



Real-time Extrapolation of Nonlinear Force-free Fields from Photospheric Vector Magnetic Fields Using a Physics-informed Neural Operator

Mingyu Jeon¹, Hyun-Jin Jeong^{2,1}, Yong-Jae Moon^{1,3}, Jihye Kang³, and Kanya Kusano⁴

¹ School of Space Research, Kyung Hee University, Yongin, Republic of Korea; moonjy@khu.ac.kr

² Centre for mathematical Plasma Astrophysics, KU Leuven, Leuven, Belgium

³ Department of Astronomy and Space Science, Kyung Hee University, Yongin, Republic of Korea

⁴ Institute for Space-Earth Environmental Research, Nagoya University, Nagoya, Japan

Received 2024 October 20; revised 2025 February 19; accepted 2025 February 25; published 2025 April 1

Abstract

In this study, we develop a physics-informed neural operator (PINO) model that learns the solution operator from 2D photospheric vector magnetic fields to 3D nonlinear force-free fields (NLFFFs). We train our PINO model using physics loss from NLFFF partial differential equations, as well as data loss from target NLFFFs. We validate our method using an analytical NLFFF model. We then train and evaluate our PINO model with 2327 numerical NLFFFs of 211 active regions from the Institute for Space-Earth Environmental Research database. The results show that our trained PINO model can generate an NLFFF within 1 s for any active region on a single consumer GPU, making real-time extrapolation of NLFFFs possible. Our artificial intelligence (AI)-generated NLFFFs are qualitatively and quantitatively quite similar to the target NLFFFs for 30 active regions. The magnetic energy of the AI-generated NLFFFs of active region 11158 follows a similar trend to the target NLFFFs as well as other conventional methods.

Unified Astronomy Thesaurus concepts: Solar magnetic fields (1503); Solar active regions (1974); The Sun (1693); Neural networks (1933)

1. Introduction

Solar coronal magnetic fields play a key role in solar eruptions such as solar flares and coronal mass ejections (P. F. Chen 2011; K. Shibata & T. Magara 2011). These energetic events can cause adverse space weather impacts on telecommunications and satellites around the Earth (M. Temmer 2021; J. Zhang et al. 2021). It is essential to understand the 3D coronal magnetic fields, but high spatio-temporal resolution measurements of vector magnetic fields are currently restricted to the photosphere (T. Wiegmann et al. 2017). Nonlinear force-free field (NLFFF) extrapolations have been widely used for coronal magnetic field modeling, where we assume that nonmagnetic forces are negligible and the Lorentz force vanishes because the corona is a low plasma β region (T. Wiegmann et al. 2017; T. Wiegmann & T. Sakurai 2021). This requires solving partial differential equations (PDEs), defined as

$$(\nabla \times \mathbf{B}) \times \mathbf{B} = \mathbf{0}, \quad (1)$$

$$\nabla \cdot \mathbf{B} = 0, \quad (2)$$

where \mathbf{B} is a 3D NLFFF. We use an observed photospheric vector magnetic field as a bottom boundary condition. The various methods, such as optimization methods (M. S. Wheatland et al. 2000; T. Wiegmann et al. 2012) and magnetohydrodynamic (MHD) relaxation methods (Z. Mikić & A. N. McClymont 1994; S. Inoue et al. 2013), have been suggested for NLFFF extrapolation.

Although an NLFFF represents a static magnetic field, it is possible to approximately track the temporal evolution of

coronal magnetic fields by calculating a series of NLFFFs from a temporal sequence of observed photospheric vector magnetic fields. By analyzing a series of NLFFFs, K. Kusano et al. (2020) introduced the κ -scheme, a physics-based flare prediction model through MHD instability triggered by magnetic reconnection. This suggests that real-time computing of a series of NLFFFs would be useful for understanding and predicting solar eruptions. However, conventional methods are time consuming; they typically take several hours to compute a single high-resolution NLFFF. A new method is needed for real-time extrapolation of high-resolution 3D coronal magnetic fields of active regions (ARs).

Recently, deep-learning methods have been used for the reconstruction of 3D magnetic fields in the solar corona (for a review, see A. Asensio Ramos et al. 2023). R. Jarolim et al. (2023) applied physics-informed neural networks (PINNs); M. Raissi et al. (2019) to calculate NLFFFs of ARs from photospheric magnetic fields, and R. Jarolim et al. (2024) extended this approach to use both photospheric and chromospheric magnetic fields via height-mapping models. They used a transfer-learning approach so that the computation time of the NLFFF for the first time step was about 1 hr, but the NLFFF of the next time step was calculated within 1 minute. The PINNs need to be retrained for new ARs because the inputs are coordinates in the computational domain. S. Rahman et al. (2024) generated global 3D solar coronal magnetic field, radial velocity, and temperature using pix2pixCC (H.-J. Jeong et al. 2022). They used synoptic maps of photospheric magnetic fields as inputs and simulation data of the MHD algorithm outside a sphere (J. A. Linker et al. 1999; Z. Mikić et al. 1999) as targets. Since the inputs are preprocessed observation data, they can use trained models for near real-time computation of the global 3D solar coronal parameters.

In this study, we consider NLFFF extrapolation as a parametric PDE problem where the parameter is a photospheric



Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](https://creativecommons.org/licenses/by/4.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

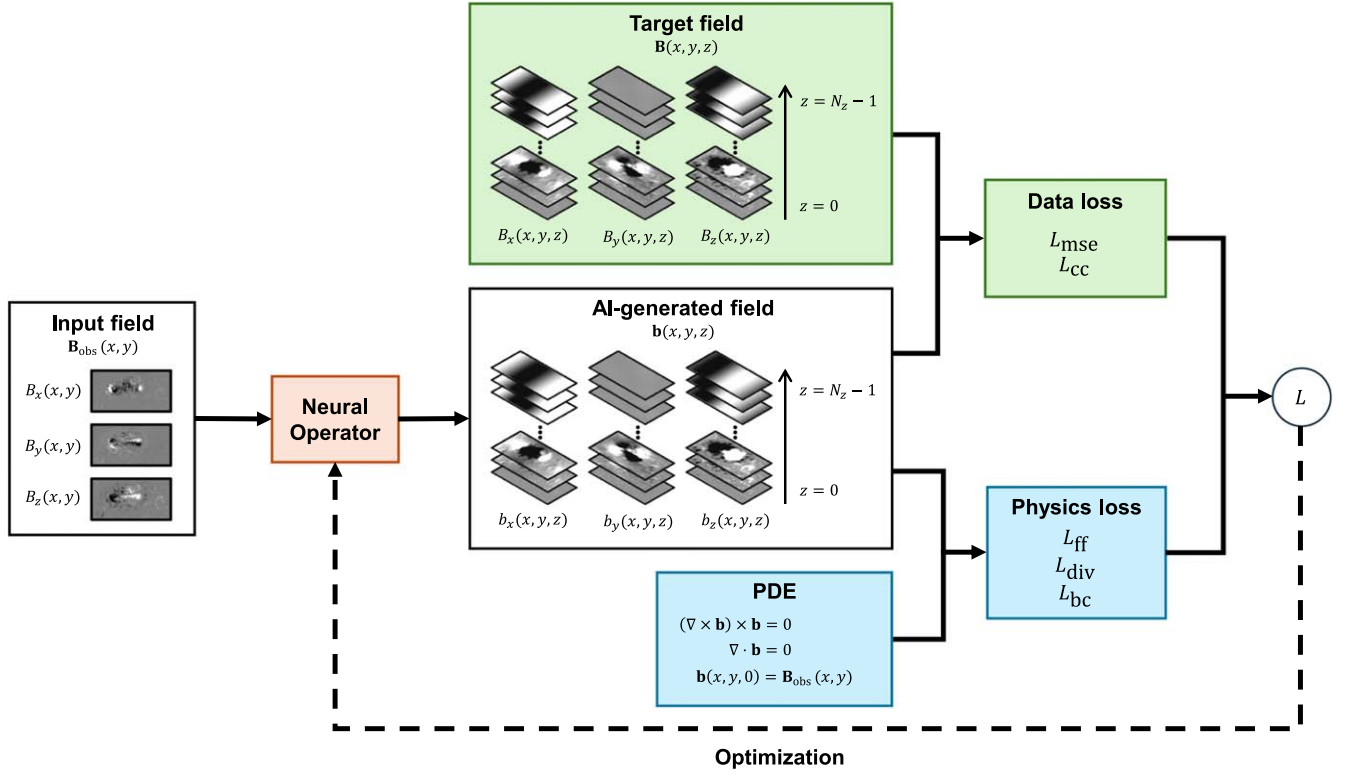


Figure 1. Overview of our method. The white and black in the images indicate positive and negative polarities, respectively, with the color map range defined as $\pm(1000 \text{ G})/(z + 1)$. The neural operator, shown as an orange box, is a U-shaped neural operator with six Fourier layers. Here, L_{mse} is a mean-squared error loss, L_{cc} is a correlation coefficient loss, L_{ff} is a force-free loss, L_{div} is a divergence-free loss, and L_{bc} is a bottom boundary condition loss.

vector magnetic field. We use a physics-informed neural operator (PINO) model (Z. Li et al. 2024) for real-time extrapolation of NLFFFs using physics losses provided by PDEs (1) and (2), as well as data loss. Our PINO model can learn the solution operator from 2D photospheric vector magnetic fields to 3D NLFFFs. To validate our method, we train and evaluate our PINO model with a semianalytical NLFFF model (B. C. Low & Y. Q. Lou 1990). Then, we train and evaluate our PINO model with numerical NLFFFs from the Institute for Space-Earth Environmental Research (ISEE) database (K. Kusano et al. 2021) for real-time extrapolations of NLFFFs of solar ARs. The performance of our trained PINO may degrade when applied to ARs with out-of-distribution properties that differ significantly from those in the ISEE data set. We explain our method in Section 2. In Section 3, we describe the data used for our model. In Section 4, we show the results and discuss them. We summarize our study in Section 5.

