

# NUMERICAL EFFICACY STUDY OF DATA ASSIMILATION FOR THE 2D MAGNETOHYDRODYNAMIC EQUATIONS

JOSHUA HUDSON

University of Maryland, Baltimore County  
 1000 Hilltop Circle  
 Baltimore, MD 21055, USA

now at

Johns Hopkins University Applied Physics Laboratory  
 11100 Johns Hopkins Road  
 Laurel, MD 20723-6099, USA

MICHAEL JOLLY\*

Department of Mathematics  
 Indiana University  
 Bloomington, IN 47405, USA

(Communicated by Robert McLachlan)

**ABSTRACT.** We study the computational efficiency of several nudging data assimilation algorithms for the 2D magnetohydrodynamic equations, using varying amounts and types of data. We find that the algorithms work with much less resolution in the data than required by the rigorous estimates in [7]. We also test other abridged nudging algorithms to which the analytic techniques in [7] do not seem to apply. These latter tests indicate, in particular, that velocity data alone is sufficient for synchronization with a chaotic reference solution, while magnetic data alone is not. We demonstrate that a new nonlinear nudging algorithm, which is adaptive in both time and space, synchronizes at a super exponential rate.

**1. Introduction.** Various approaches to data assimilation, whereby observations are incorporated into mathematical models, date back at least fifty years. Satellite-borne weather observations became available in the 1960s, inspiring Charney, Halem and Jastrow [10] to inject data directly into the nonlinear term of a model for the atmosphere (see also [21, 30]). A remarkable treasure of geomagnetic data from over four centuries of ships' logs has been gathered [26], which prompts the question of how such information could be incorporated into models of the magnetic field at the core-mantle boundary [23]. We present in this paper numerical test results on data assimilation by nudging applied to the magnetohydrodynamic (MHD) equations.

The nudging approach to data assimilation dates back to the 1970s [28, 32]. It is easily described in terms of an evolution equation

$$\frac{d}{dt}y = F(y) \tag{1}$$

---

2010 *Mathematics Subject Classification.* Primary: 34D06, 76W05.

*Key words and phrases.* Magnetohydrodynamic equations, data assimilation, synchronization.

\* Corresponding author: Michael Jolly.

for which initial data is unknown. Measurements are collected at sparse locations in space at all times  $t \in [0, t_0]$ , represented by  $I_h(y(t))$ , where  $I_h$  is some interpolating linear operator approximating the identity with resolution  $h$ . This data is then fed into the initial value problem

$$\frac{d}{dt}Y = F(Y) - \mu I_h(Y - y), \quad (2a)$$

$$Y(0) = 0. \quad (2b)$$

It was shown for a reaction diffusion equation [4] and the 2D Navier–Stokes equations (NSE) [3] that if  $\mu$  is sufficiently large, and  $h$  sufficiently small, then  $Y(t)$  converges to the reference solution  $y(t)$  at an exponential rate as  $t$  increases. Thus, after only a small amount of time  $t_0$ , one can effectively use  $Y(t_0)$  as initial data in (1), and solve for  $t > t_0$ . The initial data in (2) can be arbitrarily chosen; it need not be zero. Convergence of this nudging type algorithm has also been established for the 3D Brinkman–Forchheimer Extended Darcy model [29], the 3D Navier–Stokes– $\alpha$  model [1], the subcritical surface quasi-geostrophic (SQG) equation [24] and the damped driven Korteweg–de Vries equation [25].

Unlike probabilistic approaches to data assimilation such as extended Kalman filters, and 3D/4D Var (see the books [13, 31]), nudging does not include uncertainty quantification. The analysis in [3] has, however, been extended in [5] to the case where the observational data is corrupted by stochastic noise. More recently, in [6], a downscaling nudging scheme has been developed in which coarse mesh statistics are obtained from discrete spatial measurements.

