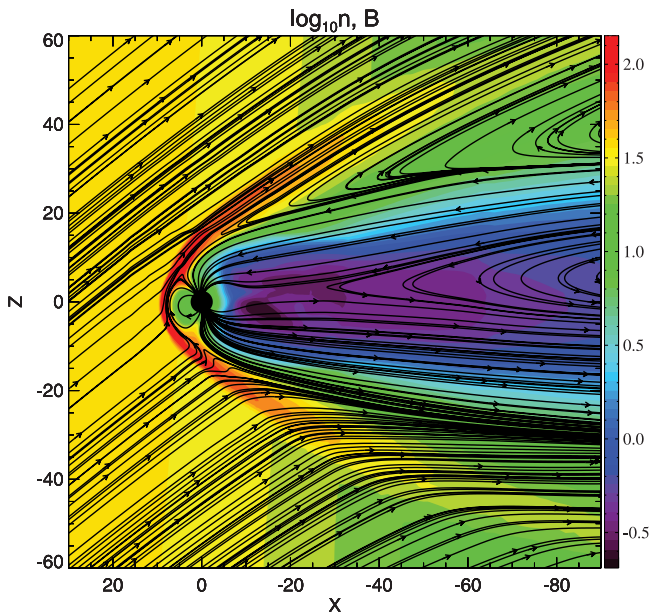


Figure 13. The number density (colors) and traces of the B_x and B_z components of the magnetic field (lines) in the Global Magnetosphere 72.5 hours after the CME initiation (0030 10 June). The black circle shows the inner boundary of the GM domain at $2.5 R_E$. The contours of the density at around $x = 10$ show the tilt of the shock. The shock is smeared out by the $1/4 R_S$ grid resolution of the IH component.

propagated through the magnetosphere. The time accurate runs in stages 5 and 6 are the computationally most expensive. In stage 5 the CME is propagating from the SC to the IH component, so only these two components are run. In stage 6 the CME propagates from the IH to the GM component and it effects all the other components near the Earth, but there is no need to run the SC component in this stage. This partitioning of the simulation saves significant computational resources so that higher-resolution grids can be used in the SC and IH components. The simulation still progresses about two times faster than real time on 256 CPUs of the SGI Altix supercomputer Columbia.

[94] The SC grid is refined to $1/20 R_S$ resolution out to $10 R_S$ along the path of the CME, and a $1/4 R_S$ resolution is maintained along the Sun-Earth axis in the IH grid (see the grid resolution boundaries in Figures 9 and 10). The refinements increase the number of grid cells to 2.5 million and 8.2 million in the SC and IH components, respectively (compare with 1.4 and 2.5 million cells in the performance test). The finer grid results in a better-resolved CME.

[95] Figure 7 shows the magnetogram-based steady state solution in the Solar Corona. We used a magnetogram of the Wilcox Solar Observatory centered around 1 May 1998. To make the simulation results easier to interpret, the starting time is set to 0000 UT 7 June 1998, when the X axis of the HGI and GSM coordinate systems used by the IH and GM components, respectively, is approximately aligned. The coordinate system of the SC component and the magnetogram are also aligned so that the selected active region is toward the positive X axis. For this setup the X axes of the SC, IH and GM components are all (approximately) aligned

with the Sun-Earth line. As the simulation progresses the grid of the SC component is rotating around with the Carrington rotation period of 25.38 days. The SWMF would allow any starting date, this change was introduced for sake of convenient plotting of the results only.

[96] The flux rope was added by the Eruptive Event generator at the 24° southern latitude and 0° longitude near an active region as shown in Figure 8. The flux rope expands to produce a CME, which propagates toward the Earth. The solution is shown 2 hours after the eruption in the SC component in Figure 9. The shock front is well defined by the jump of the velocity from the ambient (about 350 km/s) value to about 650 km/s value. The asymmetric shape of the shock is due to the velocity and magnetic field structure of the ambient solar wind. The high-velocity compression wave behind the shock is formed by the reconnection of the magnetic field lines.

[97] The CME gets in the vicinity of the Earth after 65 hours of simulation time. Figure 10 shows the solution in the Inner Heliosphere. The shock front is visualized as an isosurface of the density enhancement relative to the initial solution. The field lines close to the equatorial plane are bent due to the Parker spiral. The field lines crossing the CME are strongly bent behind the shock front. The plasma parameters along the X axis are shown in Figure 11. The shock front is at $200 R_S$ followed by a rotation of the magnetic field. The velocity peaks at $150 R_S$ with 540 km/s. This gradual velocity increase is probably due to the reconnection of the magnetic flux behind the CME. This second peak is not another shock wave (there is no jump in pressure); rather, it appears to be a compression wave.

