Draft version February 19, 2024
Typeset using LaTeX **twocolumn** style in AASTeX631

Field Line Universal relaXer (FLUX): A Fluxon Approach to Coronal Magnetic Field Modeling

Chris Lowder [1], Chris Gilly [1], and Craig DeForest [1]

[1] *Southwest Research Institute, Boulder, CO 80302, USA*

ABSTRACT

We describe a novel method for modeling the global, steady solar wind using photospheric magnetic fields as a driving boundary condition. Prior wind models in this class include both rapid heuristic methods that use potential field extrapolation and variants thereof, trading rigor for computation speed, and detailed 3D magnetohydrodynamic (MHD) models that attempt to simulate the entire solar corona with a degree of physical rigor, but require large amounts of computation. The Field Line Universal relaXer (FLUX), an open-source numerical code which implements the 'fluxon' semi-lagrangian approach to MHD modeling, provides an intermediate approach between these two general classes. In particular, the fluxon approach to MHD describes the magnetic field through discrete analogues of magnetic field lines, relaxing these structures to a stationary state of force balance. In this work we introduce a one-dimensional solar wind solution along each fieldline, providing an ensemble of solutions that are interpolated back onto a uniform grid at an outer boundary surface. This provides advantages in physical rigor over heuristic semi-analytic techniques, and in computational efficiency over full 3D MHD techniques. Here we describe the underlying methodology and the FLUXPipe modeling pipeline process.

1. INTRODUCTION

The solar wind plays a primary role in the space weather environment at Earth. Further, the wind is thought to be driven by boundary conditions low down: magnetic conditions near the photosphere and physical conditions in the solar corona (eg. Parker 1958; Gibson 1973; Zirker 1977; Cranmer 2009). The magnetic field shapes the solar wind both as a source of energy (and, indirectly, mass from the solar photosphere) and as a conduit that affects the accelerating flow through the spatial relationships between local flux tube geometry and location of energy and momentum deposition (eg. Wang 1993; Suess et al. 1998).

Current models of global solar wind flow fall into two major classes: (1) heuristic models such as WSA (Wang & Sheeley 1990; Arge & Pizzo 2000; Arge et al. 2003, 2004) or WSA/CSSS (Poduval & Zhao 2014) that rely on analytic extrapolation of field geometry, coupled with empirical heuristics to describe the relationship between field geometry and wind flow; and (2) 3D magnetohydrodynamic models that attempt to simulate the corona directly, using resistive MHD and models of the underlying physics (Downs et al. 2010; Rempel et al. 2009; Mackay & Yeates 2012). The former are very fast to implement, but are very limited in their ability to capture the coronal geometry and solar wind behavior. The latter more directly model the underlying physics of the solar corona, however are famously very difficult and slow to implement, and they also suffer from effects such as numerical resistivity that diffuses the sharp topological boundaries observed in the physical corona.

The Field Line Universal relaXer (FLUX) simulation code models the solar corona as a collection of discrete analogue magnetic domains, via a quasi-Lagrangian grid of discrete field lines ('fluxons'). Each of these fluxon structures carries a finite quantity of magnetic flux, interacting with neighboring fluxons and relaxing in time. This approach is computationally efficient, scales easily, and avoids certain grid-based numerical issues - falling between the two classes of models previously described. This provides a unique and multi-functional tool for coronal modeling. Each fluxon provides an ideal framework for one-dimensional solar wind modeling, which in concert provides full solar wind mapping to the edge of the corona.

Corresponding author: Chris Lowder
chris.lowder@swri.org

The fluxon approach has been used primarily to study magnetic stability and the importance of reconnection to magnetic eruption processes such as CMEs (Rachmeler et al. 2009, 2010), in the 'zero-beta' domain with no included plasma. Recently, we have adapted the FLUX code to support steady solar wind solutions and to interact with other modeling codes in a hybrid modeling framework. The techniques we adopted are of interest because of the unique challenges of the fluxon approach to modeling, and in the present article we describe those techniques as a prelude to future work comparing the fluxon framework to other solar wind models.

The fluxon method was originally described by DeForest & Kankelborg (2007); here we describe the current implementation and its application to solar wind modeling, including adaptations to both the code and the underlying force laws and method. In §2 we describe the methodology behind this work, including the building blocks and mechanics of the fluxon model (§2.1) and methods of solar wind modeling (§2.2). The encapsulation of an automated framework for fluxon modeling of the solar wind is described in §3. Results from the model are described in §4. A summary of results and discussion are provided in §5, with final thoughts and conclusions in §6.

2. METHODOLOGY

2.1. *Coronal modeling*

The *Field Line Universal relaXer* (FLUX) code provides a framework for modeling MHD evolution in the solar corona (DeForest & Kankelborg 2007). 'Fluxons' are the piecewise-linear analogue for magnetic field lines, composed of segments connecting individual vertices. Each fluxon carries a finite quantity of magnetic flux. Within the simulation, forces are calculated and applied to vertex points, relaxing to an equilibrium state. Given the nature of the computed forces, the resulting output is a nonlinear force-free field. Figure 1 illustrates a relaxed fluxon coronal model.

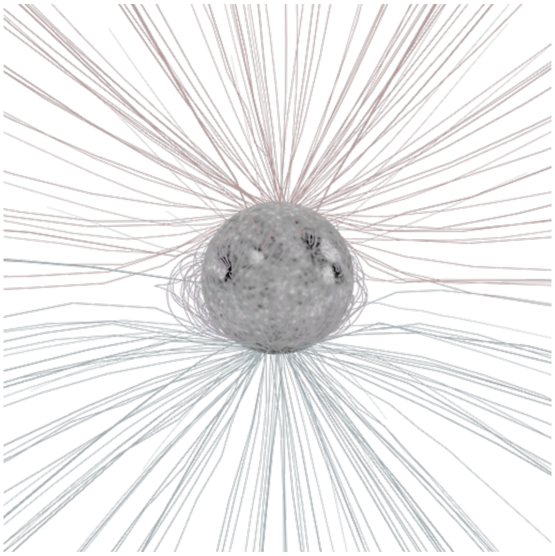

Figure 1. Relaxed geometry from the FLUX model. A synthetic map of B_r is displayed in gray, with fluxons color mapped by radial extent.