Abridged nudging algorithms in which data are collected on only a subset of the variables have been analyzed for several systems. It was shown in [16] that data from just one component of velocity is sufficient to recover a reference solution of the 2D NSE. Similarly, for 2D Rayleigh–Bénard (RB) convection with no-slip boundary conditions, it was shown in [15] that velocity data alone suffices; temperature observations are not needed. Then for the RB problem with stress-free boundary conditions it was proved in [19] that one can synchronize using data only in the horizontal component of velocity. Charney’s conjecture, that temperature data alone would suffice for RB convection, remains unproved, though it is shown to be so for both a 3D porous media convection model [17], and 3D planetary geostrophic model [18].

All of the above works on abridged algorithms provide rigorous estimates for lower bounds on  $\mu$  and upper bounds on the spatial resolution  $h$ . It is likely that the algorithms perform better, i.e., convergence is achieved with data that is much more coarse than the estimates require. This is in fact demonstrated in numerical tests carried out for the 2D NSE [20, 22], and 2D RB convection model [2, 14].

The numerical results for the MHD equations presented in this paper are compared with rigorous estimates in [7]. Since there are two fields, there are several natural abridged algorithms to consider: velocity data alone, magnetic field data alone, and one component of both. In addition, we test an algorithm that uses just one of the two Elsässer variables, which are the sum and difference of the two fields. As in numerical tests on other systems, we generally demonstrate that the algorithms work much better than the rigorous estimate guarantee. Considering the vast historical geomagnetic data, however, it should be noted that the abridged algorithm using only magnetic field data does not appear to converge.

In Section 2 we present three nudging algorithms and state the rigorous estimates for their convergence proved in [7]. Then in Section 3 we determine a near optimal value of  $\mu$  through numerical experiments over a short time interval. That is followed by longer time simulations which demonstrate convergence. Other algorithms which have not (yet) been analyzed are tested numerically in Section 4. Some are abridged algorithms in which data is collected in a certain subset of the state variables. Others are inspired by [27] in which the nudging term is amplified when small. Finally in Section 5 we demonstrate the robustness of the main algorithm to noise in the observed data.

**2. Rigorous results.** The MHD equations are a set of equations that model fluid dynamics when the fluid is conductive and coupled with an evolving magnetic field. This situation arises when an electrically conductive fluid is in the presence of a magnetic field; as the fluid moves, the magnetic field responds, and in turn induces motion in the fluid. These nonlinear interactions are quadratic, and couple the evolution equations of the velocity and magnetic fields.

Let  $u$  and  $b$  represent the velocity and magnetic fields respectively, and  $p$  the fluid pressure. We consider the MHD system as follows:

**System 2.1** (MHD).

$$\partial_t u - \nu \Delta u + (u \cdot \nabla) u + \frac{1}{\rho} \nabla \left( p + \frac{1}{2\mu_0} |b|^2 \right) = \frac{1}{\rho\mu_0} (b \cdot \nabla) b + f, \quad (3a)$$

$$\partial_t b - \lambda \Delta b + (u \cdot \nabla) b + \nabla q = (b \cdot \nabla) u + g, \quad (3b)$$

$$\nabla \cdot u = \nabla \cdot b = 0, \quad (3c)$$

on the domain  $\Omega = [0, 2\pi]^2$ , equipped with periodic boundary conditions. We denote the fluid viscosity by  $\nu$  and the fluid density by  $\rho$  (which we assume to be constant and homogeneous). The role of viscosity for the magnetic field is played by  $\lambda := (\mu_0 \sigma)^{-1}$ , where  $\mu_0$  is the permeability of free space and  $\sigma$  the electrical conductivity of the fluid. The extra pressure term  $q$  is included for computational purposes; the divergence free condition on  $b$  will force  $q$  to be exactly 0, but we choose to include  $q$  so that the divergence free condition can be explicitly enforced in our simulations. We assume for simplicity that the forcing functions,  $f$  and  $g$ , are time independent.

In this context, we define the nondimensional Grashof number to be:

$$G := \max \left\{ \frac{1}{\nu^2}, \frac{1}{\lambda^2} \right\} \max \left\{ \|f\|_{L^2(\Omega)}, \frac{1}{\sqrt{\rho\mu_0}} \|g\|_{L^2(\Omega)} \right\}. \quad (4)$$

The Grashof number is a comparison between the amount of energy being put into the system and the amount of energy being dissipated, and serves as a complexity parameter; at higher values for  $G$  we expect more information is needed to capture the dynamics.