2. Method

2.1. Physics-informed Neural Operator

We use a PINO model (Z. Li et al. 2024) for real-time extrapolation of NLFFFs from photospheric vector magnetic fields. Neural operators refer to neural networks that can learn nonlinear operators of parametric PDE problems (N. B. Kovachki et al. 2021; S. A. Faroughi et al. 2024). The two main neural operator architectures are Deep Operator Network (DeepONet), introduced by L. Lu et al. (2021, 2022), and Fourier Neural Operator (FNO), proposed by Z. Li et al. (2020). DeepONet and FNO were independently developed with different viewpoints on operator learning, but both show comparable performance for relatively simple problems (L. Lu et al. 2022). Although neural

operators typically depend solely on input-output pairs for training, incorporating physics information into them offers the combined benefits of neural operators (e.g., real-time computation through inference without retraining) and PINNs (e.g., compatibility with underlying physics through regularization of physics-informed loss) (G. E. Karniadakis et al. 2021; S. Wang et al. 2021; S. Goswami et al. 2023; S. A. Faroughi et al. 2024; Z. Li et al. 2024). We calculate physics loss based on PDEs (1) and (2), as well as data loss using pairs of our artificial intelligence (AI)-generated NLFFFs and target ones.

Figure 1 shows an overview of our method. The architecture of our PINO model is a U-shaped neural operator architecture (M. A. Rahman et al. 2022) with six Fourier layers implemented in the neural operator library (Z. Li et al. 2020; N. B. Kovachki et al. 2021). It has useful features like enabling deeper neural operators, fast training, data efficiency, and robustness across various hyperparameter selections (M. A. Rahman et al. 2022). Both the lifting block and projection block have 256 hidden channels. The initial width of the U-shaped neural operator is 32. The numbers of output channels for each Fourier layer are 32, 64, 64, 64, 64, and 32. The numbers of Fourier modes are 16, 8, 8, 8, 8, and 16. The scaling factors are 1.0, 0.5, 1.0, 1.0, 2.0, and 1.0. Details about neural operator theory and the U-shaped neural operator can be found in M. A. Rahman et al. (2022) and Z. Li et al. (2024) and references therein.

Our PINO model generates a 3D NLFFF $\mathbf{b}(x, y, z)$ with a shape $N_x \times N_y \times N_z \times 3$ from a photospheric vector magnetic field $\mathbf{B}_{\text{obs}}(x, y)$ with a shape $N_x \times N_y \times 1 \times 3$, where 3 represents the number of vector components, 1 is the number of input channels, and N_z is the number of output channels. We put the bottom boundary on the $z=0$ plane and the top boundary on the $z = N_z - 1$ plane. Since the magnetic field

strength decreases with the height from the bottom boundary, we normalize the target field \mathbf{B} by dividing it by a factor $B_0/(z+1)$, which is empirically determined by several trials. Here, B_0 is a typical magnetic field strength of an AR, and we use $B_0 = 200$ G for the Low and Lou NLFFFs (B. C. Low & Y. Q. Lou 1990) and $B_0 = 2500$ G for the ISEE NLFFFs (K. Kusano et al. 2021). This scaling is applied only for the calculation of data losses; it does not apply when calculating physics losses to ensure accurate derivative calculations.

Our PINO model uses data loss and physics loss during training. The data loss is calculated by comparing our AI-generated field \mathbf{b} with the target field \mathbf{B} using a mean-squared error loss L_{mse} and a correlation coefficient loss L_{cc} , which is a mean-squared error between 1 and the concordance correlation coefficient. We calculate the data losses for each layer of a constant z and average them. For the physics loss, we calculate the PDE loss and bottom boundary condition loss. The PDE loss is calculated using the force-free loss L_{ff} and the divergence-free loss L_{div} , which are computed using the PDEs (1) and (2), respectively. The derivatives are calculated using a second-order finite difference scheme. We assume a uniform grid spacing of length 1. The bottom boundary condition loss L_{bc} is a mean-squared error between output data $\mathbf{b}(x, y, 0)$ and input data $\mathbf{B}_{\text{obs}}(x, y)$. The formulae for the losses are as follows:

$$L_{\text{mse}} = \frac{1}{N_z} \sum_z \langle |\mathbf{b}(z) - \mathbf{B}(z)|^2 \rangle, \quad (3)$$

$$L_{\text{cc}} = \frac{1}{N_z} \sum_z \langle |1 - \text{cc}(\mathbf{b}(z), \mathbf{B}(z))|^2 \rangle, \quad (4)$$

$$L_{\text{ff}} = \left\langle \frac{|(\nabla \times \mathbf{b}) \times \mathbf{b}|^2}{|\mathbf{b}|^2 + \epsilon} \right\rangle, \quad (5)$$

$$L_{\text{div}} = \langle |\nabla \cdot \mathbf{b}|^2 \rangle, \quad (6)$$

$$L_{\text{bc}} = \langle |\mathbf{b}(z=0) - \mathbf{B}_{\text{obs}}|^2 \rangle, \quad (7)$$

where $\langle \cdot \rangle$ denotes the average over the grid points, and $\epsilon = 10^{-7}$ is a small number to avoid division by zero. To calculate the concordance correlation coefficient, we flatten the magnetic field data $\mathbf{b}(z)$ and $\mathbf{B}(z)$ for each layer of a constant z and apply the following formula:

$$\text{cc}(x, y) = \frac{2\rho\sigma_x\sigma_y}{\sigma_x^2 + \sigma_y^2 + (\mu_x - \mu_y)^2}, \quad (8)$$

where σ_x and σ_y are the corresponding standard deviations, μ_x and μ_y are the corresponding means, and ρ is the Pearson correlation coefficient between the two variables x and y , given by $\rho(x, y) = \text{cov}(x, y)/(\sigma_x\sigma_y)$, where $\text{cov}(x, y)$ is the covariance of the two variables. We use `torchmetrics` (N. S. Detlefsen et al. 2022) to calculate the data losses.

The total loss is the weighted sum of these losses:

$$L = w_{\text{mse}}L_{\text{mse}} + w_{\text{cc}}L_{\text{cc}} + w_{\text{ff}}L_{\text{ff}} + w_{\text{div}}L_{\text{div}} + w_{\text{bc}}L_{\text{bc}}, \quad (9)$$

where we use $w_{\text{mse}} = 1$, $w_{\text{cc}} = 0.01$, $w_{\text{ff}} = 100$, $w_{\text{div}} = 100$, and $w_{\text{bc}} = 10$. These values are empirically determined after several trials. The effect of changing the weights of physics losses is discussed for the semianalytical Low and Lou field (B. C. Low & Y. Q. Lou 1990) in Section 4. To minimize the total loss, we use an Adam optimizer (D. P. Kingma & J. Ba 2014) with a learning rate of 10^{-5} . We use a batch size of one, and each batch consists of a pair comprising single vector

magnetic field data on the bottom boundary and the entire volume data of the corresponding 3D NLFFF. The total number of epochs is set to 100. At the end of each epoch, we calculate the loss using the validation set and select the model with the smallest validation loss across the entire epoch. The composition of the training, validation, and test sets is described in Section 3. Training and inference are performed on an NVIDIA RTX 4070 Ti GPU. The source code used in this study is available at <https://github.com/mgjeon/rmag>.