2.1.1. *What are fluxons?*

Each fluxon structure represents a geometrical analogue for a bundle of magnetic fieldlines along a particular path through the solar corona. Figure 2 shows a cartoon schematic of a fluxon solar corona for reference. A definite quantity of magnetic flux is contained along each fluxon. Each fluxon structure is broken down into a series of component parts, as outlined below in a glossary of terms.

- Fluxon - Solar magnetic fieldline analogue structure
- Fluxel - Line segments along fluxons
- Vertex - Points of intersection between fluxel segments

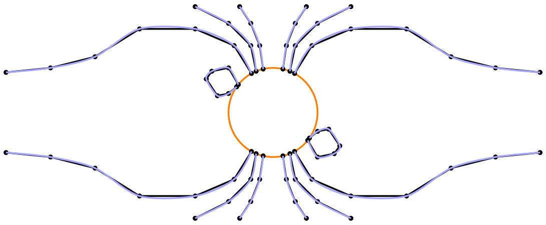

Figure 2. Schematic fluxon representation of solar magnetic field lines. Closed and open magnetic structures are displayed in blue with exaggerated scales.

A fluxon structure itself remains intact through the relaxation process, unless manual reconnection is utilized to alter the topology of the system. However, fluxels and vertices along a given fluxon can be dynamically adjusted through relaxation, adapting to meet the geometrical requirements imposed. New fluxels and vertices are inserted to better approximate fieldline curvature without excessive angular changes between segments. In the opposite case, where fieldline curvature relaxes, excess fluxels and vertices are removed to reduce the computational burden.

With physical quantities only defined at points along each fluxon, interpolation is required for calculating magnetic and plasma parameters at arbitrary locations, or for interfacing with grid-based simulations. As with many interpolation schemes, the first step involves the selection of suitable points. For the simple case of falling directly atop a fluxon vertex point, that value can be directly assigned. In the more general case, for an arbitrary interpolant point within the simulation domain, ideally a 3-simplex of four points can be selected that enclose the spatial point in question. The interpolation algorithm searches the neighborhood of fluxon vertex points, and from nearby points constructs an ideal set of four vertex points that surround the interpolant point while minimizing the 3-simplex volume. The interpolated value is generated by weighing each vertex point by the volume of the 3-simplex formed by the remaining three neighboring points and the interpolant point itself.

For a sufficiently populated domain with sufficient fluxon vertex points from which to choose, creating an ideal neighborhood of points is generally not an issue. For unique geometries, or near the edge of fluxon occupied domain, suitable points can become more sparse. In these edge cases, the interpolation routine falls back in dimensionality, selecting and weighing from a 2-simplex, a 1-simplex, or in the rarest of cases a 0-simplex (from 3, 2, and 1 point, respectively). These cases run the risk of enhanced errors, and are toggled 'off' by default.

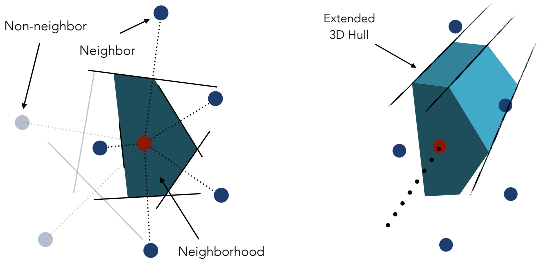

Figure 3. Two-dimensional (left) and three-dimensional (right) representation of defining the neighborhood of a fluxel segment. A two-dimensional encapsulating polygon composed of perpendicular bisectors to neighboring fluxel projections is extended into a three-dimensional hull volume. This process is repeated for each fluxel segment along a given fluxon.

Though fluxons are one-dimensional objects, a volumetric representation of each fluxon is useful when computing the spatial domain of each element. Figure 3 shows a schematic of this definition. For each fluxel element, a perpendicular plane is constructed, and intersections with nearby fluxons computed. From these intersecting points a minimum area two-dimensional polygon is defined from perpendicular bisectors, classifying adjacent fluxons as neighbors or non-neighbors. This two-dimensional polygon is extruded into three dimensions, defining a hull volume. This process is then repeated for each fluxel segment of a fluxon, and for every fluxon.

2.1.2. *How do they relax?*

The FLUX model works to relax a given initial magnetic topology at a given timestep to a force-free state. By calculating forces (magnetic pressure, magnetic tension from fieldline curvature, and a pseudo-force to space vertices along fluxons) at each vertex point from neighboring vertices and fluxel segments, these structures can be shifted in the net force direction, repeating this process until a sufficiently relaxed state is reached.

Equation 1 describes the magnetic curvature force at each vertex point. $\Delta\theta_v$ defines the angle extended by the two fluxel segments extending from that vertex, and l describes the length of the fluxel segment. Integrating along from the previous vertex to the next, the curvature force simply reduces to this angular difference between segments.

$$F_{cn,v} = \frac{1}{2}\int_{v-1}^{v+1} \mathcal{F}_{cn}\, ds = \frac{l}{2}\frac{2\Delta\theta_v}{l} = \Delta\theta_v \tag{1}$$

Equation 2 describes the magnetic pressure force exerted on each fluxel segment. The integration is performed from one vertex point to the next, along the length of the fluxel, and reduces down to a perpindicular gradient of the magnetic field. Due to the nature of the fluxon model, the magnetic field within each hull cross-section of a fluxel is simply the magnetic flux subdivided into angular wedges made from tesselation with neighboring fluxels. This term then simply reduces to a sum over each hull segment i, involving the normal to each wedge segment boundary $\hat{n_i}$, the angular extent of each $\Delta\phi_i$, and the radius out to that hull boundary edge r_i. Figure 4 displays these geometric elements in the context of a fluxel hull segment.

$$\mathbf{F}_{pn} = \int_{v}^{v+1} \mathcal{F}_{pn}\, ds = l\frac{\langle \nabla_\perp B\rangle}{\langle B\rangle} = l\sum_i (\hat{n}_i \Delta\phi_i)/(\pi r_i) \tag{2}$$