We apply the general idea of (2) to the MHD equations. We assume that for a given reference solution,  $(u, b)$ , of (2.1), we have data being collected on some subset of the fields  $\{u_1, u_2, b_1, b_2\}$ . A nudging term could be introduced into the evolution equation for any variable on which we are collecting data, and so we can consider a different algorithm for each combination of the variables we assume to be measuring.

**Remark 1.** As the pressure field,  $p$ , does not have an evolution equation, we cannot directly make use of any data collected on  $p$  with an equation like (2).

In this paper, we consider  $I_h$  to be the projection onto the low modes,  $P_N$ , defined as follows: if  $u$  has the Fourier expansion  $u(t, x) = \sum_{k \in \mathbb{Z}^2} \hat{u}(t, k) e^{ik \cdot x}$ , then

$$P_N u(t, x) = \sum_{k_1, k_2 = -N}^N \hat{u}(t, k) e^{ik \cdot x}.$$

So in this case,  $h = \frac{1}{N}$ , and it is easy to show that for  $u \in H^1$ ,

$$\|P_N u - u\|_{L^2} \lesssim h \|\nabla u\|_{L^2}. \quad (5)$$

Although we only consider the case  $I_h = P_N$ , the theoretical results are the same for any  $I_h$  satisfying (5), such as volume interpolants. Rigorous results have also been obtained for nodal interpolants, which satisfy a less restrictive version of (5), at the cost of requiring higher resolution in the measurement data [7] (see [16] for examples of different interpolants satisfying (5), as well as interpolants which satisfy other regularity conditions). Indeed, an advantage of nudging is the flexibility of the interpolant. Practically speaking, Fourier modes are not expected to be directly measured in observed data, but could be obtained via a fast Fourier transform from data at nodes on a grid, albeit with an aliasing error (see [9] for more on spectral filtering of observables).

The first algorithm we consider utilizes data collected on all the variables (except the pressure). As it uses more information about the reference solution than the other algorithms we consider, we expect it to have the best performance.

**Algorithm 2.2.** *Solve*

$$\begin{aligned} \partial_t \mathbf{U} - \nu \Delta \mathbf{U} + (\mathbf{U} \cdot \nabla) \mathbf{U} - \frac{1}{\rho \mu_0} (\mathbf{B} \cdot \nabla) \mathbf{B} &= -\frac{1}{\rho} \nabla \tilde{p} + f + \mu P_N(u - \mathbf{U}), \\ \partial_t \mathbf{B} - \lambda \Delta \mathbf{B} + (\mathbf{U} \cdot \nabla) \mathbf{B} - (\mathbf{B} \cdot \nabla) \mathbf{U} &= -\nabla q + g + \mu P_N(b - \mathbf{B}), \\ \nabla \cdot \mathbf{U} &= 0, \nabla \cdot \mathbf{B} = 0, \end{aligned}$$

for  $(\mathbf{U}, \mathbf{B})$  on  $[t_0, t_0 + T]$ , with the initial condition  $\mathbf{U}(t_0), \mathbf{B}(t_0) \equiv 0$ .

Note that we fold the affect of  $|B|$  into the definition of  $\tilde{p}$ , for notational simplicity.

In the next algorithm, we require data to be collected on only the horizontal components of each field.