2.2. Evaluation Metrics

We use the metrics described in M. S. Wheatland et al. (2000) and C. J. Schrijver et al. (2006) to evaluate the performance of our PINO model. The magnetic field vectors at grid point i are represented as \mathbf{b}_i for our AI-generated field and \mathbf{B}_i for the target field. The symbol N denotes the total number of grid points in the computational volume. The following five comparison metrics compare our AI-generated field \mathbf{b} with the target field \mathbf{B} :

$$C_{\text{vec}} = \frac{\sum_i \mathbf{b}_i \cdot \mathbf{B}_i}{(\sum_i |\mathbf{b}_i|^2 \sum_i |\mathbf{B}_i|^2)^{1/2}}, \quad (10)$$

$$C_{\text{CS}} = \frac{1}{N} \sum_i \frac{\mathbf{b}_i \cdot \mathbf{B}_i}{|\mathbf{b}_i| |\mathbf{B}_i|}, \quad (11)$$

$$E_n = \frac{\sum_i |\mathbf{b}_i - \mathbf{B}_i|}{\sum_i |\mathbf{B}_i|}, \quad (12)$$

$$E_m = \frac{1}{N} \sum_i \frac{|\mathbf{b}_i - \mathbf{B}_i|}{|\mathbf{B}_i|}, \quad (13)$$

$$\epsilon = \frac{\sum_i |\mathbf{b}_i|^2}{\sum_i |\mathbf{B}_i|^2}, \quad (14)$$

where C_{vec} is the vector correlation metric, C_{CS} is the Cauchy–Schwartz metric, E_n is the normalized vector error, E_m is the mean vector error, and ϵ is the ratio of the magnetic energy of our AI-generated field \mathbf{b} to the target field \mathbf{B} . We use $E'_n = 1 - E_n$ and $E'_m = 1 - E_m$ such that all comparison metrics have a value of 1 when \mathbf{b} and \mathbf{B} are identical.

The following two quality metrics measure the force freeness and divergence freeness of a field, \mathbf{B} :

$$\sigma_J = \left(\sum_i \frac{|\mathbf{J}_i \times \mathbf{B}_i|}{|\mathbf{B}_i|} \right) / \left(\sum_i |\mathbf{J}_i| \right), \quad (15)$$

$$\langle |f_i| \rangle = \frac{1}{N} \sum_i \frac{|\nabla \cdot \mathbf{B}_i| \Delta V_i}{|\mathbf{B}_i| A_i}, \quad (16)$$

where $\mathbf{J} = \nabla \times \mathbf{B}/4\pi$ is the current density in Gaussian units, σ_J is the current-weighted average of the sine of the angle between the magnetic field and current density, and $\langle |f_i| \rangle$ is the average value of the magnitude of the fractional flux increase with a surface area A_i of a small volume ΔV_i around grid point i (M. S. Wheatland et al. 2000). For a rectangular uniform grid with cell sizes δx , δy , and δz in each direction, the volume ΔV_i is given by $\Delta V_i = \delta x \delta y \delta z$, and the surface area A_i is given by $A_i = 2(\delta x \delta y + \delta y \delta z + \delta z \delta x)$. The quality metrics σ_J and $\langle |f_i| \rangle$ have a value of 0 when \mathbf{B} is perfectly force free and divergence free.

Table 1
Quantitative Comparison between Target NLFFFs and AI-generated NLFFFs

	C_{vec}	C_{CS}	E'_n	E'_m	ϵ	σ_J	$\langle f_i \rangle$
Test case for semianalytical NLFFFs ($l = 0.3$, $\Phi = 0.25\pi$)	1.00	1.00	1.00	1.00	1.00	0.02	3.3×10^{-5}
AI generated ($w_{\text{ff}/\text{div}} = 100$)	1.00	0.98	0.91	0.81	1.01	0.24	5.8×10^{-3}
AI generated ($w_{\text{ff}} = 1000$)	1.00	0.96	0.89	0.77	1.01	0.21	7.4×10^{-3}
AI generated ($w_{\text{ff}} = 10$)	1.00	0.98	0.91	0.83	1.01	0.26	5.4×10^{-3}
AI generated ($w_{\text{div}} = 1000$)	1.00	0.96	0.89	0.78	1.01	0.25	4.4×10^{-3}
AI generated ($w_{\text{div}} = 10$)	1.00	0.98	0.91	0.83	1.02	0.24	6.2×10^{-3}
AI generated ($w_{\text{ff}/\text{div}} = 0$)	1.00	0.98	0.92	0.87	1.01	0.29	5.8×10^{-3}
AI ($w_{\text{ff}/\text{div}} = 100$) + optimization method	1.00	0.96	0.87	0.71	1.00	0.08	2.4×10^{-3}
Test set for numerical NLFFFs (483 NLFFFs of 30 ARs)	1.00	1.00	1.00	1.00	1.00	0.83	4.3×10^{-4}
AI generated ($w_{\text{ff}/\text{div}} = 100$)	0.96	0.94	0.70	0.58	0.93	0.85	5.3×10^{-3}
AI generated ($w_{\text{ff}/\text{div}} = 0$)	0.94	0.93	0.67	0.55	0.90	0.84	7.5×10^{-3}
AI ($w_{\text{ff}/\text{div}} = 100$) + optimization method	0.97	0.94	0.72	0.58	0.96	0.47	1.5×10^{-3}

Note. The comparison metrics C_{vec} , C_{CS} , E'_n , E'_m , and ϵ of AI-generated NLFFFs equal to 1 when the target and AI-generated NLFFFs are identical. The quality metrics σ_J and $\langle |f_i| \rangle$ are equal to 0 when the field is perfectly force free and divergence free. The first rows of each section show the metrics of the target NLFFFs, and the other rows show the metrics of the AI-generated NLFFFs.

3. Data

For the analytical test case, we use the Low and Lou semianalytical NLFFF model (B. C. Low & Y. Q. Lou 1990). The field is determined by four parameters: n , m , l , and Φ . We use the Low and Lou NLFFFs with $n = 1$ and $m = 1$. We split training, validation, and test sets based on l and Φ . We select the Low and Lou NLFFF with $l = 0.3$ and $\Phi = 0.25\pi$ as a test field. We calculate 6000 Low and Lou NLFFFs with randomly selected l and Φ parameters with the ranges $l \in [0.15, 0.25] \cup [0.35, 0.45]$ and $\Phi \in [0, 0.2\pi] \cup [0.3\pi, \pi]$, ensuring that the test field is excluded from the training and validation sets. These fields are randomly divided into 5000 fields for the training set and 1000 fields for the validation set. These fields have a resolution of $N_x \times N_y \times N_z = 64 \times 64 \times 64$.

To train our PINO model for more complex NLFFFs, we use 2327 numerical NLFFFs from the ISEE database for NLFFFs of solar ARs (K. Kusano et al. 2021), which consists of 211 ARs from 2010 to 2022. These NLFFFs are not entirely independent of each other because NLFFFs extrapolated from the same ARs at different time steps are included. This database includes 3D NLFFFs extrapolated from photospheric vector magnetic fields by the MHD relaxation method (S. Inoue et al. 2013), with a resolution of $N_x \times N_y \times N_z = 512 \times 256 \times 256$. The photospheric vector magnetic fields are obtained from the Space weather Helioseismic and Magnetic Imager (HMI; P. H. Scherrer et al. 2011) Active Region Patch (SHARP) data remapped to a Lambert cylindrical equal-area (CEA) projection (M. G. Bobra et al. 2014). The bottom boundary data for the ISEE NLFFFs are the SHARP data only resized to $N_x \times N_y = 512 \times 256$ using the IDL CONGRID function, regardless of the size of the ARs, and without any further preprocessing described in T. Wiegmann et al. (2006). The physical size of each pixel varies for NLFFFs from different ARs in the ISEE database. Since this data set uses only SHARP data, we require vector magnetic field data processed according to the SHARP pipeline for the application of our trained PINO model. The physical sizes of the ARs range from 78 Mm \times 44 Mm (AR 11736) to 943 Mm \times 413 Mm (AR 12242). The length in the x -direction has a mean of 346 Mm, a median of 314 Mm, and a standard deviation of 139 Mm. The length in the y -direction has a mean of 168 Mm, a median of 160 Mm, and a standard deviation of 60 Mm. We

split the NLFFFs of the ISEE data set into training, validation, and test sets based on ARs because the NLFFFs of the same AR are not independent. We include AR 11158 and 12673 in the test set, and the other ARs are manually selected to ensure that the training, validation, and test sets cover the similar time period. Our training set consists of 1795 NLFFFs of 151 ARs (from AR 11078 to 12975), and our validation and test sets contain 49 NLFFFs of 30 ARs (from AR 11846 to 12975) and 483 NLFFFs of other 30 ARs (from AR 11158 to 12673), respectively. The list of ARs for the data sets is provided in the Appendix. The number of NLFFFs for each AR varies from one to 390. For more details, please refer to the ISEE database (K. Kusano et al. 2021). For data augmentation, we rotate the NLFFF by 180° around the z -axis. To do that, we reverse the order of the NLFFF data array along the x - and y -axis using the `torch.flip` function and change the sign of the x - and y -components of the NLFFF.

4. Results and Discussion

Table 1 presents the quantitative comparison between the target NLFFFs and the AI-generated NLFFFs. The target NLFFFs include the test case for semianalytical NLFFFs and the test set for numerical NLFFFs. The test case for semianalytical NLFFFs is the Low and Lou semianalytical field (B. C. Low & Y. Q. Lou 1990) with $l = 0.3$ and $\Phi = 0.25\pi$, which is not used to train the PINO model. The test set for numerical NLFFFs consists of 483 numerical NLFFFs of 30 ARs (from AR 11158 to 12673) in the ISEE database (K. Kusano et al. 2021). We select the model based on the loss given by Equation (9) for the validation set, not based on the metrics given by Equations (10)–(16). The AI-generated NLFFFs are the output data of our trained PINO model. One PINO model is used for the semianalytical fields and the other one for the numerical fields. For the metrics of numerical fields, we first calculate the average metrics for each AR and then take the overall average of these individual averages for all ARs.

