Dynamical System Analysis & Forecasting of the Geodynamo

Luke Fairbanks & Ashley Thorshov

Table of Contents

Dynamical Systems: General

- differential equation sets/iterated maps
- (often) conservation law w/ set of flux/forcing terms eg. momentum, charge, continuity, induction
- relation equations or constraints eg. equation of state, symmetries
- approximations or otherwise misaligned
- even deterministic systems chaotic
- low-dimensional data eg. fluids, plasma, biochemistry, electrical systems, networks, etc..
- few analytic solutions, numerical approximation next best

Dynamical System Definition

-Assume generalized dynamical system of the form

$$\frac{dx_i(t)}{dt} = f_i(\mathbf{x}(t), p)$$

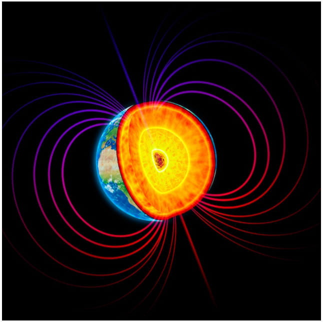
$$i = 1, ..., D$$

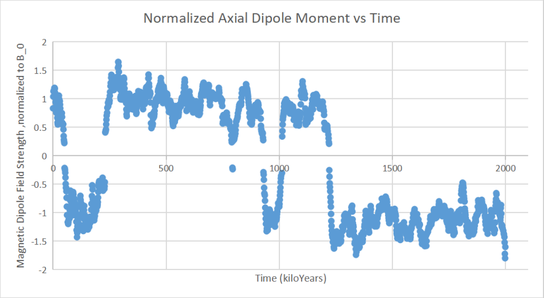
D is the dimension, p is the parameters

-Constructed from first principles or inferred from data

Geodynamo Problem Setup

In coordination with SIO & other groups*





Differential System of Geodynamo

'Where are the physics of this system?'*

Momentum Conservation (dimensionless)

$$\frac{E}{P_m}\frac{D\mathbf{u}}{Dt} + 2\hat{\mathbf{z}} \times \mathbf{u} = -\nabla p + (\nabla \times \mathbf{B}) \times \mathbf{B} + E\nabla^2\mathbf{u} + \frac{R_a E P_m}{P_r}(T + \xi)\frac{\mathbf{r}}{r_{cmb}}$$

Magnetic Induction (dimensionless)

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla^2\mathbf{B} + \nabla \times (\mathbf{u} \times \mathbf{B})$$

Heat Evolution

$$\frac{\partial T}{\partial t} = \frac{P_m}{P_r}\nabla^2 T - \mathbf{u} \cdot \nabla T + Q$$

(Dynamics) Parameters and Symbols Reference

- Rayleigh number: $R_a = \frac{g_0 \alpha \Delta T d^3}{\nu \kappa}$
- (viscous/fluid) Prandtl number: $P_r = \frac{\nu}{\kappa}$
- Magnetic Prandtl Number: $P_m = \frac{\nu}{\eta}$
- Ekman number: $E = \frac{\nu}{\Omega d^2}$
- kinematic viscosity: ν
- Thermal diffusivity: κ
- axial rotation frequency of Earth: Ω
- magnetic diffusivity: η
- thermal expansion coefficient: α
- gravity at core-mantle boundary(CMB): g_0
- length gap between inner-core boundary(ICB) and CMB: d
- Material derivative: $\frac{D}{Dt} = \frac{\partial}{\partial t} + (\mathbf{u} \cdot \nabla)$

Research Group Method Comparison

Data Assimilation

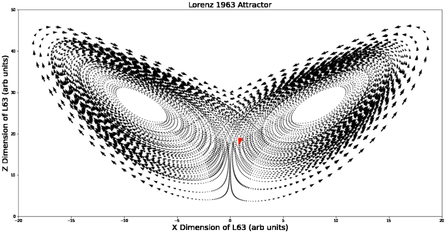
- aligns parameters of defined model approximating dynamics
- robust in ideal conditions
- difficult to train & expensive
- eg. PAHMC, MinAOne

Data-Driven Approach

- fits ML architectural model
- may include physics informed structures
- fast & easy training
- flexible, robust, generalizable
- eg. Reservoir Computing, Data-Driven Forecasting

Dynamical Flow Map

(Graph derived from ODEINT) 'flow map' from strange attractor (Lorenz)