[98] The Global Magnetosphere is initialized with the solar wind obtained from the Inner Heliosphere. The boundary conditions are given at the upstream boundary of the GM component at the $x = 32 R_E$ plane (in GSM coordinates). The solar wind parameters at the $x = 32 R_E$, $y = z = 0$ point are velocity $v_x = -369$ km/s, $v_y = 27$ km/s, $v_z = 8$ km/s, magnetic field $B_x = -1.6$ nT, $B_y = 1.3$ nT, $B_z = 0.6$ nT, number density $n = 13$ cm $^{-3}$, and thermal pressure $p = 0.014$ nPa. We note that the transverse velocity is almost entirely due to the orbital motion of the Earth. Figure 12 shows the variation of the solar wind parameters during the time accurate simulation. These representative values are taken at the center point of the inflow boundary of the GM grid. Note that the IH component provides values in the whole plane of the inflow boundary, so the transverse gradients are fully taken into account. Having the transverse gradients eliminates the problem of a time varying B_x component, which would otherwise result in a finite divergence of the incoming magnetic field.

[99] The variation in the magnetic field is relatively mild, but there is a well-defined rotation of the magnetic field direction between 85 and 95 hours of simulation time. The initial shock front between 72 and 73 hours and the compression wave between 90 and 95 hours are clearly visible in the velocity, density, and pressure variations. As the CME shock reaches the Earth, the solar wind velocity jumps from 370 km/s to 460 km/s, and the number density increases from 13 cm $^{-3}$ to 49 cm $^{-3}$. The shock front is resolved by about 4–6 grid points in the IH grid, which has $1/4 R_S$ resolution along the Sun-Earth line. This results in a

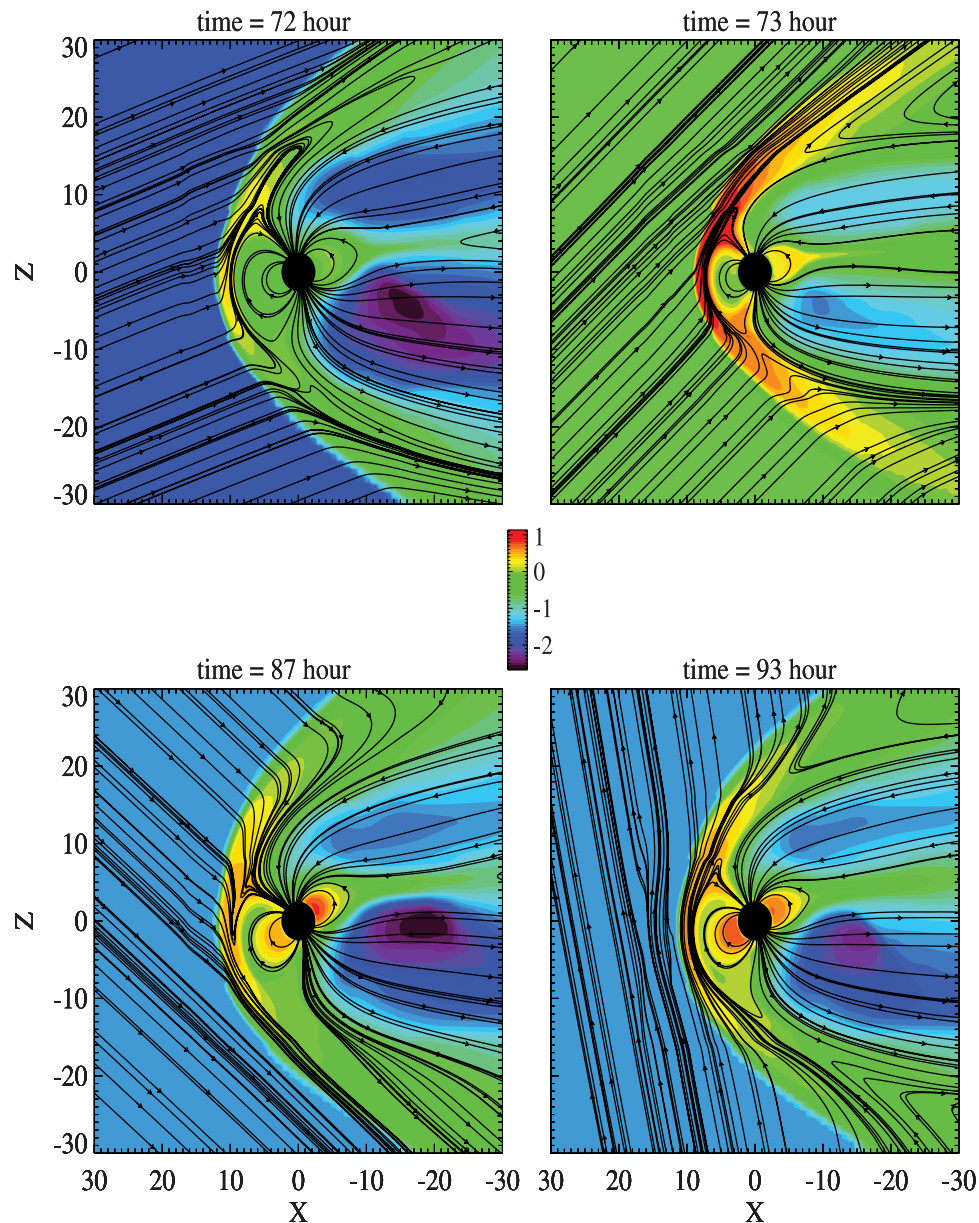


Figure 14. The 10 based logarithm of pressure in nPa (colors) and traces of the B_x and B_z components of the magnetic field (lines) in the Global Magnetosphere at various times after the CME initiation: before the shock arrival (top left), after the shock arrival (top right), during the southward B_z period (bottom left) and after the compression wave (bottom right). The simulation times correspond to 0000, 0100, 1500, and 2100 UT, respectively, on 10 June.

shock width of about $160 R_E$ so the shock front passes in about 40 min of simulation time.