The semianalytical field section of Table 1 shows that the comparison metrics C_{vec} , C_{CS} , E'_n , E'_m , and ϵ of our AI-generated field are 1.00, 0.98, 0.91, 0.81, and 1.01, respectively. These values are quite close to 1, demonstrating that our trained PINO model can generate an NLFFF that closely matches the target semianalytical NLFFF. However, the

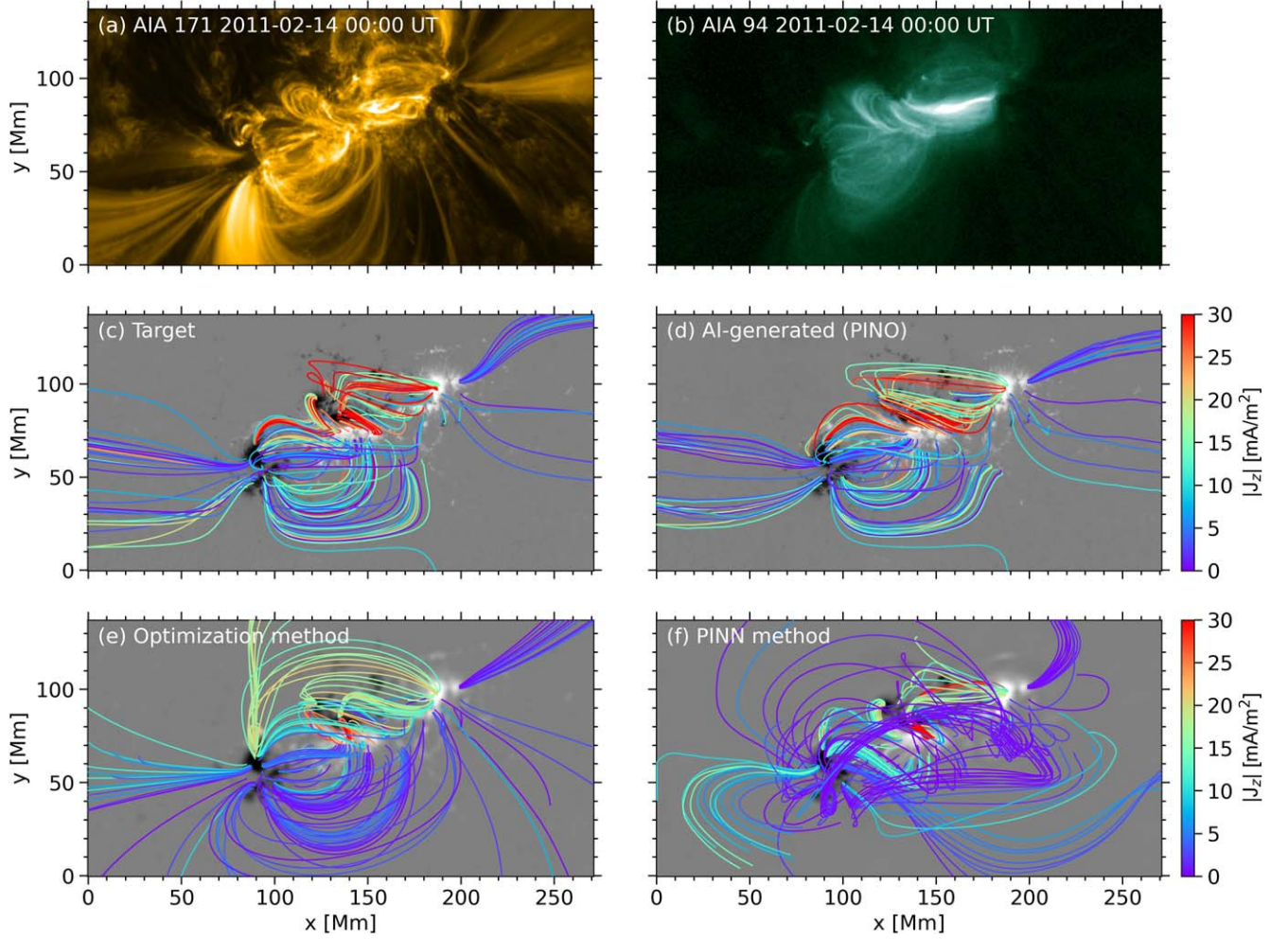


Figure 2. EUV observations of AR 11158 and projected magnetic field lines from different extrapolation methods. The color of the field lines indicates the magnitude of the vertical current density in the photosphere. EUV observations in 171 and 94 Å are shown in (a) and (b), respectively. (c) Target magnetic field of the ISEE NLFFF database. (d) AI-generated magnetic field of our trained PINO model. (e) Magnetic field of the optimization method (T. Wiegelmann et al. 2012). (f) Magnetic field of the PINN method ($\lambda_{\text{ff/div}} = 0.1$; R. Jarolim et al. 2023).

quality metrics σ_J and $\langle |f_i| \rangle$ are 0.24 and 5.8×10^{-3} , respectively, which are not as good compared to the values of 0.02 and 3.3×10^{-5} for the target semianalytical field. This indicates that, even with physics-informed losses, ensuring the force freeness and divergence freeness of NLFFFs generated by a single PINO model is challenging. We also investigate the effect of varying the weights of the physics losses, w_{ff} and w_{div} . The default values are $w_{\text{ff}} = 100$ and $w_{\text{div}} = 100$. We vary either w_{ff} or w_{div} to 1000 or 10 while keeping all other weights fixed. As expected, increasing w_{ff} or w_{div} reduces σ_J and $\langle |f_i| \rangle$, respectively, whereas decreasing w_{ff} or w_{div} increases the values of the quality metrics. However, higher values of $w_{\text{ff/div}}$ tend to worsen the comparison metrics. In comparison, lower values yield slightly better or comparable comparison metrics relative to $w_{\text{ff/div}} = 100$ but give worse quality metrics, either σ_J or $\langle |f_i| \rangle$. The comparison metrics without physics losses are better than those with physics losses; however, the quality metrics are worse or comparable. Therefore, the effect of the physics losses is not very significant, though there is a positive effect on σ_J for the semianalytical field. Based on these, we select $w_{\text{ff}} = w_{\text{div}} = 100$, as mentioned in Section 2.

Our approach is valid for the numerical field case as well as for the semianalytical field case. In Table 1, the comparison metrics

C_{vec} , C_{CS} , E'_n , E'_m , and ϵ of our AI-generated NLFFFs are 0.96, 0.94, 0.70, 0.58, and 0.93 for the test set, respectively. Our AI-generated NLFFFs show good performance on average, especially considering that these NLFFFs are generated within 1 s using only the observed photospheric vector magnetic fields. Our results indicate that the proposed approach is an efficient alternative for rapid coronal magnetic field reconstruction, given the inherent uncertainties in the force-free assumption itself and existing NLFFF extrapolation methods. The results from the test set demonstrate how well our PINO model performs on new ARs that are not used for model training or selection. Our PINO model for numerical fields uses the SHARP vector magnetic field data at the bottom of a 3D Cartesian box, so the model can only be used with AR-sized photospheric vector magnetic field data remapped to the Lambert CEA projection following the SHARP pipeline (M. G. Bobra et al. 2014). The quality metric for divergence freeness $\langle |f_i| \rangle$ of our AI-generated NLFFFs from the PINO model with physics losses ($w_{\text{ff/div}} = 100$) is 5.3×10^{-3} , which is better than 7.5×10^{-3} from the PINO model without physics losses ($w_{\text{ff/div}} = 0$). However, the quality metric for force freeness σ_J is 0.85 with physics losses and 0.84 without physics losses, indicating that both cases achieve a similar level of force freeness. This may be due to the difficulty of balancing multiobjective