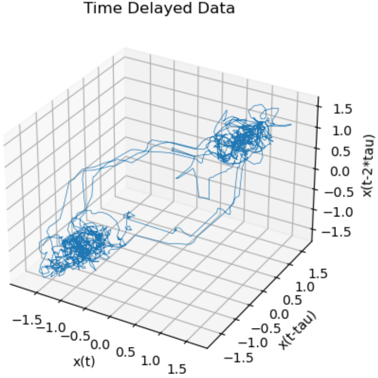
However; LowD data from highD systems

eg. magnetic dipole moment of earth rather than states of liquid metal in outer core region

Initial Findings: Time Delay Embedding Attractor Reconstruction(only 3D)

noting double-well => consistent w/ 'flipping dipole'

$\tau = 8\Delta_t = 8000 year$
$D_{E,3DPlot} = 3 < D_E \approx 5$



Takken's Embedding Theorem for State Space Reconstruction

Theorem: can reconstruct 'proxy' space equivalent (diffeomorphism) to original dynamical space via functional transformations of observed dimensions

generalized state reconstruction:

$$\mathbf{s}(n) = [h(\mathbf{x}(n)), h(\mathbf{g}^{T_1}(\mathbf{x}(n))), ..., h(\mathbf{g}^{T_{D_e-1}}(\mathbf{x}(n)))]$$

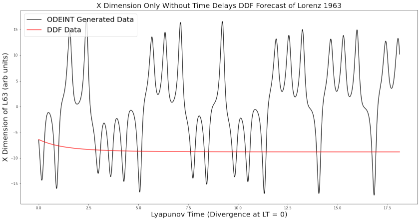
time delay embedding form:

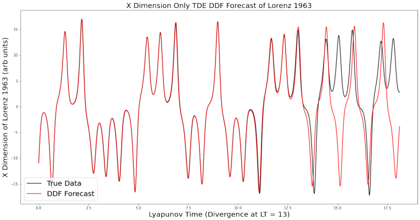
$$\mathbf{s}(n) = [\mathbf{y}(n), \mathbf{y}(n-\tau), ..., \mathbf{y}(n-(D_E-1)\tau)]$$

τ time delay, D_E embedding dimension

Why TDE?

otherwise DDF may fall short; unaware of higher dimensionality & geometry of the attractor





Embedding Dimension D_E

Want 'false nearest neighbors' percentage near zero

- data 'projected' from the highD system, many points 'near' each other
- 'unfolding' attractor

$$R_{D_E}(n)^2 = \sum_{d}^{D_E * D_0} [s_d(n) - s_d^{NN}(n)]^2$$

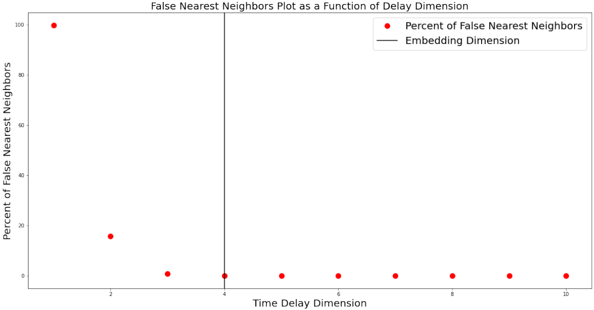
$$R_{D_E+1}(n)^2 = \sum_{d}^{(D_E+1)*D_0} [s_d(n) - s_d^{NN}(n)]^2$$

$$R_{nD_E} = \frac{R_{D_E+1}(n)}{R_{D_E}(n)}$$

D_0 dimensionality of observation, s_d state components, s_d^{NN} nearest neighbor state components

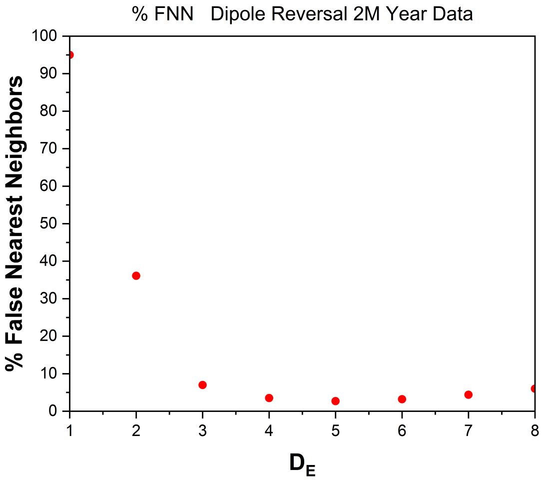
Embedding Dimension D_E Graph

FNN analysis (Lorenz 63' X)



note computational efficiency considerations, noise

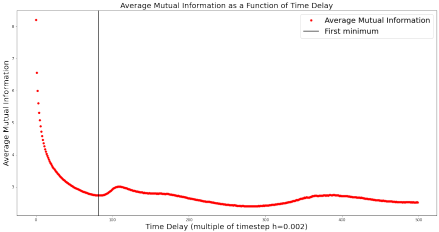
Initial Findings: False Nearest Neighbors