[100] Figure 13 shows the shock front entering the GM domain. Note that the tilt angle of the shock is readily obtained from the IH component. This information is usually not available from the point-wise satellite measurements. Figure 14 displays the effect of the shock and the following CME on the magnetosphere. The pressure behind the bow shock sharply increases from 72 to 73 hours of simulation time due to the shock arrival. The bow shock moves in from $12 R_E$ to $9 R_E$. The southward turning of the magnetic field at 85 hours results in a significant increase of the pressure in the closed field line region near the Earth due

to the coupling with the Inner Magnetosphere model. By 87 hours of simulation time, the bow shock has moved back to $12 R_E$. The compression wave arrives at around 93 hours and it pushes back the bow shock to $11 R_E$. The northward B_z component results in a reduced pressure in the closed field line region near the Earth.

[101] The distortion of the magnetosphere can be characterized with the D_{st} index, which we estimate as the B_z component averaged over the surface of a sphere of radius $3.5 R_E$ centered around the Earth. This radius is chosen to be far enough from the inner boundary at $2.5 R_E$. The variation of the D_{st} and the cross polar cap potentials is shown in Figure 15. The variations are relatively mild due

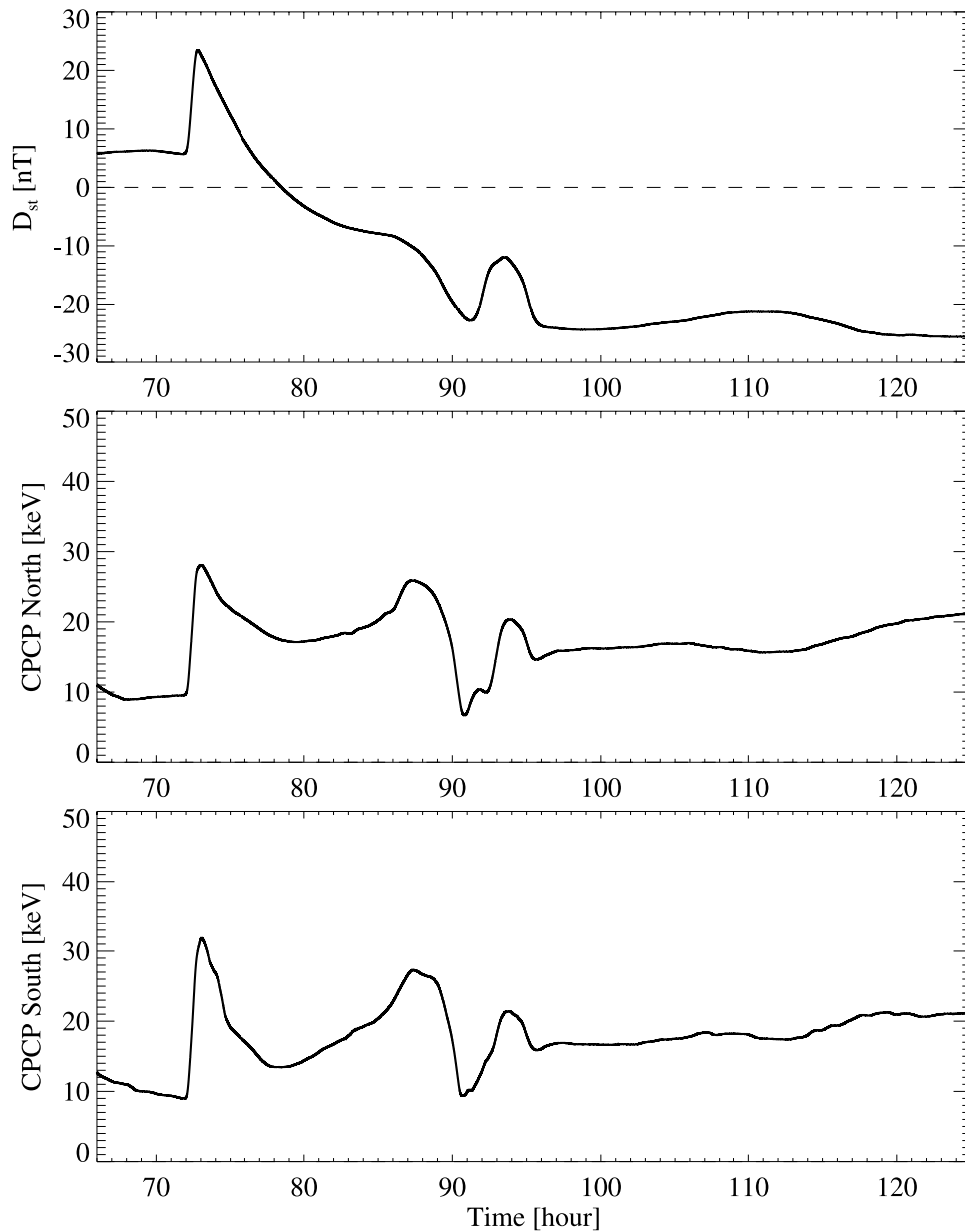


Figure 15. The variation of the D_{st} index and the cross polar cap potentials between 66 hours (1800 9 June) and 125 hours (0500 12 June) of simulation time.

to the weak magnetic field in the simulated CME, but they are consistent with the expected response of the coupled GM, IM, IE, and UA components. The D_{st} index increases from 6 nT to 25 nT, while the cross polar cap potentials jump from 10 kV to about 30 kV when the shock hits the magnetosphere at 72.5 hours of simulation time. As the ring current builds up, the D_{st} drops to