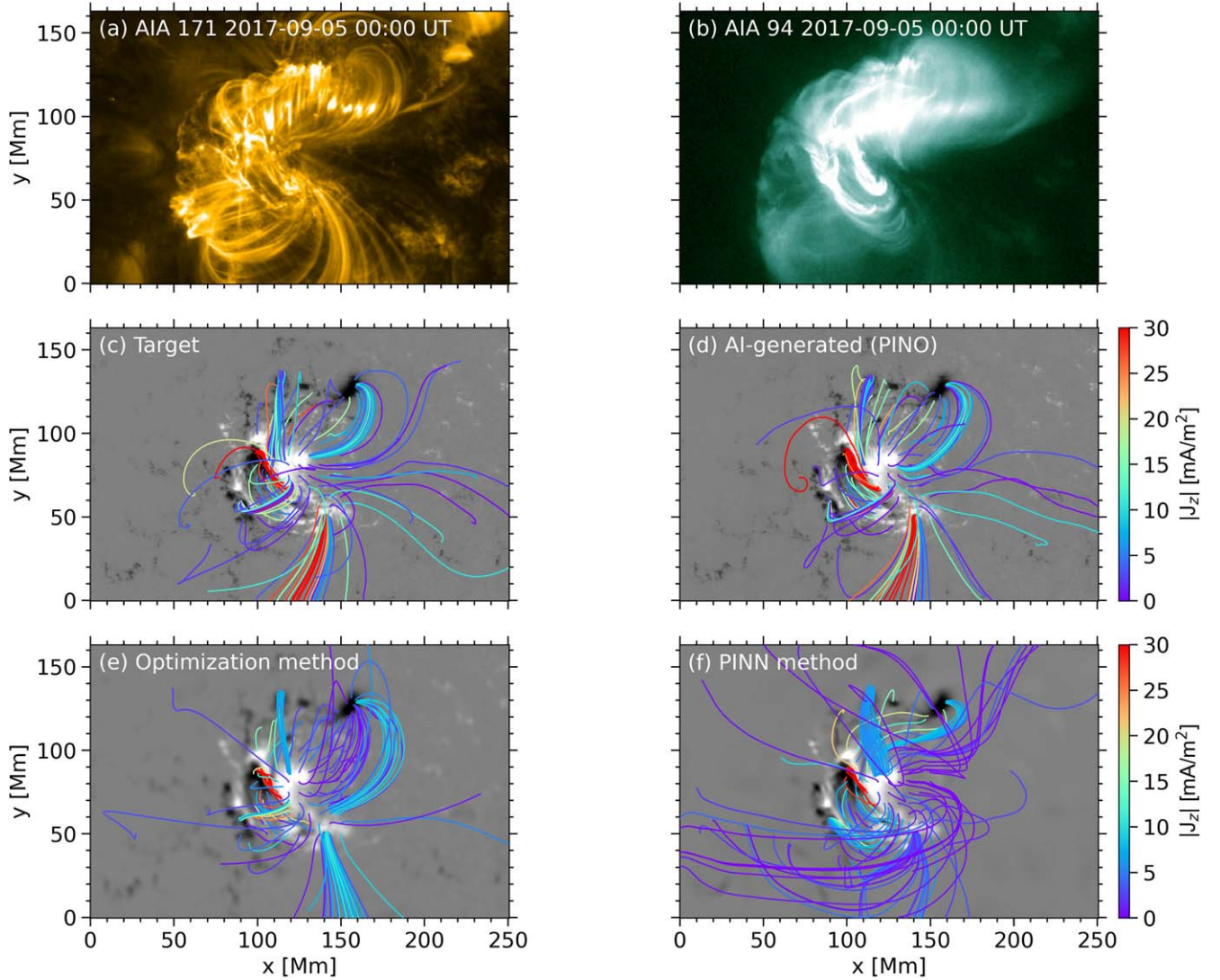


Figure 3. EUV observations of AR 12673 and projected magnetic field lines from different extrapolation methods. The format of the figure is the same as Figure 2.

losses. In the numerical field case, unlike in the semianalytical case, the comparison metrics C_{vec} , C_{CS} , E'_m , E'_m , and ϵ improve when physics losses are used. Without physics losses, these metrics are 0.94, 0.93, 0.67, 0.55, and 0.90 for the test set, respectively. This indicates that physics losses help generate NLFFFs of ARs that are more similar to the target NLFFFs obtained from the MHD relaxation method (S. Inoue et al. 2013).

In Figures 2 and 3, we compare the EUV observations and the magnetic field lines extrapolated from several methods for ARs 11158 and 12673, respectively. The field lines are traced using *streamtracer*.⁵ Panels (a) and (b) in both figures show the EUV observations by Solar Dynamics Observatory (SDO; W. D. Pesnell et al. 2012)/Atmospheric Imaging Assembly (AIA; J. R. Lemen et al. 2012) at 171 and 94 Å, respectively, which display the structures of coronal loops in these ARs. The panels from (c) to (f) show the projected magnetic field lines from the same footpoints on the photosphere from the following: (c) the target ISEE NLFFFs; (d) our AI-generated NLFFFs; (e) the NLFFFs from the optimization method (T. Wiegmann et al. 2012); and (f) the NLFFFs from the PINN method (R. Jarolim et al. 2023; $\lambda_{\text{ff/div}} = 0.1$).

The AI-generated NLFFFs by our trained PINO model in panels (d) of Figures 2 and 3 show similar overall shapes to the target NLFFFs in panels (c), which confirms the ability of our trained PINO model to successfully generate NLFFFs for these ARs that are quite qualitatively similar to the target NLFFFs. This is also consistent with the quantitative results in Table 1. The coronal loop structures visible in the 171 Å images show a good match with the magnetic field lines of our AI-generated NLFFFs. The alignment of the red high current density regions in our AI-generated NLFFFs with the bright regions observed in the 94 Å EUV images further supports the ability of our trained PINO model to generate quite accurate NLFFFs for the ARs.

The overall field line structures of our AI-generated NLFFFs are also similar to those by other extrapolation methods in panels (e) and (f) of Figures 2 and 3. Our PINO model shows qualitatively comparable performance to these conventional methods. The vertically integrated current density maps in Figures 4 and 5 provide an additional representation of the magnetic topology. The spatial distribution of the current density from our AI-generated NLFFFs in panels (d) is similar to those from the target NLFFFs in panels (c) and other extrapolation methods in panels (e) and (f). While conventional

⁵ <https://github.com/sunpy/streamtracer>

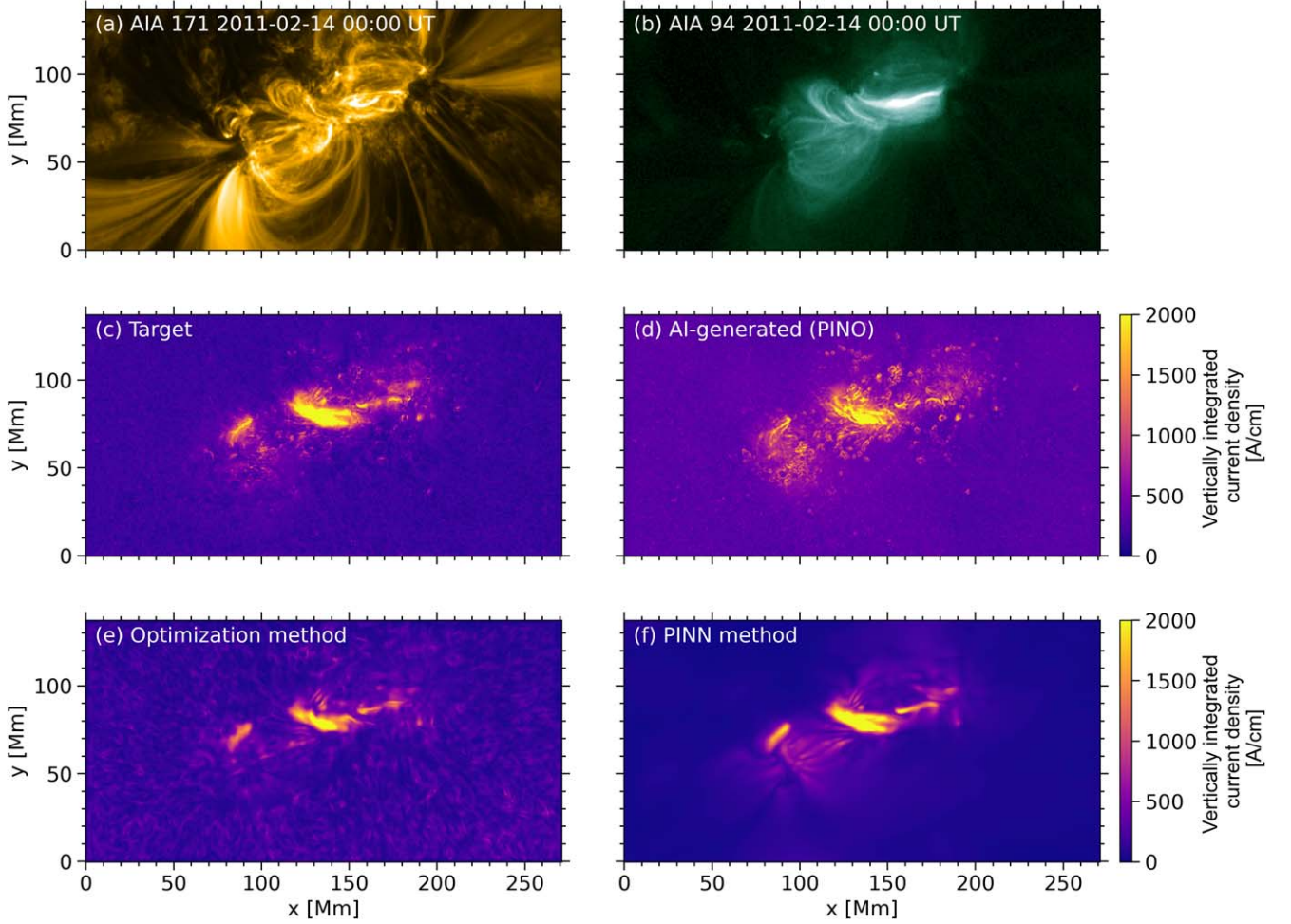


Figure 4. EUV observations of AR 11158 and vertically integrated current density maps from different extrapolation methods. The format of the figure is the same as Figure 2.

methods require several hours to compute a single high-resolution ($512 \times 256 \times 256$) NLFFF for an AR, our AI-generated NLFFFs can be obtained within 1 s on a single consumer GPU (NVIDIA RTX 4070 Ti) and within 2 s on a single consumer CPU (Intel i5-13600K). This significant speedup is achieved because our PINO model does not require retraining for new ARs.