Considering these two forces described, the net force on each vertex point is given by Equation 3. The curvature force applies directly to the vertex point. The second set of terms describes the average force between the two fluxel segments neighboring the vertex point in question. Figure 5 shows a schematic of how these forces apply for a given vertex point.

$$\mathbf{F}_v = \mathbf{F}_{cn,v} + \frac{1}{2}\left(\mathbf{F}_{pn,f} + \mathbf{F}_{pn,f-1}\right) \tag{3}$$

Finally, the computed force at each vertex point is used to shift said vertex point in space, moving to-

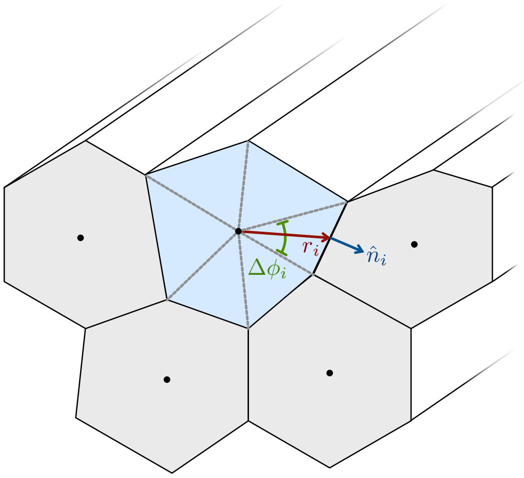

Figure 4. Schematic diagram illustrating the cross-section of a fluxel hull segment, along with adjacent fluxel hulls. The angle subtended by each hull segment is given by $\Delta\phi_i$, with radius to the outer boundary as r_i, and the normal to the hull boundary given as $\hat{n}_i$.

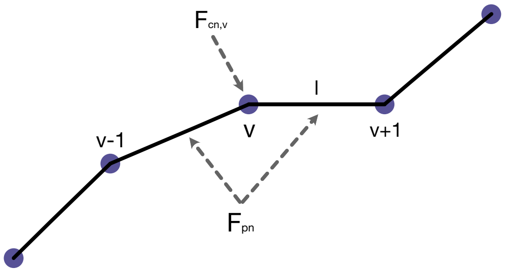

Figure 5. Schematic diagram illustrating a given vertex point v, and adjacent fluxel segments. The acting locations for both the magnetic curvature force, $F_{cn,v}$, and the magnetic pressure force, F_{pn}, are displayed.

wards relaxation. Equation 4 describes this shift, as a quasi-Eulerian step. Each of the forces F_j are summed over. The squared term in parentheses acts as a stiffness coefficient, preventing endless corrections near equilibrium. The remaining terms describe the step in space $\partial\tau$ and the distance to the nearest neighbor $r_{min,i}$ (within neighboring fluxels). Together, these terms work to relax to a near force-free state, while controlling stiffness to prevent oscillation around equilibrium.

$$\Delta\mathbf{x}_i = \delta\tau \left(\frac{|\sum \mathbf{F}_j|}{\sum |\mathbf{F}_j|} \right)^2 r_{min,i} \sum \mathbf{F}_j \tag{4}$$

During this process the structure of each fluxon can dynamically adjust to the given geometry. Where the curvature between fluxel segments exceeds a user-defined threshold, additional vertex points and fluxels are dynamically inserted to allow for this curvature to relax. Additionally, where a given curvature is over-defined by accumulated vertex points and fluxels, these are consolidated to reduce computational burden during the relaxation process. A further description of the relaxation process is found in the original methodology paper (DeForest & Kankelborg 2007).

2.1.3. *How can we model a data-driven corona?*

The FLUX model allows for placement of fluxon structures extending from an inner photospheric boundary at 1 $R_\odot$ out to 21.5 $R_\odot$ (0.1 AU). Note that while open fluxon structures are rooted to the photosphere at 1 $R_\odot$ on one end, the remaining end is free to relax to an equilibrium position. This outer 'boundary' at 21.5 $R_\odot$ is used to provide a surface on which to interpolate values of intersecting fluxons to compare with other models and observations, and to demarcate open fluxon structures. This surface does not restrict fluxon movement or positioning, as fieldlines are free to relax while intersecting this surface.

Input magnetograms can be utilized to setup an initial fluxon 'world' for relaxation. Here the term world refers to the data structure containing appropriate boundary conditions, fluxons (and contained vertex points), along with relational data. This world structure is the analogue in the fluxon model of the grid-based simulation domain. This world is allowed to relax and respond to internally generated forces.

Taking an input magnetogram, the distribution of magnetic flux can be segmented into discrete fluxons using a form of dithering. Dithering, which has etymological roots in manual vibrations reducing error in mechanical computers, can be used in a digital context for purposes of converting a grayscale image to a binary black-and-white version. The error created in each pixel conversion is shifted along a defined pattern, distributing the error across the image. This method is suited for the conversion of magnetogram distributions of magnetic flux (ranging scalar values in each pixel) into a map of fluxon locations (each fluxon carrying discrete units of magnetic flux). To avoid a directional bias when dithering input magnetograms, the area of each map is read using a Hilbert curve tracing (Hilbert 1891). For more details on using Hilbert curves for initializing fluxon placement see Rachmeler et al. (2010).

The FLUX model is able to assimilate various types of magnetic input data, including synoptic magnetograms, synchronic magnetograms, flux transport model data, or combinations thereof. These initial states are then allowed to relax given appropriate internal magnetic forces from fluxon structures. The resulting time series represents a series of nearly force-free equilibria, wherein fluxons structures can be linked and tracked.

The combination of an input magnetogram and seeded fluxon footprint locations can be used to define an initial three-dimensional fluxon distribution. These two data sets are used to trace the corresponding fluxons into the coronal volume using a potential field source surface (PFSS) model - `pfsspy` (Stansby et al. 2020). This sets the initial topological state of the model, which then undergoes subsequent relaxation to a nonlinear force-free field. Note that although the initial open / closed topology is set by a PFSS model, the resulting relaxed field solution is a Nonlinear Force Free Field (NLFFF) in nature, and extends far beyond the typical source surface of a PFSS model. The FLUX model contains the ability to manually reconnect field line structures, which is of particular interest to study the nature of open / closed boundary reconnection in future work.