about -25 nT. The solar wind B_z turns northward at around 89 hour, which causes a drop in the cross polar cap potentials. The compression wave arriving between 90 and 95 hours temporarily increases the D_{st} index and the cross polar cap potentials. As the simulation continues, the D_{st} index and the cross polar cap potentials remain roughly constant with a moderate increase after 110 hours. This may be associated with the slow southward turning of the B_z component of the solar

wind and the gradual increase of the density as shown in Figure 12.

[102] The CME shock reaches the Earth at about 72.5 hours of simulation time, which corresponds to 0030 UT on 10 June 1998 for the selected eruption time 0000 7 June. Figures 16 and 17 show the field-aligned current and the resulting electric potential for the Northern and Southern Hemispheres, respectively, before (0000 UT) and after (0100, 0200, and 0300 UT) the shock arrival. The field aligned currents are received by the Ionosphere Electrodynamics component from the Global Magnetosphere, and the IE component calculates the potential. Note the increase of the field-aligned currents at 0100 UT which results in a peak in the cross polar cap potentials. Later at 0200 and 0300 UT the region-2 currents build up and the

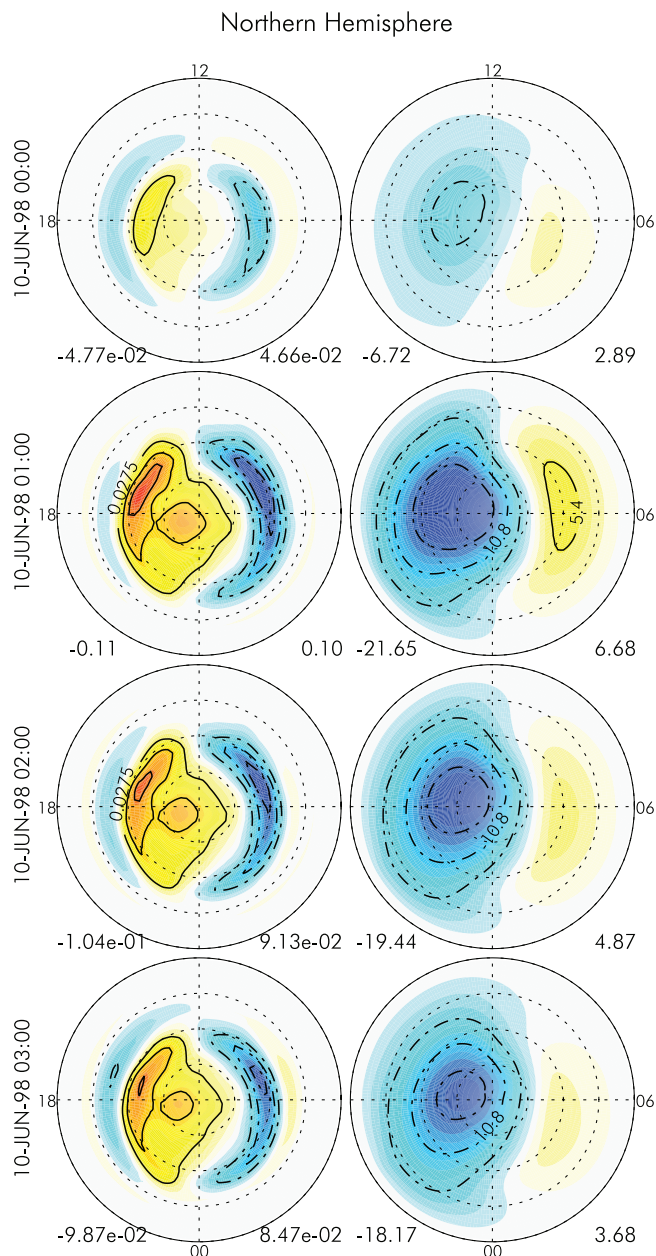


Figure 16. The field-aligned current (in mA/m^2 , left) and the resulting electric potential (in kV, right) for the Northern Hemisphere just before (top) and after the shock arrival. In each plot the center of the circle is the north magnetic pole and the outermost circle corresponds to the 60° magnetic latitude. The noon meridian (1200) is at the top, dusk (1800) and dawn (0600) are on the left and right, respectively. The minimum (blue) and maximum (red) values are shown below each plot. See text for discussion.

shielding effect reduces the electric potentials, while the region-1 current remains roughly the same as at 0100 UT.