Time Delay τ

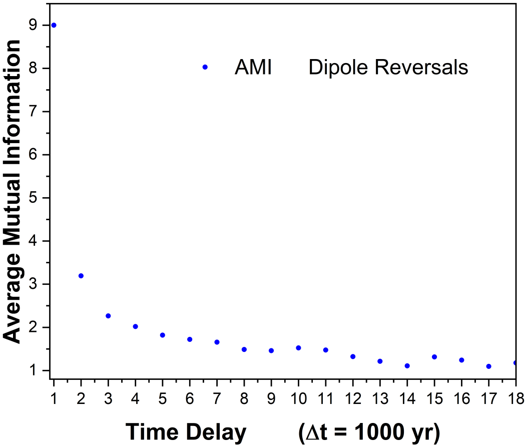
τ typically determined by average mutual information

- $I_{AB} = \sum_{\mathbf{y}(n),\mathbf{y}(n-\tau)} P(\mathbf{y}(n), \mathbf{y}(n-\tau)) log_2 \frac{P(\mathbf{y}(n),\mathbf{y}(n-\tau))}{P(\mathbf{y}(n))P(\mathbf{y}(n-\tau))}$
- want low AMI so relatively independent without total decorrelation
- scalar multiple of observation timing , ΔT, $\tau = \zeta \Delta T$, $\zeta \in \mathbb{Z}$



AMI vs τ for L63' x data

Initial Findings: Average Mutual Information



Lyapunov exponents

Eigenvalues of Jacobian of dynamical system

$$\mathbf{T}_{ab}(x) = \frac{\partial F_a(\mathbf{X})}{\partial x_b}$$

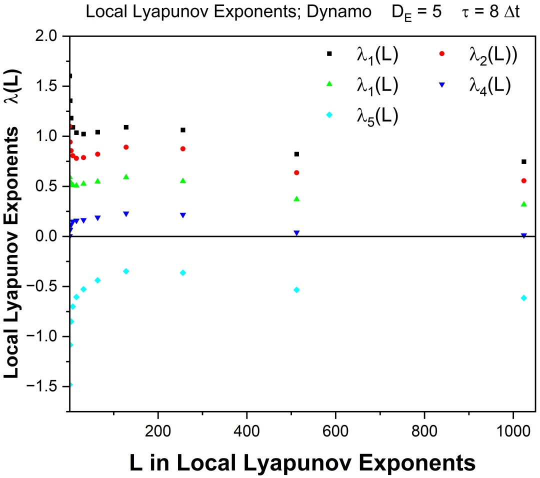
F here being update rule/map

Measure of chaotic nature of system $e^{\lambda t}$, predictability

lyapunov timescale $t_\lambda = \frac{1}{\lambda}$

Initial Findings: Lyapunov Spectrum

noting net negativity, one of the set near zero, some positives



Initial Data-Driven Forecasting Results

ran the PADM2M data through DDF

high frequency noise

Proposed solutions

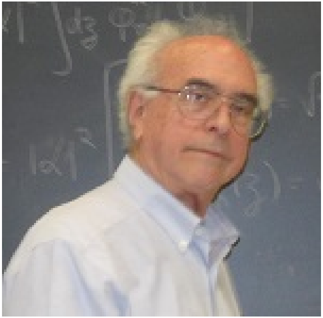
- 'Averaging out' noise in the data
- hyper-parameters; R, β
- Interpolator
- Physical model matching

Momentum of Project, prior to passing of Henry Abarbanel

- analyses of geodynamo data, PADM2M & simulated, in coordination with SIO group; Matthias Morzfeld, Catherine Constable, & William Davis
- DDF alignment to PADM2M & MHD simulated data
- Galerkin approximation of geodynamo outer core region
- then synchronize/align systems
- finally analyze emergent results

Acknowledgements

- Henry Abarbanel
- other H group members: Randall Clark, Lawson Fuller, Daniel Primosche, etc....
- collaborators eg. SIO researchers, groups behind Geodynamo modeled data & simulation data
- UCSD Physics Department