One of the direct benefits of this assimilation methodology is the detailed mapping of open magnetic flux linkages. Of particular interest is the detailed response of the quantity and distribution of open magnetic flux to changes in the underlying magnetic configuration (emergence, cancellation, migration, etc). With direct mapping along each fluxon structure from source to open end, these relations can be studied in more robust detail.

2.2. *Solar wind modeling*

A one-dimensional open fluxon provides a unique structure on which to compute a solar wind profile. The hull area profile of each fluxon is stored along the length of it, providing an expansion factor.

If we start with a modified Parker 1D solar wind solution,

$$\frac{dv}{dr} = \frac{v}{r}\,\frac{rg - 2c_s^2\left[1 + \frac{r}{2f_r}\left(\frac{df_r}{dr}\right)\right]}{c_s^2 - v^2}, \tag{5}$$

where our sound speed c_s and expansion factor $f(r)$ are defined by

$$c_s = \sqrt{\frac{2kT}{\mu m_p}} \quad \text{and} \quad f(r) = \frac{A(r)}{A_0}\left(\frac{r_0^2}{r^2}\right), \tag{6}$$

we can iterate to find a transonic solution for each open fluxon given an initial set of boundary conditions at the root. Each fluxon has an initial base temperature of 1e6 K. A binary search tree is utilized to seek out this solution for each open fluxon by adjusting these initial parameters at the base, with an example of such a search illustrated in Figure 6. Note in particular the sequence of solar 'breeze' solutions - those indicated from blue to yellow. These solutions fail to reach the critical supersonic point. Also note the large discontinuity plotted in red - a nonphysical solution encountered through the numerical integration process. A singularity in Equation 5 at the sound speed requires the use of an adaptive integration technique to bypass this point. After iterating with this search tree and discarding invalid solutions, the algorithm hones in towards a final velocity profile. Given a series of open fluxon structures, the methodology outlined above can be repeated en masse, mapping an ensemble of transonic solar wind solutions onto a two-dimensional output surface.

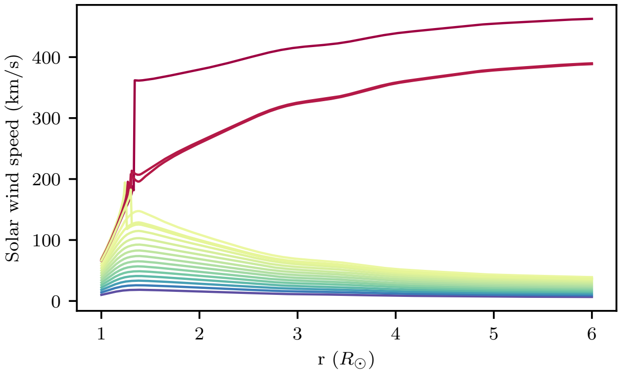

Figure 6. Example of binary search tree method for finding a transonic solar wind velocity profile for a given open fluxon. An initial solar breeze solution is found (blue), with increasingly warm colors honing in on the ultimate stable transonic solution (red).

3. FLUXPIPE

FLUXPipe represents a pipelinification of the FLUX code, such that one can specify a Carrington Rotation of interest and a number of magnetic footpoints, and the code will carry out the entire pipeline with no additional input.

As an overview, the pipeline performs these steps, which will be described in the following subsections:

1. Retrieve and process a polar corrected HMI synoptic chart of radial magnetic field strength.

2. Trace a Hilbert curve across the image to find footpoints that represent equal amounts of magnetic flux.

3. Trace the footpoints into the corona using the `pfsspy.pfss` function, determining an initial magnetic topology.

4. Use the FLUX relaxation code to relax the fields into a realistic topology and near force-free state, but allowing for currents (such that it is no longer a potential field).

5. Generate solutions for the solar wind velocity within the open magnetic field lines.

This effectively allows one to use the fluxon code in a straightforward way to output a data-driven solar wind map generated at a 21.5 $R_\odot$ radius surface.

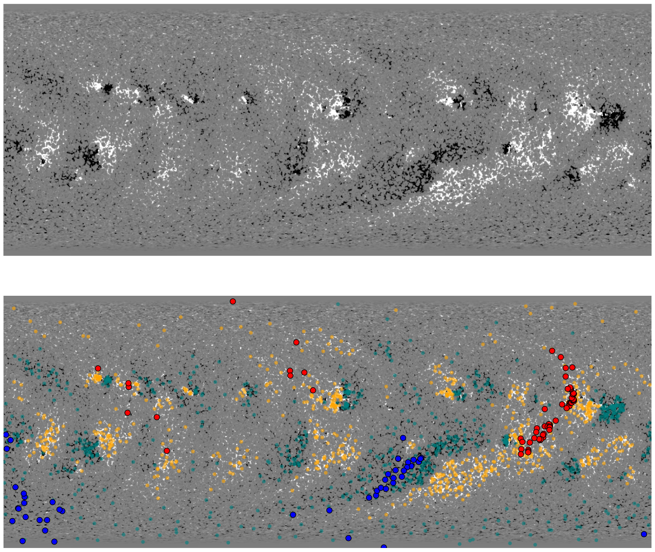

Figure 7. *(top)* Input HMI polar-corrected synoptic magnetogram for Carrington Rotation 2160. *(bottom)* Footpoint placement of fluxons using the Hilbert curve methodology. Open footpoints are mapped in red (positive) and blue (negative), with closed footpoints mapped in orange (positive) and teal (negative).

3.1. *Pipeline usage*

To run the code, one simply calls the `config_runner.py` file with the Carrington Rotation number of interest as input. One can specify a single rotation or a list of rotations, as well as several other parameters, in the `config.ini` file. The runner file then calls `magnetogram2wind.pdl`, which is the main pipeline script, and utilizes both PDL and Python libraries and scripts. In the following subsections we will walk through each step of the FLUXPipe algorithm using Carrington Rotation 2160 as an example case.

3.1.1. *Data retrieval and processing*

An input polar-field-corrected HMI synoptic chart of radial magnetic field strength is retrieved using the Joint Science Operations Center (JSOC) through SunPy. It is then downsampled using the `astropy.nddata.block_reduce` function with `numpy.nansum` as the reduction function. This map is plotted in Figure 7 for reference.