[103] This test has demonstrated that the SWMF can efficiently model the eruption of a CME, its propagation in the interplanetary space, and the interaction with the coupled magnetosphere-ionosphere-thermosphere system of the Earth. Our model CME is fairly weak but not atypical of

CMEs generating moderate magnetic storms. We attribute this to the initiation mechanism, which used, for the first time, a relatively small flux rope embedded into a magnetogram based solar wind solution. Other simulations with the BATS-R-US code [Manchester et al., 2004, 2005] used more energetic flux ropes, which generated faster CMEs with much stronger magnetic fields.

[104] The GM, IM, IE, and UA components of the SWMF have been tested with extremely strong CMEs, such as the storms in October 2003, using satellite observations for the solar wind input. We found that the framework can model even these extreme events robustly and efficiently, and we were able to reproduce the huge changes in the D_{st} index and the extreme compression of the magnetosphere. These simulations will be the subject of future papers.

7. Conclusions and Future Work

[105] Our purpose is to introduce the Space Weather Modeling Framework to the space physics community, describe its potential, and encourage participation in its future development. The SWMF is continuously improved and extended. It is our hope that the space physics community will join us in our efforts to integrate new physics component versions and new physics components into the SWMF. The SWMF has been successfully ported to the Community Coordinated Modeling Center (CCMC), and it is available for community use to do runs on demand. The SWMF is also publicly available as an open source code at the <http://csem.engin.umich.edu/SWMF/> Web site, but access requires registration. The current public release of the framework is a prototype for modeling the Sun-Earth system and for doing space weather studies, and it should be used in close collaboration with CSEM. The exact terms and conditions of registration can be found at the Web site.

[106] The SWMF consists of the core of the framework and the implemented component versions. To get an impression of the size of the SWMF and its various parts, here are some representative numbers. The core is about 23,000 lines written in an object oriented style in Fortran 90. The components consist of about 200,000 lines written in Fortran 90 and Fortran 77. The component wrappers and the coupler interfaces are about 10,000 lines of Fortran 90 code. There is about 8000 lines of Perl and shell scripts that help the installation, configuration, source code manipulation, binary data conversion, manual generation, parameter checking, etc. The output of some of the components can be visualized with the help of about 17,000 lines of IDL scripts.

[107] The SWMF is well documented. The 122-page user manual explains the use of the SWMF with examples, and it contains a full documentation of all the input parameters generated from the XML files. There is a reference manual that is generated from the comments in the source code of the core of the SWMF. The testing procedures and the maintenance are also documented.

[108] The SWMF can run on any Unix/Linux based system with a Fortran 90 compiler, the MPI library, and a Perl interpreter. The SWMF can run on a laptop with one or two components; however, it scales well to several hundreds