Figure 6 shows the time series of total and free magnetic energy calculated from NLFFFs of AR 11158. Although the magnetic energy of our AI-generated NLFFFs is about 5% lower than that of the target fields, it follows a similar trend and aligns well with the magnetic energy obtained from the optimization method. Since our trained PINO model generates an NLFFF within 1 s, the total computation time for 601 SHARP data with the full cadence of 12 minutes for the five-day observation is less than 10 minutes. This shows the sufficient potential of our PINO model as a method for real-time solar coronal magnetic field modeling. The dashed black vertical lines indicate flares $\geq M1$ from 00:00 UTC on 2011 February 12 to 00:00 UTC on 2011 February 17. The free magnetic energy also shows trends similar to those obtained with other methods. While the drop in free magnetic energy at the X2.2 flare is captured by our AI-generated NLFFFs, the M6.6 flare is not well captured. This discrepancy may be due to the high divergence freeness of our AI-generated NLFFFs (G. Valori et al. 2013). This shows a limitation of our method:

a single PINO model may not be able to provide accurate NLFFFs for all boundary condition data.

There may be concerns about the force freeness and divergence freeness of our AI-generated NLFFFs, as indicated by their quality metrics compared to the target NLFFFs in Table 1. Attempts to resolve this by increasing the loss weights w_{ff} and w_{div} , or by using a vector potential, result in unstable training without significant improvements. Using preprocessed boundary conditions (T. Wiegmann et al. 2006) as an input for our trained PINO model also does not lead to significant improvement. This may be due to insufficient boundary conditions of NLFFF problems from using only the bottom boundary condition as an input. Another possible reason is that our PINO model does not accept coordinate inputs, unlike PINN, which can achieve excellent force freeness and divergence freeness through automatic differentiation. Also, without coordinate inputs, it is necessary to hold the entire volume data in GPU memory during training. This requires about 11 GB for numerical NLFFFs from the ISEE database. This is why we use a batch size of one, as we trained the model on an NVIDIA RTX 4070Ti with 12 GB of GPU memory. Using a subset of coordinate inputs for each batch, as in PINNs (R. Jarolim et al. 2023), could mitigate memory limitations for larger ARs or higher altitudes. However, our examinations with DeepONet, which uses coordinate inputs in operator learning, show several challenges: overly smooth bottom boundary fields; worsening comparison metrics; and increasing

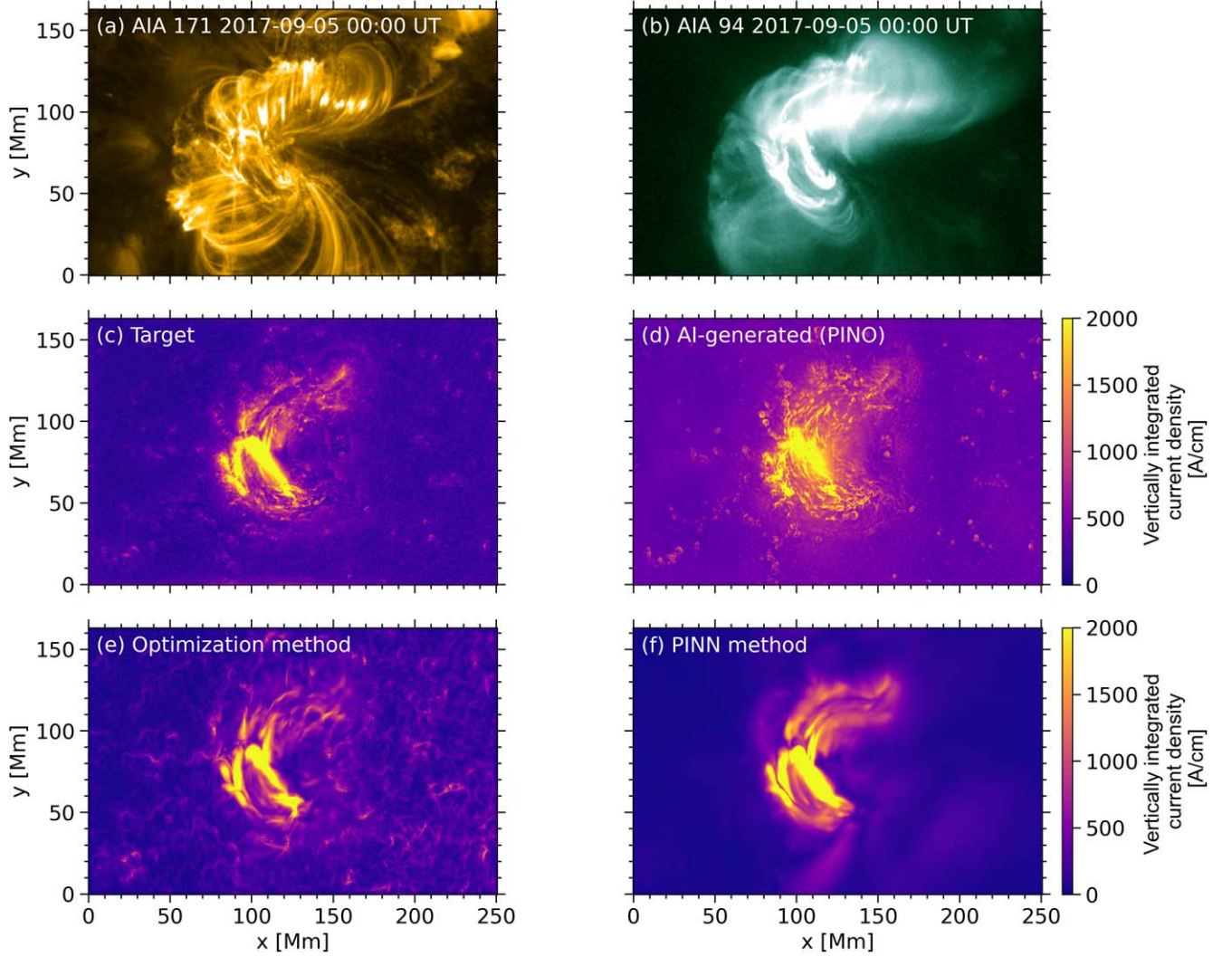


Figure 5. EUV observations of AR 12673 and vertically integrated current density maps from different extrapolation methods. The format of the figure is the same as Figure 2.

inference time due to the need for point sampling in the computational domain. Thus, we use the U-shaped neural operator architecture, which does not rely on coordinate inputs. Improving PINO using coordinate inputs is a future research direction.

As another option to solve the above problem, our PINO model can provide the input for the optimization method (T. Wiegmann et al. 2012). We find that using an AI-generated NLFFF as the initial field for the optimization method leads to a refined AI-generated NLFFF that has better-quality metrics, with minimal changes or slight improvements in the comparison metrics, as shown in Table 1 and indicated by the “AI + optimization method.” For σ_J , the values decrease from 0.24 to 0.08 for the semianalytical field and from 0.85 to 0.47 for the numerical fields. For $\langle |f_i| \rangle$, the values decrease from 5.8×10^{-3} to 2.4×10^{-3} for the semianalytical field and from 5.3×10^{-3} to 1.5×10^{-3} for the numerical fields. Our GPU-accelerated optimization method code converges within 5 minutes for a resolution of $512 \times 256 \times 256$. The total computation time remains below the 12 minutes cadence of SHARP data for observed photospheric vector magnetic fields. However, this is just one approach we have found to address the issue of quality metrics in AI-generated NLFFFs, and further verification of this method is needed.