References

- Shallow Water Equation paper: https://doi.org/10.48550/arXiv.2303.16363
- Randall's thesis paper, will put link to escholarship where one can find UCSD phd theses since he just submitted, been working off of an earlier draft
https://escholarship.org/search?q=randall%20clark&searchType=eScholarship&searchUnitType=series
- Jones Geodynamo: Planetary Magnetic Fields and Fluid Dynamos;Chris A. Jones;Annual Review of Fluid Mechanics 2011 43:1, 583-614
- dropbox to other geodynamo project work:
https://www.dropbox.com/scl/fo/2gubylryjehdj5ei1rsz2/h?dl=0&rlkey=rbc4cgsw0n1n8041z9jknqadf
- dropbox to 'shared documents(Randall's thesis draft, Henry's Geodyanmo draft, my draft of paper utilizing sympy to reflect and extend Henry's geodynamo theory for use in Galerkin approximation, documents from a class project under Henry where we analyzed the geodynamo data):
https://www.dropbox.com/scl/fo/wg9pi6jt5zqpd9imsrcl4/h?dl=0&rlkey=71sl7mhsenrls40mhxmgex82y
- github where some of our Data driven forecasting methods may be referenced:
https://github.com/RandarserousRex/Thesis-DDF-Code

image acknowledgement

- magnetic field illustration $https://www.google.com/url?sa = i\&url = https\%3A\%2F\%2Fcosmosmagazine.com\%2Fearth\%2Fearth - sciences\%2Fcore - conundrum - how - old - is - earths - magnetic - field\%2F\&psig = AOvVaw3QJrA1RDPo - yoKv1BSM4wx\&ust = 1686355651445000\&source = images\&cd = vfe\&ved = 0CBAQjRxqFwoTCIDF2L7ytP8CFQAAAAAdAAAAABAE$

Vorticity

Curl of momentum w/ vorticity $\boldsymbol{\omega} = \nabla \times \mathbf{u}$

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + \mathbf{u} \cdot \nabla \boldsymbol{\omega} = (\boldsymbol{\omega} \cdot \nabla)\mathbf{u} + \nabla \times \boldsymbol{F}$$

F encapsulates

$$\boldsymbol{F} = -2\hat{\mathbf{z}} \times \mathbf{u} + (\nabla \times \mathbf{B}) \times \mathbf{B} + E\nabla^2\mathbf{u} + \frac{R_a E P_m}{P_r}(T + \xi)\frac{\mathbf{r}}{r_{cmb}}$$

waiving equation of state

Solenoidal Field Representation

Soleoidal nature of the fields

$$\nabla \cdot \mathbf{u} = \nabla \cdot \mathbf{B} = 0$$

angular momentum operator $\mathbf{\Lambda} = \mathbf{r} \times \nabla$

$\mathbf{r}$ position

forming fields, $\mathbf{u}(r, \theta, \phi, t)$ or $\mathbf{B}(r, \theta, \phi, t)$

$$\mathbf{Field}(r, \theta, \phi) = \nabla \times (\mathbf{\Lambda} a((r, \theta, \phi))) + \mathbf{\Lambda} b(r, \theta, \phi)$$

Then $\mathbf{r} \cdot \mathbf{Field} = \mathbf{\Lambda}^2 a$ and $\mathbf{\Lambda} \cdot \mathbf{Field} = \mathbf{\Lambda}^2 b$

Scalar Field Special Function Expansion

Expansion of special functions st scalar field functions a,b,d,g, and T may be represented (general scalar field symbol s)

$$s(r,\theta,\phi,t) = \sum_{n=1}^{\infty}\sum_{l=0}^{\infty}\sum_{m=-l}^{l} S_{lmn}(t)\chi_n(r)Y_{lm}(\theta,\phi)$$

with spherical harmonics $Y_{lm}(\theta,\phi)$ and spherical bessel functions $\chi_n(z) = Aj_n(z) + By_n(z)$

Sympy Computational Extension of Theory

Reduce highD dynamics of core to DE's $O(D_E)$
sympy to systematically match Henry's theory, extend

- handles low-level math
- iterable prototyping process
- coupling w/ established systems
- potential for emergent meta-analyses

Geodynamo Problem Setup

We wish to tune a DDF model to alignment with both this data & other comparable data sets, but at the same time explore the possibilities of a 'Galerkin approximation' akin to what Lorenz did with the atmosphere in '63

ie. reducing a high dimensional fluid/plasma dynamical system to a set of simplified differential equations

-Note existence of other geodynamo proxy models eg. double potential well, brownian motion/SDE, low dimensional DE's, etc... Then to conduct

analyses on that system in a feedback loop between data, simulation, and prediction(DDF) systems

TISEAN analysis & Notes before Theory Deep Dive

Also noting we took some time to double check via fresh python algorithms to check these numbers as well as a short stint learning TISEAN(Nonlinear Time Series Analysis codebase), & python 'Nolds' package which serves a similar purpose
-ie. more sanity checking

concurrent results more or less, but more importantly initial steps of implementing a 'generalized dynamical system analysis algorithm' (tbd)

Rather than exhaustively show results, we'll include the papers & codes in the shared documents