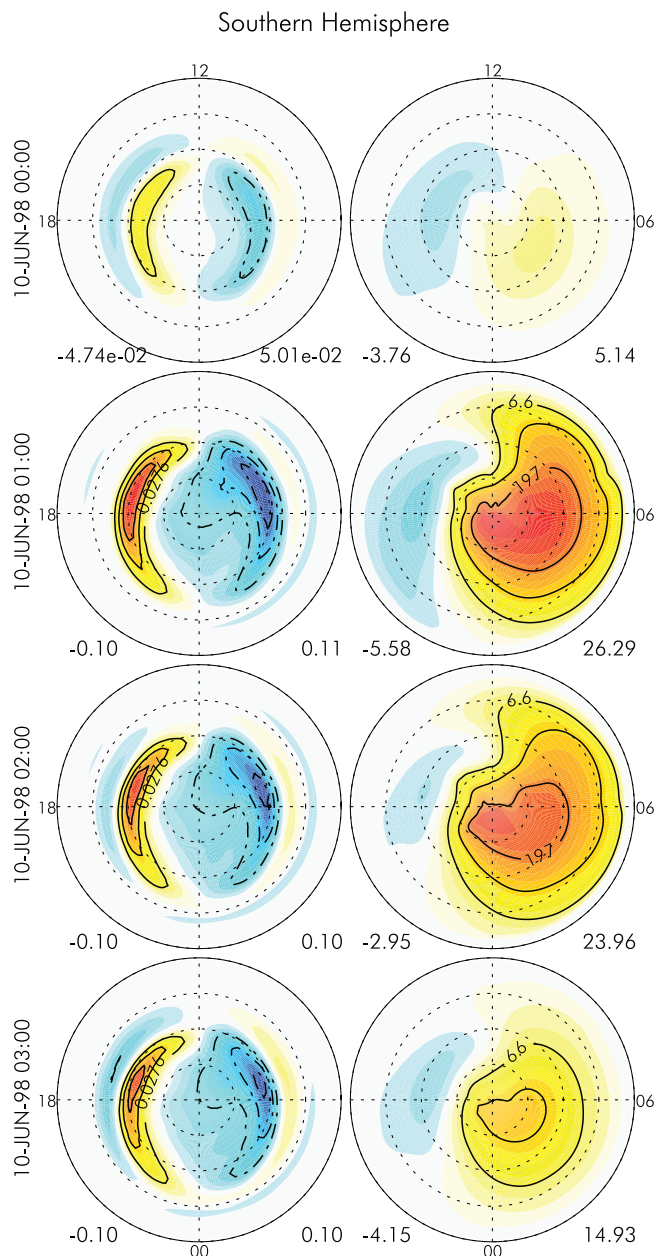


Figure 17. Same as Figure 16 for the Southern Hemisphere.

of processors of the fastest supercomputers with all the components running together. The SWMF has been ported to a large number of platforms and compilers: the SGI Altix system with the Intel compiler, the SGI Origin 3000 systems with the MIPS f90 compiler, the Compaq ES45 with the HP f90 compiler, several Linux clusters with the NAG f95 and Portland Group pgf90 compilers, and the Macintosh OSX with the NAG f95 and IBM xlf90 compilers.

[109] In parallel with the code development, we are also working on the validation of the individual components and the coupled framework. We have extensively modeled the quiet northward IMF conditions that occurred on 22–23 October 2003, and we have also simulated the Halloween storms of 30 October 2003. These simulation results and

their comparison with measurements will be presented in other papers.

[110] **Acknowledgments.** At the University of Michigan this work has been supported by NASA AISRP grant NNG04GP89G, NASA ESS CT cooperative agreement NCC5-614, and by DoD MURI grant F49620-01-1-0359. G.T. has been partially supported by the the Hungarian Science Foundation (OTKA, grants T047042 and T037548).

[111] Shadia Rifai Habbal thanks the referees for their assistance in evaluating this paper.

References

- Allen, G., W. Bengert, T. Goodale, H.-C. Hege, G. Lanfermann, T. Radke, E. Seidel, and J. Shalf (2000), The Cactus code: A problem solving environment for the grid, in *9th IEEE International Symposium on High-Performance Distributed Computing*, pp. 25–32, IEEE Comput. Soc. Press, Los Alamitos, Calif.
- Antiochos, S. K., C. R. DeVore, and J. A. Klimchuk (1999), A model for solar coronal mass ejections, *Astrophys. J.*, *510*, 485.
- Beutier, T., and D. Boscher (1995), A three-dimensional analysis of the electron radiation belt by the Salammbó code, *J. Geophys. Res.*, *100*, 14,853.
- Buis, S., D. Declat, E. Gondet, S. Massart, T. Morel, and O. Thual (2003), PALM: A dynamic parallel coupler for data assimilation, paper presented at EGS-AGU-EUG Joint Assembly, Eur. Geophys. Soc., Nice, France.
- Chen, J., and D. Garen (1993), Interplanetary magnetic clouds: Topology and driving mechanism, *Geophys. Res. Lett.*, *20*, 2319.
- Chen, M. W., M. Schulz, G. Lu, and L. R. Lyons (2003), Quasi-steady drift paths in a model magnetosphere with AMIE electric field: Implications for ring current formation, *J. Geophys. Res.*, *108*(A5), 1180, doi:10.1029/2002JA009584.
- De Zeeuw, D. L., S. Sazykin, R. A. Wolf, T. I. Gombosi, A. J. Ridley, and G. Tóth (2004), Coupling of a global MHD code and an inner magnetosphere model: Initial results, *J. Geophys. Res.*, *109*, A12219, doi:10.1029/2003JA010366.