3.1.2. *Fluxon footpoints*

The magnetic map is then traced with a Hilbert curve to find footpoints that represent equal amounts of magnetic flux. The equal-flux value is iterated using Eulerian minimization until the number of defined footpoints is close to the number requested by the user. The second panel of Figure 7 shows these footpoints for CR2160.

3.1.3. *Topology initialization*

The footpoints that result from the Hilbert algorithm are traced into the corona using the `pfsspy.pfss` function. A tracing function is defined from the input magnetogram, after which the initialized footpoints are traced into the corona, and are separated into open and closed fieldlines. Beyond 2.5 $R_\odot$, open fieldlines are extended radially out to 21.5 $R_\odot$. This provides an initial condition and input topology of a PFSS magnetic field geometry, which is then ingested by the FLUX code to generate a collection of fluxons that can be used by FLUX. The first panel in Figure 8 shows this initial unrelaxed world.

3.1.4. *Relaxation*

As outlined in §2.1.2, forces are computed along each fluxon in the simulation, with spatial coordinates shifted in relaxation space towards a force-free equilibrium. This relaxes the fields into a realistic topology and near force-free state, but allows for non-potential fields by allowing currents. The second panel of Figure 8 shows the result of 2000 steps of relaxation.

3.1.5. *Wind calculation*

Open fluxons are gathered and the solar wind calculation begins as outlined in §2.2. For each open fluxon, the one-dimensional Parker solar wind equation (Equation 5) is integrated along its length, using the computed expansion factor along the way. A binary search tree integration method is used to hone in on a transonic solution. The resulting solar wind solutions are gathered together as an ensemble, and mapped at their intersection with a spherical surface at 21.5 $R_\odot$. Other methods for calculating the solar wind are also being implemented for model verification and to meet future needs.

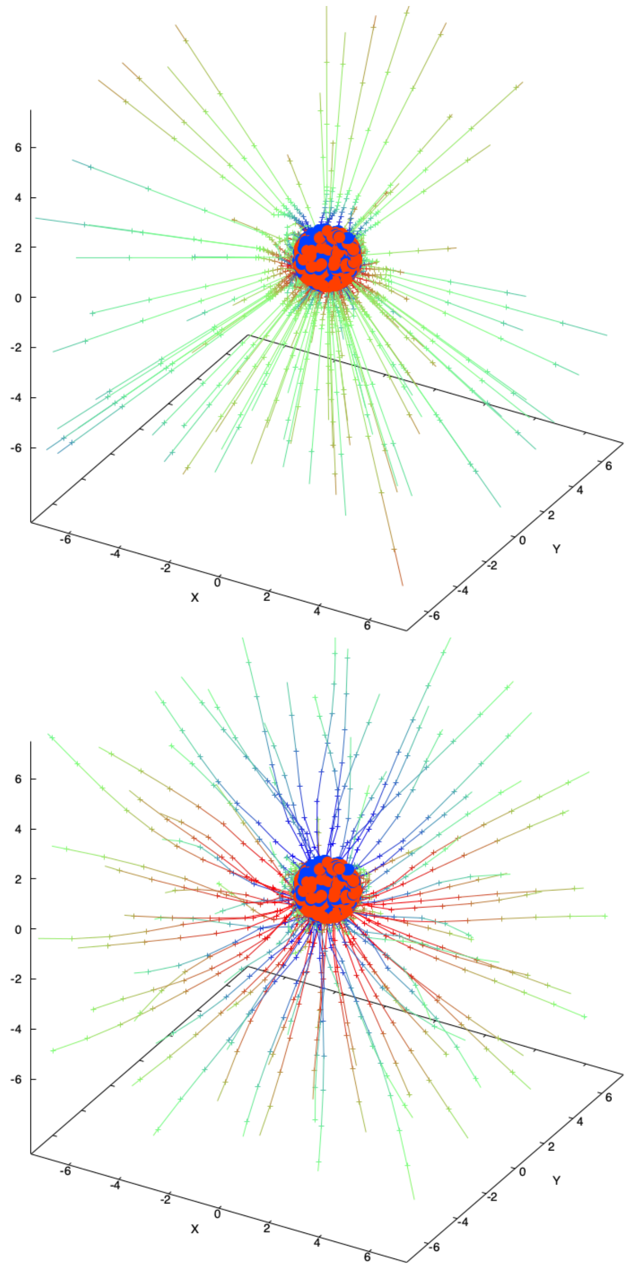

Figure 8. Carrington rotation 2160: *(upper)* The fluxon simulation as initialized. *(bottom)* The fluxon simulation after 2000 steps of relaxation, in a state of transition towards a fully relaxed force-free field state.

4. RESULTS

To demonstrate the output of the FLUX code we have chosen to examine four Carrington rotations of interest. These rotations mark specific points in the rise and fall of the solar magnetic activity cycle. The behavior of the solutions fall into two relatively distinct groups as shown in Figure 9 and Figure 10.

In these figures, the panels are as follows: Panel (a) shows what at which point in the solar cycle the Carrington rotation lies, allowing for a general understanding of the context of each figure. Panel (b) shows the locations of the footpoints of the open magnetic fields at the base of the simulation. The color of these points corresponds to the latitude of the footpoints, which becomes useful in the following panel to understand the source region of the various features in the plot. Panel (c) shows the locations of the fluxons at the outer boundary of the simulation. Because the color of each point is the same as in panel (b), one can correlate the lower and upper points of specific bundles of fluxons. Panel (d) displays the log ratio of the expansion factors at the bottom and top of the domain. Darker blue points indicate more expansion. The size of the points also grows as a function of the expansion ratio. In essence, this panel shows how much expansion the fluxons have undergone in their path from the lowest to the highest part of the domain. Panel e) shows the calculated wind speed at the top of the domain, which is strongly correlated with the expansion factor. For panels (b-e) we use a sine latitude projection to maintain equal area per pixel for comparison with other maps, and to not overemphasise the poles. Panel (f) presents a histogram of those wind speeds.