5. Summary and Conclusion

In this study, we have developed a PINO model for the real-time extrapolation of NLFFFs in the solar corona. After training with physics loss based on force-free and divergence-free conditions and data loss, our PINO model can generate 3D NLFFFs in less than 1 s on a GPU and 2 s on a CPU for new ARs using only observed photospheric vector magnetic fields. The AI-generated NLFFFs by our trained PINO model closely resemble the target NLFFFs both qualitatively and quantitatively, including the Low and Lou (B. C. Low & Y. Q. Lou 1990) semianalytical and ISEE numerical NLFFFs (K. Kusano et al. 2021). The qualitative similarity is illustrated by the magnetic field line structures (Figures 2 and 3) and the spatial distribution of the current density (Figures 4 and 5) for AR 11158 and 12673 in the test set. The quantitative comparison metrics— C_{vec} , C_{CS} , E_n^l , E_m^l , and ϵ —are 1.00, 0.98, 0.91, 0.81, and 1.01 for the semianalytical NLFFF test case, and 0.96, 0.94, 0.70, 0.58, and 0.93 for 30 ARs in the numerical NLFFF test set (Table 1). The trend of 3D magnetic energy from AI-generated NLFFFs (Figure 6) is similar to the target NLFFFs. The AI-generated NLFFFs also exhibit similarities to other NLFFF extrapolation methods

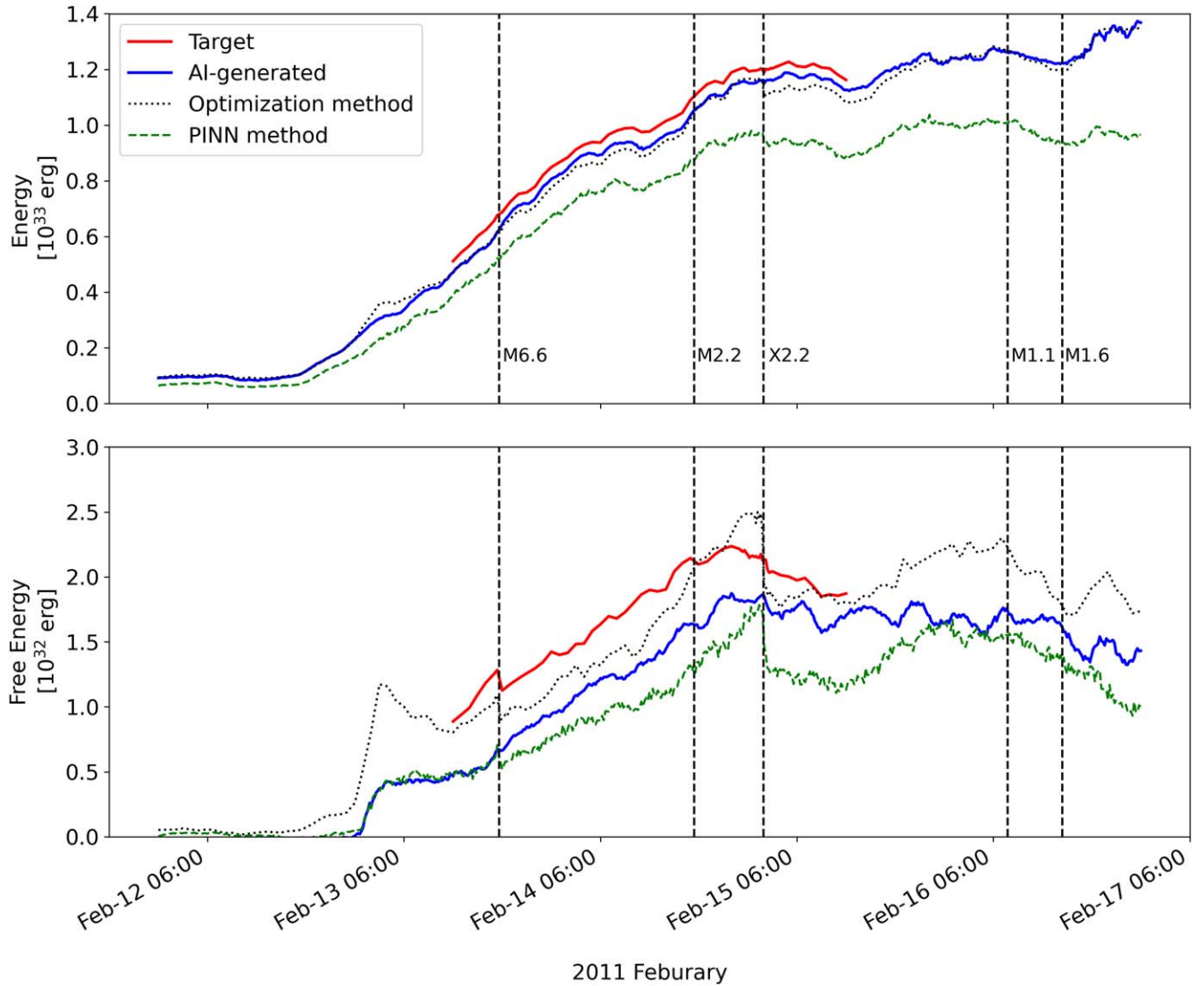


Figure 6. Time series of (top) total magnetic energy and (bottom) free magnetic energy calculated from NLFFFs of AR 11158. The dashed black vertical lines indicate flares $\geq M1$. We use SHARP data as the input data of the PINO model. The solid red line shows magnetic energy from the target fields of the ISEE NLFFF database, which covers the period from 12:00 UTC on 2011 February 13 to 12:00 UTC on 2011 February 15. The solid blue line is for our trained PINO, the dotted black line is for the optimization method (T. Wiegelmann et al. 2012), and the dashed green line is for the PINN method (R. Jarolim et al. 2023). The three lines correspond to the period from 00:00 UTC on 2011 February 12 to 00:00 UTC on 2011 February 17. The data for the black and green lines are obtained from R. Jarolim et al. (2023).

in terms of field line structures (Figures 2 and 3), current density distribution (Figures 4 and 5), and magnetic energy trend (Figure 6).

The ability to generate high-resolution NLFFFs in a fraction of the time required by conventional methods is a major contribution since conventional NLFFF extrapolation techniques often take several hours to compute a single high-resolution NLFFF (at least for the first one). This reduction in computation time by our trained PINO model shows its potential as a method for real-time solar coronal magnetic field modeling. For example, it could be useful for a physics-based flare prediction method based on a series of NLFFFs (K. Kusano et al. 2020). We also propose a solution for unsatisfactory quality metrics of our AI-generated NLFFFs, which is one limitation of our method. When we use the AI-generated field as the initial field for the optimization method, it seems to produce a refined NLFFF with better-quality metrics. The quality metrics for force freeness σ_f and

divergence freeness $\langle |f_i| \rangle$ both decrease for the semianalytical NLFFF test case and for 30 ARs in the numerical NLFFF test set (Table 1). For the semianalytical field, σ_f decreases from 0.24 to 0.08, and $\langle |f_i| \rangle$ decreases from 5.8×10^{-3} to 2.4×10^{-3} . For the numerical fields, σ_f decreases from 0.85 to 0.47, and $\langle |f_i| \rangle$ decreases from 5.3×10^{-3} to 1.5×10^{-3} . However, further verification is needed, as the optimization method usually struggles to converge to new solutions.

We plan to focus on improving AI-generated NLFFFs by using a larger and more diverse data set of NLFFFs from various ARs for training, leading to more accurate real-time extrapolation of NLFFFs. Moreover, integrating traditional PDE solvers into the training process, a concept known as differentiable physics (K. Um et al. 2020), may further enhance the accuracy of the AI-generated NLFFFs. Although this study addresses parametric NLFFF PDEs, we expect that our approach could be extended to full MHD equations for more comprehensive real-time reconstructions of the solar corona.

Acknowledgments

We acknowledge the use of the ISEE database from Nagoya University. We appreciate the referee's constructive comments. We also appreciate the community's efforts in developing the open-source packages used in this work. The SDO data are courtesy of NASA/SDO and the AIA and HMI science teams. We acknowledge the support from JSPS KAKENHI (No. JP21H04492) for maintaining the publicly available ISEE NLFFF database from Nagoya University. This research was supported by the BK21 FOUR program through the National Research Foundation of Korea (NRF) under the Ministry of Education (MoE) (Kyung Hee University, Human Education Team for the Next Generation of Space Exploration), the BK21 FOUR program of Graduate School, Kyung Hee University (No. GS-1-JO-NON-20242364), the Basic Science Research Program through the NRF funded by the MoE (No. RS-2023-00248916), the Korea Astronomy and Space Science Institute under the R&D program (No. 2024-1-850-02) supervised by the Ministry of Science and ICT (MSIT), and the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. RS-2023-00234488, Development of solar synoptic magnetograms using deep learning).

Software: PyTorch (A. Paszke et al. 2019), TorchMetrics (N. S. Dettelsen et al. 2022), Matplotlib (J. D. Hunter 2007), NumPy (C. R. Harris et al. 2020), SciPy (P. Virtanen et al. 2020), Astropy (Astropy Collaboration et al. 2013, 2018, 2022), SunPy (The SunPy Community et al. 2020).