- Ebihara, Y., and M. Ejiri (2002), Numerical simulation of the ring current: Review, *Space Sci. Rev.*, *105*, 377.
- Ebihara, Y., M. Ejiri, H. Nilsson, I. Sandahl, M. Grande, J. F. Fennell, J. L. Roeder, D. R. Weimer, and T. A. Fritz (2004), Multiple discrete-energy ion features in the inner magnetosphere: 9 February 1998, event, *Ann. Geophys.*, *22*, 1297.
- Elkington, S. R., M. K. Hudson, and A. A. Chan (1999), Acceleration of relativistic electrons via drift-resonant interaction with toroidal-mode Pc-5 ULF oscillations, *Geophys. Res. Lett.*, *26*, 3273.
- Fok, M.-C., R. A. Wolf, R. W. Spiro, and T. E. Moore (2001), Comprehensive computational model of the Earth's ring current, *J. Geophys. Res.*, *106*, 8417.
- Forbes, T. G., and P. A. Isenberg (1991), A catastrophe mechanism for coronal mass ejections, *Astrophys. J.*, *373*, 294.
- Fuller-Rowell, T. J., and D. S. Evans (1987), Height-integrated Pedersen and Hall conductivity patterns inferred from TIROS-NOAA satellite data, *J. Geophys. Res.*, *92*, 7606.
- Gibson, S. E., and B. C. Low (1998), A time-dependent three-dimensional magnetohydrodynamic model of the coronal mass ejection, *Astrophys. J.*, *493*, 460, doi:10.1086/305107.
- Gombosi, T. I., D. L. De Zeeuw, C. P. Groth, K. G. Powell, C. R. Clauer, and P. Song (2001), From Sun to Earth: Multiscale MHD simulations of space weather, in *Space Weather, Geophys. Monogr. Ser.*, vol. 125, edited by P. Song, H. J. Singer, and G. L. Siscoe, pp. 169–176, AGU, Washington, D. C.
- Gombosi, T. I., et al. (2004), Solution-adaptive magnetohydrodynamics for space plasmas: Sun-to-Earth simulations, computing in science and engineering, *Frontiers Simul.*, *6*, 14–35.
- Groth, C. P. T., D. L. De Zeeuw, T. I. Gombosi, and K. G. Powell (2000), Global three-dimensional MHD simulation of a space weather event: CME formation, interplanetary propagation, and interaction with the magnetosphere, *J. Geophys. Res.*, *105*, 25,053.
- Gurnis, M., et al. (2003), GeoFramework: A modeling framework for solid earth geophysics, *Eos. Trans. AGU*, *84*(46), Fall Meet. Suppl., Abstract NG12D-06.
- Hill, C., C. DeLuca, V. Balaji, M. Suarez, A. da Silva, and the ESMF Joint Specification Team (2004), The architecture of the Earth System Modeling Framework, *Comput. Sci. Eng.*, *6*, 18.
- Kóta, J., and J. R. Jokipii (1999), The transport of CIR accelerated particles, *Proc. Int. Conf. Cosmic Rays 26th*, *6*, 512.
- Leboeuf, J.-N., V. Decyk, A. Dimits, and D. Shumaker (2003), Gyrokinetic calculations of ITG turbulence in general toroidal geometry within the Summit Framework, paper presented at 45th Annual Meeting of the Division of Plasma Physics, Am. Phys. Soc., Albuquerque, N.M.