4.1. *Model observations*

Between all of these rotations, the distribution of solar wind speeds maintains a clustering around 600 km/s, with faster and slower outliers as a result of the nozzling effect of the changing computed expansion factor along each open fluxon.

Carrington rotation 2135, occurring towards the peak of solar activity cycle 24, displays relatively little open polar magnetic field; The open field lines are all rooted in equatorial regions. The various features in the equatorial zone expand to fill the space, but it isn't possible to predict the wind speed by considering footpoint location alone. Panels d) and e) demonstrate the correlation between the solar wind speed and the expansion factor ratio.

Carrington rotation 2160, on the other side of the peak of solar activity cycle 24, displays somewhat similar behavior. In this case, however, a more distinct patch of open magnetic field is rooted at the southern pole. These polar fields take up more of the outer boundary, leading to moderate wind speeds in that region. On the right side of the plot, however, a group of open field lines from the equator have expanded by several orders of magnitude, and this has led to a large region of much faster wind speeds. Again, the expansion factor plays a key role in determining the acceleration of solar wind outflows.

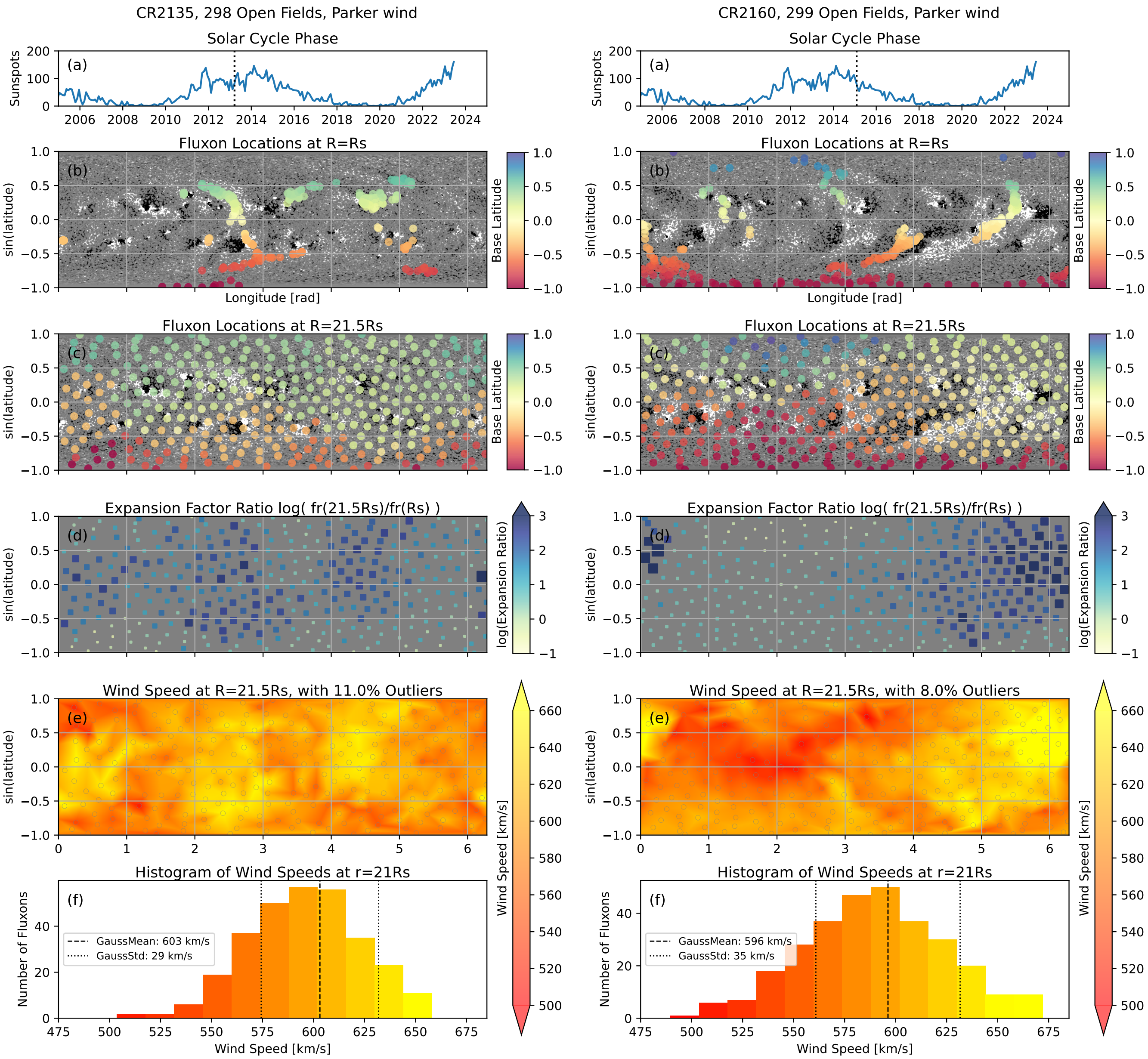

Figure 9. Results of running FLUXPipe on Carrington rotations CR2135 and CR2160. These rotations were at the middle and ending of solar maximum, respectively. (a) The location within the solar cycle is marked with a vertical dotted line. (b) Footpoints of open fluxons overlying the HMI magnetogram, with color representing the footpoint latitude. (c) The location of the fluxons on the outer boundary at $R = 21.5R_s$. The colors match the latitude of the fluxon footpoints, allowing for correlations to be drawn. (d) The log of the ratio of the expansion factor (Equation 6) between the two altitudes. (e) the solar wind speed as determined by integrating Equation 5. (f) A histogram of those wind speeds.

Carrington rotation 2193, situated in the declining phase of solar cycle 24, shows remarkably differing behavior. The majority of open fluxons are rooted at both the north and south poles, with pockets of open field at lower latitudes. In this case, the polar fields dominate the domain, and the equator to pole variation of wind speed is the dominant feature. We do see the expansion effects of small parcels of equatorial open magnetic field, particularly the isolated patch near 2 radians (115 degrees). As this expands it leave a significant trace in the map of expansion factors.