Appendix Numerical Field Data Set Composition

The training set consists of 1795 NLFFFs from 151 ARs: 11078, 11089, 11092, 11093, 11108, 11109, 11117, 11130, 11131, 11133, 11140, 11163, 11164, 11165, 11166, 11169, 11176, 11183, 11190, 11195, 11196, 11199, 11236, 11257, 11260, 11261, 11263, 11271, 11283, 11289, 11301, 11302, 11305, 11312, 11316, 11325, 11327, 11330, 11339, 11354, 11358, 11362, 11363, 11374, 11384, 11386, 11387, 11389, 11390, 11391, 11393, 11402, 11416, 11422, 11428, 11429, 11455, 11459, 11460, 11465, 11471, 11476, 11484, 11486, 11492, 11497, 11512, 11515, 11520, 11543, 11555, 11560, 11562, 11564, 11579, 11582, 11585, 11589, 11591, 11596, 11598, 11613, 11618, 11620, 11635, 11640, 11652, 11654, 11660, 11665, 11682, 11698, 11711, 11718, 11719, 11723, 11726, 11730, 11731, 11736, 11745, 11748, 11755, 11765, 11776, 11777, 11793, 11818, 11827, 11835, 11877, 11884, 11890, 11936, 11944, 11967, 12089, 12109, 12121, 12135, 12144, 12146, 12149, 12152, 12158, 12173, 12175, 12177, 12186, 12192, 12203, 12205, 12209, 12216, 12217, 12219, 12221, 12222, 12232, 12242, 12297, 12371, 12403, 12422, 12489, 12494, 12544, 12738, 12816, 12936, and 12975. The length in the x -direction ranges from a minimum of 78 Mm (AR 11736) to a maximum of 943 Mm (AR 12242), with a mean of 348 Mm, a median of 310 Mm, and a standard deviation of 152 Mm. The length in the y -direction ranges from a minimum of 44 Mm (AR 11736) to a maximum of 413 Mm (AR 12242), with a mean of 167 Mm, a median of 157 Mm, and a standard deviation of 62 Mm.

The validation set consists of 49 NLFFFs from 30 ARs: 11846, 11875, 11899, 11909, 11917, 11946, 11974, 11991, 12002, 12014, 12017, 12026, 12036, 12047, 12080, 12236,

12249, 12253, 12259, 12271, 12277, 12303, 12325, 12360, 12381, 12415, 12420, 12436, 12473, and 12781. The length in the x -direction ranges from a minimum of 234 Mm (AR 12017) to a maximum of 649 Mm (AR 12080), with a mean of 366 Mm, a median of 340 Mm, and a standard deviation of 103 Mm. The length in the y -direction ranges from a minimum of 95 Mm (AR 12271) to a maximum of 368 Mm (AR 12259), with a mean of 182 Mm, a median of 169 Mm, and a standard deviation of 58 Mm.

The test set consists of 483 NLFFFs from 30 ARs: 11158, 11861, 11882, 11908, 11916, 11931, 11968, 11977, 11996, 12011, 12021, 12034, 12042, 12049, 12082, 12241, 12251, 12257, 12268, 12275, 12280, 12305, 12335, 12367, 12396, 12418, 12434, 12443, 12488, and 12673. The length in the x -direction ranges from a minimum of 195 Mm (AR 12275) to a maximum of 535 Mm (AR 12305), with a mean of 313 Mm, a median of 287 Mm, and a standard deviation of 85 Mm. The length in the y -direction ranges from a minimum of 86 Mm (AR 12275) to a maximum of 267 Mm (AR 12082), with a mean of 159 Mm, a median of 155 Mm, and a standard deviation of 48 Mm.

ORCID iDs

Mingyu Jeon  <https://orcid.org/0009-0004-7798-5052>
Hyun-Jin Jeong  <https://orcid.org/0000-0003-4616-947X>
Yong-Jae Moon  <https://orcid.org/0000-0001-6216-6944>
Jihye Kang  <https://orcid.org/0000-0001-6213-4088>
Kanya Kusano  <https://orcid.org/0000-0002-6814-6810>

References

- Asensio Ramos, A., Cheung, M. C. M., Chifu, I., & Gafeira, R. 2023, *LRSP*, **20**, 4
- Astropy Collaboration, Price-Whelan, A. M., Lim, P. L., et al. 2022, *ApJ*, **935**, 167
- Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., et al. 2018, *AJ*, **156**, 123
- Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, *A&A*, **558**, A33
- Bobra, M. G., Sun, X., Hoeksema, J. T., et al. 2014, *SoPh*, **289**, 3549
- Chen, P. F. 2011, *LRSP*, **8**, 1
- Dettelsen, N. S., Borovec, J., Schock, J., et al. 2022, *JOSS*, **7**, 4101
- Faroughi, S. A., Pawar, N. M., Fernandes, C., et al. 2024, *J. Comput. Inform. Sci. Eng.*, **24**, 040802
- Goswami, S., Bora, A., Yu, Y., & Karniadakis, G. E. 2023, *Machine Learning in Modeling and Simulation: Methods and Applications* (Berlin: Springer), 219
- Harris, C. R., Millman, K. J., van der Walt, S. J., et al. 2020, *Natur*, **585**, 357
- Hunter, J. D. 2007, *CSE*, **9**, 90
- Inoue, S., Magara, T., Pandey, V. S., et al. 2013, *ApJ*, **780**, 101
- Jarolim, R., Thalmann, J. K., Veronig, A. M., & Podladchikova, T. 2023, *NatAs*, **7**, 1171
- Jarolim, R., Tremblay, B., Rempel, M., et al. 2024, *ApJL*, **963**, L21
- Jeong, H.-J., Moon, Y.-J., Park, E., Lee, H., & Baek, J.-H. 2022, *ApJS*, **262**, 50
- Karniadakis, G. E., Kevrekidis, I. G., Lu, L., et al. 2021, *NatRP*, **3**, 422
- Kingma, D. P., & Ba, J. 2014, arXiv:1412.6980
- Kovachki, N. B., Li, Z., Liu, B., et al. 2021, arXiv:2108.08481
- Kusano, K., Iijima, H., Kaneko, T., et al. 2021, ISEE Database for Nonlinear Force-Free Field of Solar Active Regions, v1.0, Nagoya University, doi:10.34515/DATA.HSC-00000
- Kusano, K., Iju, T., Bamba, Y., & Inoue, S. 2020, *Sci*, **369**, 587
- Lemen, J. R., Title, A. M., Akin, D. J., et al. 2012, *SoPh*, **275**, 17
- Li, Z., Kovachki, N., Azizzadenesheli, K., et al. 2020, arXiv:2010.08895
- Li, Z., Zheng, H., Kovachki, N., et al. 2024, *ACM/IMS J. Data Sci.*, **1**, 1
- Linker, J. A., Mikić, Z., Biesecker, D. A., et al. 1999, *JGR*, **104**, 9809
- Low, B. C., & Lou, Y. Q. 1990, *ApJ*, **352**, 343
- Lu, L., Jin, P., Pang, G., Zhang, Z., & Karniadakis, G. E. 2021, *NatMI*, **3**, 218
- Lu, L., Meng, X., Cai, S., et al. 2022, *CMAME*, **393**, 114778

- Mikić, Z., Linker, J. A., Schnack, D. D., Lionello, R., & Tarditi, A. 1999, [PhPI](#), **6**, 2217
- Mikić, Z., & McClymont, A. N. 1994, in ASP Conf Ser. 68, Solar Active Region Evolution: Comparing Models with Observations, ed. K. S. Balasubramaniam & G. W. Simon (San Francisco, CA: ASP), 225
- Paszke, A., Gross, S., Massa, F., et al. 2019, arXiv:1912.01703
- Pesnell, W. D., Thompson, B. J., & Chamberlin, P. C. 2012, [SoPh](#), **275**, 3
- Rahman, M. A., Ross, Z. E., & Azizzadenesheli, K. 2022, arXiv:2204.11127
- Rahman, S., Jeong, H.-J., Siddique, A., Moon, Y.-J., & Lawrance, B. 2024, [ApJS](#), **271**, 14
- Raissi, M., Perdikaris, P., & Karniadakis, G. E. 2019, [JCoPh](#), **378**, 686
- Scherrer, P. H., Schou, J., Bush, R. I., et al. 2012, [SoPh](#), **275**, 207
- Schrijver, C. J., De Rosa, M. L., Metcalf, T. R., et al. 2006, [SoPh](#), **235**, 161
- Shibata, K., & Magara, T. 2011, [LRSP](#), **8**, 6
- Temmer, M. 2021, [LRSP](#), **18**, 4
- The SunPy Community, Barnes, W. T., Bobra, M. G., et al. 2020, [ApJ](#), **890**, 68
- Um, K., Brand, R., Fei, Y., Holl, P., & Thürey, N. 2020, arXiv:2007.00016
- Valori, G., Démoulin, P., Pariat, E., & Masson, S. 2013, [A&A](#), **553**, A38
- Virtanen, P., Gommers, R., Oliphant, T. E., et al. 2020, [NatMe](#), **17**, 261
- Wang, S., Wang, H., & Perdikaris, P. 2021, [SciA](#), **7**, eabi8605
- Wheatland, M. S., Sturrock, P. A., & Roumeliotis, G. 2000, [ApJ](#), **540**, 1150
- Wiegmann, T., Inhester, B., & Sakurai, T. 2006, [SoPh](#), **233**, 215
- Wiegmann, T., Petrie, G. J. D., & Riley, P. 2017, [SSRv](#), **210**, 249
- Wiegmann, T., & Sakurai, T. 2021, [LRSP](#), **18**, 1
- Wiegmann, T., Thalmann, J. K., Inhester, B., et al. 2012, [SoPh](#), **281**, 37
- Zhang, J., Temmer, M., Gopalswamy, N., et al. 2021, [PEPS](#), **8**, 56