- Liemohn, M. W., A. J. Ridley, D. L. Gallagher, D. M. Ober, and J. U. Kozyra (2004), Dependence of plasmaspheric morphology on the electric field description during the recovery phase of the April 17, 2002 magnetic storm, *J. Geophys. Res.*, *109*, A03209, doi:10.1029/2003JA010304.
- Low, B. C. (2001), Coronal mass ejections, magnetic flux ropes, and solar magnetism, *J. Geophys. Res.*, *106*, 25,141.
- Luhmann, J. G., S. C. Solomon, J. A. Linker, J. G. Lyon, Z. Mikic, D. Odstrcil, W. Wang, and M. Wiltberger (2004), Coupled model simulation of a Sun-to-Earth space weather event, *J. Atmos. Sol. Terr. Phys.*, *66*, 1243.
- Manchester, W. B., IV (2003), Buoyant disruption of magnetic arcades with self-induced shearing, *J. Geophys. Res.*, *108*(A4), 1162, doi:10.1029/2002JA009252.
- Manchester, W. B., IV, T. I. Gombosi, I. I. Roussev, A. J. Ridley, D. L. De Zeeuw, I. V. Sokolov, K. G. Powell, and G. Tóth (2004), Modeling a space weather event from the Sun to the Earth: CME generation and interplanetary propagation, *J. Geophys. Res.*, *109*, A02107, doi:10.1029/2003JA010150.
- Manchester, W. B., IV, A. J. Ridley, T. I. Gombosi, and D. L. De Zeeuw (2005), Modeling the Sun-Earth propagation of a very fast CME, *Adv. Space Res.*, in press.
- Moen, J., and A. Brekke (1993), The solar flux influence of quiet-time conductances in the auroral ionosphere, *Geophys. Res. Lett.*, *20*, 971.
- Powell, K. G., P. L. Roe, T. J. Linde, T. I. Gombosi, and D. L. De Zeeuw (1999), A solution-adaptive upwind scheme for ideal magnetohydrodynamics, *J. Comput. Phys.*, *154*(2), 284, doi:10.1006/jcph.1999.6299.
- Reynders, J. (1996), POOMA: A framework for scientific simulations on parallel architectures, in *Parallel Programming Using C++*, edited by G. F. Wilson and P. Lu, pp. 553–594, MIT Press, Cambridge, Mass.
- Richmond, A. D., and Y. Kamide (1988), Mapping Electrodynamic features of the high-latitude ionosphere from localized observations: Technique, *J. Geophys. Res.*, *93*, 5741.
- Richmond, A. D., E. C. Ridley, and R. G. Roble (1992), A thermosphere/ionosphere general circulation model with coupled electrodynamics, *J. Geophys. Res.*, *96*, 1071.
- Ridley, A. J., and M. W. Liemohn (2002), A model-derived stormtime asymmetric ring current driven electric field description, *J. Geophys. Res.*, *107*(A8), 1151, doi:10.1029/2001JA000051.
- Ridley, A. J., D. L. De Zeeuw, T. I. Gombosi, and K. G. Powell (2001), Using steady-state MHD results to predict the global state of the magnetosphere-ionosphere system, *J. Geophys. Res.*, *106*, 30,067.
- Ridley, A. J., T. I. Gombosi, D. L. De Zeeuw, C. R. Clauer, and A. D. Richmond (2003), Ionospheric control of the magnetospheric configuration: Neutral winds, *J. Geophys. Res.*, *108*(A8), 1328, doi:10.1029/2002JA009464.
- Ridley, A. J., T. I. Gombosi, and D. L. De Zeeuw (2004), Ionospheric control of the magnetospheric configuration: Conductance, *Ann. Geophys.*, *22*, 567.
- Robinson, R. M., R. R. Vondrak, K. Miller, T. Dabbs, and D. A. Hardy (1987), On calculating ionospheric conductances from the flux and energy of precipitating electrons, *J. Geophys. Res.*, *92*, 2565.
- Roussev, I. I., T. G. Forbes, T. I. Gombosi, I. V. Sokolov, D. L. De Zeeuw, and J. Birn (2003a), A three-dimensional flux rope model for coronal mass ejections based on a loss of equilibrium, *Astrophys. J.*, *588*, L45.
- Roussev, I. I., T. I. Gombosi, I. V. Sokolov, M. Velli, W. Manchester, D. L. De Zeeuw, P. Liewer, G. Tóth, and J. Luhmann (2003b), A three-dimensional model of the solar wind incorporating solar magnetogram observations, *Astrophys. J.*, *595*, L57.
- Roussev, I. I., I. V. Sokolov, T. G. Forbes, T. I. Gombosi, M. A. Lee, and J. I. Sakai (2004), A numerical model of a coronal mass ejection: Shock development with implications for the acceleration of GeV protons, *Astrophys. J.*, *605*, L73.
- Shprits, Y. Y., and R. M. Thorne (2004), Time dependent radial diffusion modeling of relativistic electrons with realistic loss rates, *Geophys. Res. Lett.*, *31*, L08805, doi:10.1029/2004GL019591.
- Sokolov, I. V., I. I. Roussev, T. I. Gombosi, M. A. Lee, J. Kóta, T. G. Forbes, W. B. Manchester, and J. I. Sakai (2004), A new field line advection model for solar particle acceleration, *Astrophys. J.*, *616*:L171, L174.
- Toffoletto, F., S. Sazykin, R. Spiro, and R. Wolf (2003), Inner magnetospheric modeling with the Rice Convection Model, *Space Sci. Rev.*, *107*, 175.
- Usmanov, A. V., M. L. Goldstein, B. P. Besser, and J. M. Fritzer (2000), A global MHD solar wind model with WKB Alfvén waves: Comparison with Ulysses data, *J. Geophys. Res.*, *105*, 12,675.
- Weimer, D. R. (1996), A flexible, IMF dependent model of high-latitude electric potential having “space weather” applications, *Geophys. Res. Lett.*, *23*, 2549.
- Wolf, R. A., M. Harel, R. W. Spiro, G.-H. Voigt, P. H. Reiff, and C. K. Chen (1982), Computer simulation of inner magnetospheric dynamics for the magnetic storm of July 29, 1977, *J. Geophys. Res.*, *87*, 5949.
- Wu, S. T., W. P. Guo, D. J. Michels, and L. F. Bulgara (1999), MHD description of the dynamical relationships between a flux rope, streamer, coronal mass ejection, and magnetic cloud: An analysis of the January 1997 Sun-Earth connection event, *J. Geophys. Res.*, *104*, 14,789.
- Wu, S. T., W. P. Guo, S. P. Plunkett, B. Schmieder, and G. M. Simnett (2000), Coronal mass ejections (CMEs) initiation: Models and observations, *J. Atmos. Sol. Terr. Phys.*, *62*, 1489.

A. Chan, S. Sazykin, R. A. Wolf, and B. Yu, Rice University, MS-108, 6100 South Main Street, Houston, TX 77005-1892, USA.

D. R. Chesney, C. R. Clauer, D. L. De Zeeuw, T. I. Gombosi, K. C. Hansen, K. J. Kane, W. B. Manchester, R. C. Oehmke, K. G. Powell, A. J. Ridley, I. I. Roussev, G. Tóth, I. V. Sokolov, Q. F. Stout, and O. Volberg, Center for Space Environment Modeling, University of Michigan, Ann Arbor, MI 48109-2143, USA. (gtoth@umich.edu)

J. Kóta, University of Arizona, 1629 E. University Blvd., Tucson, AZ 85721, USA.