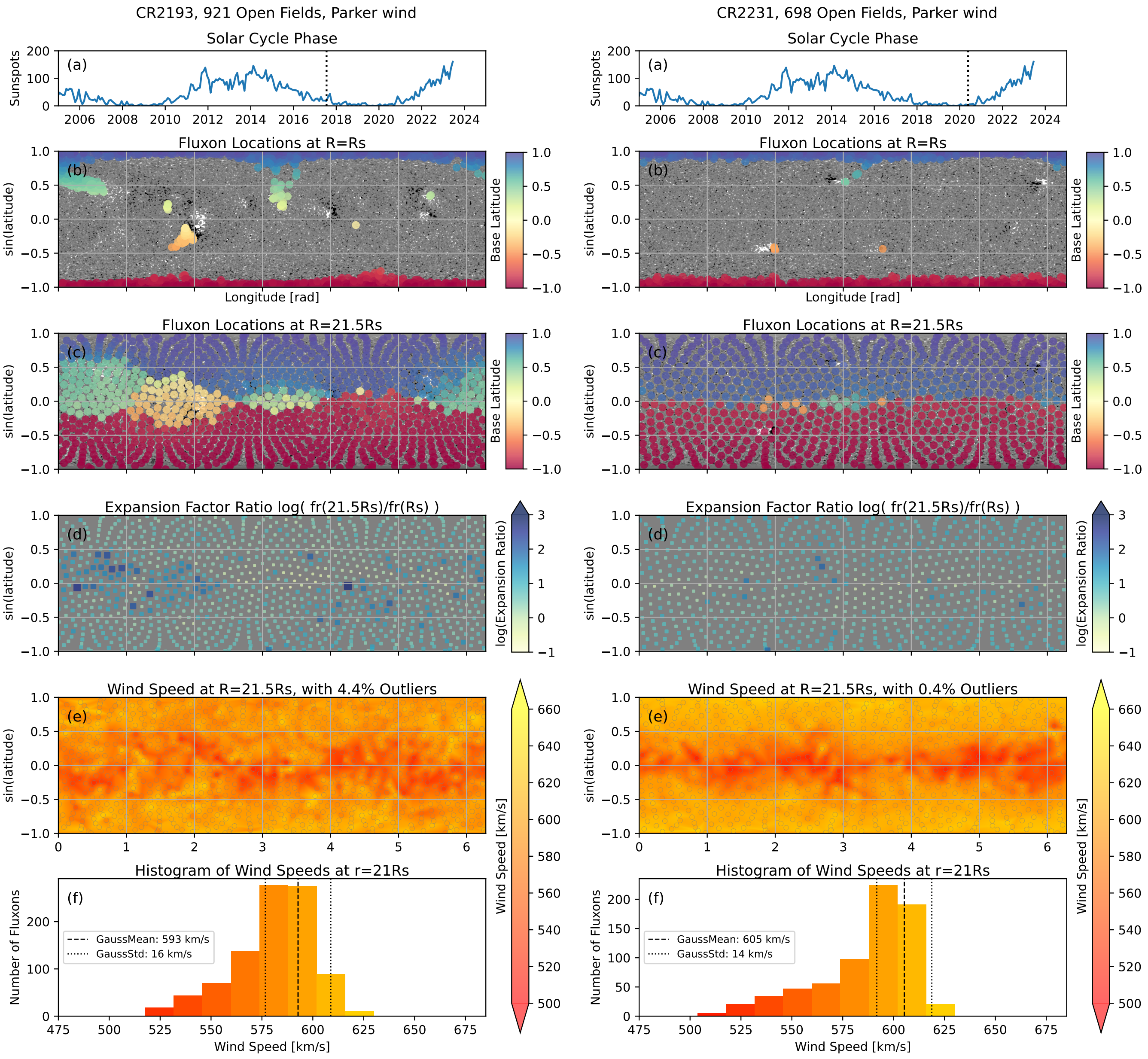

Figure 10. Results of FLUXPipe on Carrington rotations CR2193 and CR2231, before and after solar minimum. Panels are as described in Figure 9. As indicated in the top panel, CR2193 sits at the beginning of solar minimum, and CR2231 lies near the end of solar minimum.

Carrington rotation 2231, placed right as solar cycle 25 begins to ramp up, shows similar behavior to CR 2193. Most of the open fluxons are rooted towards the poles, with in this case fewer open fluxons towards the equatorial regions. Here we see the effects of some isolated patches of equatorial open field, but without large scale effects in the map of expansion factor. In both rotations 2193 and 2231, the dominant behavior of the outflow wind maps is a peak near each of the polar regions, with a progression towards a relatively slower meandering band of outflow wind speeds near the equator.

5. DISCUSSION

Fluxon modeling allows a unique approach to simulating the solar coronal magnetic field, yielding a number of advantages. The discretized nature of fluxon magnetic fieldlines allows for rapid assimilation, relaxation, and solar wind modeling. It avoids many of the grid-based issues that arise from numerical grid-based issues, such as numerical diffusion. Fluxons preserve the magnetic topology, unless magnetic reconnection is specifically triggered, preserving fieldline quantities such as winding.

The one-dimensional nature of fluxon magnetic fieldlines also provides the ideal case for analyzing the resulting solar wind acceleration. From the relaxed coronal magnetic configuration, each topologically open fluxon is a one-dimensional analogue of a bundle of open magnetic fieldlines. The spatial geometry, location of nearby neighbors, and magnetic expansion factor can be used to iteratively solve for a transonic solar wind solution.

Significant enhancements have been made towards leveraging FLUX as a pipeline tool for analyzing the solar wind, including the development of FLUXPipe for automated processing. The underlying code has been multithreaded, to allow for relaxation and solar wind computation more quickly.

5.1. *Code and data availability*

The FLUX code base is maintained as an open source GitHub repository, preserving the commit history of the code throughout its development lifetime. This contains installation instructions and a user-guide wiki set of documentation (Lowder et al. 2023). The codebase itself is a mixture of underlying C code doing the heavy computational lifting, and overarching Perl and Python code for interfacing with the simulation.

Given the unique nature of the fluxon modeling approach, sharing data is not always a direct process. Data export is possible to a regular uniform three-dimensional grid for sharing / comparisons with other model data. Solar wind data is also interpolated onto a two-dimensional spherical surface for interfacing with other wind models.

6. CONCLUSIONS

The FLUX model provides an intermediate approach between rapid heuristic methods and intensive 3D magnetohydrodynamic models, providing the best of both worlds. The FLUX model has the distinct advantages of being computationally efficient (scaling with the magnetic complexity of the two-dimensional photospheric boundary) and preserving connectivity to allow for tracking the history of a bundle of magnetic flux. Open fluxons extending from the photospheric boundary are used to compute a set of modified one-dimensional isothermal Parker solar wind solutions, with transonic solutions interpolated to an outer spherical boundary uniform grid at 21.5 $R_\odot$.

The FLUXPipe routine expands on this by providing an automated methodology for running a FLUX solar wind mapping from starting at input magnetogram assimilation to ending with output wind map gridding. This methodology was used to analyze four solar rotations throughout the solar cycle to catalog solar wind behavior.

6.1. *Future work*

While this manuscript focuses on laying out the fluxon methodology and coronal modeling framework, ongoing work will expand consideration to a further number of Carrington rotations throughout the solar cycle, with a followup manuscript in preparation. In particular a higher cadence of rotations allows for condensation of data into time-latitude maps to follow the ebb and flow in time and space of modeled solar wind outflows.

Other code enhancements in progress involve more complex / faster solar wind solutions, mass loading along fluxons, fluid pressure forces, and variable magnetic flux for each fluxon. For FLUX model solutions out to 21.5 $R_\odot$ (0.1 AU), the effects of the Parker Spiral may become an issue when linking open fieldlines down to their photospheric source. A Parker Spiral radially-dependent pseudoforce can be added to the set of force laws to consider this effect on mapping connectivity in more detail.

The authors gratefully acknowledge support of this work through NASA Heliophysics Living with a Star program grant 80NSSC17K0683, AFOSR grant FA9550-19-S-0003, and NASA Heliophysics Supporting Research grant NNH20ZDA001N. HMI Magnetic field maps are procured courtesy of NASA/SDO and the HMI science team. The authors also are particularly thankful for the insightful and helpful referee comments and suggestions.

Software: SunPy (Mumford et al. 2023; The SunPy Community et al. 2020), Astropy (Astropy Collaboration et al. 2013, 2018, 2022)

REFERENCES

Arge, C. N., Luhmann, J. G., Odstrcil, D., Schrijver, C. J., & Li, Y. 2004, Journal of Atmospheric and Solar-Terrestrial Physics, 66, 1295, doi: 10.1016/j.jastp.2004.03.018

Arge, C. N., Odstrcil, D., Pizzo, V. J., & Mayer, L. R. 2003, in American Institute of Physics Conference Series, Vol. 679, Solar Wind Ten, ed. M. Velli, R. Bruno, F. Malara, & B. Bucci, 190–193, doi: 10.1063/1.1618574

Arge, C. N., & Pizzo, V. J. 2000, J. Geophys. Res., 105, 10465, doi: 10.1029/1999JA000262

Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, A&A, 558, A33, doi: 10.1051/0004-6361/201322068

Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., et al. 2018, AJ, 156, 123, doi: 10.3847/1538-3881/aabc4f

Astropy Collaboration, Price-Whelan, A. M., Lim, P. L., et al. 2022, apj, 935, 167, doi: 10.3847/1538-4357/ac7c74

Cranmer, S. R. 2009, Living Reviews in Solar Physics, 6, 3, doi: 10.12942/lrsp-2009-3

DeForest, C. E., & Kankelborg, C. C. 2007, Journal of Atmospheric and Solar-Terrestrial Physics, 69, 116, doi: 10.1016/j.jastp.2006.06.011

Downs, C., Roussev, I. I., van der Holst, B., et al. 2010, ApJ, 712, 1219, doi: 10.1088/0004-637X/712/2/1219

Gibson, E. G. 1973, The quiet sun, Vol. 303

Hilbert, D. 1891, Mathematische Annalen, 38, 459, doi: 10.1007/BF01199431

Lowder, C., Lamb, D., Gilly, C. R., & DeForest, C. 2023, lowderchris/fluxon-mhd: v2.3, v2.3, Zenodo, doi: 10.5281/zenodo.7975952

Mackay, D. H., & Yeates, A. R. 2012, Living Reviews in Solar Physics, 9, 6, doi: 10.12942/lrsp-2012-6

Mumford, S. J., Freij, N., Stansby, D., et al. 2023, SunPy, v4.1.5, Zenodo, doi: 10.5281/zenodo.7850372

Parker, E. N. 1958, The Astrophysical Journal, 128, 664, doi: 10.1086/146579

Poduval, B., & Zhao, X. P. 2014, ApJL, 782, L22, doi: 10.1088/2041-8205/782/2/L22

Rachmeler, L. A., DeForest, C. E., & Kankelborg, C. C. 2009, ApJ, 693, 1431, doi: 10.1088/0004-637X/693/2/1431

Rachmeler, L. A., Pariat, E., DeForest, C. E., Antiochos, S., & Török, T. 2010, ApJ, 715, 1556, doi: 10.1088/0004-637X/715/2/1556

Rempel, M., Schüssler, M., & Knölker, M. 2009, ApJ, 691, 640, doi: 10.1088/0004-637X/691/1/640

Stansby, D., Yeates, A., & Badman, S. T. 2020, Journal of Open Source Software, 5, 2732, doi: 10.21105/joss.02732

Suess, S. T., Poletto, G., Wang, A. H., Wu, S. T., & Cuseri, I. 1998, SoPh, 180, 231, doi: 10.1023/A:1005001618698

The SunPy Community, Barnes, W. T., Bobra, M. G., et al. 2020, The Astrophysical Journal, 890, 68, doi: 10.3847/1538-4357/ab4f7a

Wang, Y. M. 1993, ApJL, 410, L123, doi: 10.1086/186895

Wang, Y. M., & Sheeley, N. R., J. 1990, ApJ, 355, 726, doi: 10.1086/168805

Zirker, J. B. 1977, Reviews of Geophysics and Space Physics, 15, 257, doi: 10.1029/RG015i003p00257