**Algorithm 2.3.** *Solve*

$$\begin{aligned} \partial_t \mathbf{U}_1 - \nu \Delta \mathbf{U}_1 + (\mathbf{U} \cdot \nabla) \mathbf{U}_1 - \frac{1}{\rho \mu_0} (\mathbf{B} \cdot \nabla) \mathbf{B}_1 &= -\frac{1}{\rho} \partial_x \tilde{p} + f_1 + \mu P_N(u_1 - \mathbf{U}_1), \\ \partial_t \mathbf{U}_2 - \nu \Delta \mathbf{U}_2 + (\mathbf{U} \cdot \nabla) \mathbf{U}_2 - \frac{1}{\rho \mu_0} (\mathbf{B} \cdot \nabla) \mathbf{B}_2 &= -\frac{1}{\rho} \partial_y \tilde{p} + f_2, \\ \partial_t \mathbf{B}_1 - \lambda \Delta \mathbf{B}_1 + (\mathbf{U} \cdot \nabla) \mathbf{B}_1 - (\mathbf{B} \cdot \nabla) \mathbf{U}_1 &= -\partial_x q + g_1 + \mu P_N(b_1 - \mathbf{B}_1), \\ \partial_t \mathbf{B}_2 - \lambda \Delta \mathbf{B}_2 + (\mathbf{U} \cdot \nabla) \mathbf{B}_2 - (\mathbf{B} \cdot \nabla) \mathbf{U}_2 &= -\partial_y q + g_2, \\ \nabla \cdot \mathbf{U} &= 0, \nabla \cdot \mathbf{B} = 0, \end{aligned}$$

for  $(\mathbf{U}, \mathbf{B})$  on  $[t_0, t_0 + T]$ , with the initial condition  $\mathbf{U}(t_0), \mathbf{B}(t_0) \equiv 0$ .

In addition to measuring the components of  $u$  and  $b$  directly, it is possible to directly measure one of the Elsässer variables  $\frac{1}{2}(u \pm \frac{1}{\rho \mu_0} b)$  [12]. With this in mind, we consider the following algorithm which requires collecting data on only one Elsässer variable, and hence, as with Algorithm 2.3, only half of the observables.

**Algorithm 2.4.** *Solve*

$$\begin{aligned} \partial_t \mathbf{U} - \nu \Delta \mathbf{U} + (\mathbf{U} \cdot \nabla) \mathbf{U} - \frac{1}{\rho \mu_0} (\mathbf{B} \cdot \nabla) \mathbf{B} &= -\frac{1}{\rho} \nabla \tilde{p} + f \\ &\quad + \frac{1}{2} \mu P_N \left( u + \frac{1}{\sqrt{\rho \mu_0}} b - \mathbf{U} - \frac{1}{\sqrt{\rho \mu_0}} \mathbf{B} \right), \\ \partial_t \mathbf{B} - \lambda \Delta \mathbf{B} + (\mathbf{U} \cdot \nabla) \mathbf{B} - (\mathbf{B} \cdot \nabla) \mathbf{U} &= -\nabla q + g \\ &\quad + \frac{1}{2} \mu P_N \left( u + \frac{1}{\sqrt{\rho \mu_0}} b - \mathbf{U} - \frac{1}{\sqrt{\rho \mu_0}} \mathbf{B} \right), \\ \nabla \cdot \mathbf{U} &= 0, \nabla \cdot \mathbf{B} = 0, \end{aligned}$$

for  $(\mathbf{U}, \mathbf{B})$  on  $[t_0, t_0 + T]$ , with the initial condition  $\mathbf{U}(t_0), \mathbf{B}(t_0) \equiv 0$ .

Sufficient conditions for the above algorithms to produce solutions which converge in  $L_2$  (and  $H^1$ ) to  $(u, b)$  were obtained in [7], which we summarize in the following theorem:

**Theorem 2.5.** *Let  $(u, b)$  be a strong solution of (2.1) with initial condition  $(u_0, b_0)$  on the global attractor. For each of the above algorithms, there exists constants  $c, C < \infty$  (which depend on  $\nu, \lambda$  and  $G$ ) such that if  $\mu > c$  and*

$$N \gtrsim \sqrt{\frac{\mu}{\min\{\nu, \lambda\}}}, \quad (6)$$

then the system of equations defined in the algorithm will have a unique strong solution,  $(\mathbf{U}, \mathbf{B})$ , on  $[t_0, \infty)$ , which converges to the reference solution  $(u, b)$  in  $L_2$ .

Specifically, for the case of Algorithm 2.2,

$$c = \mathcal{O}(G^2), \quad (7)$$

and

$$\|u(t) - \mathbf{U}(t)\|_{L^2}^2 + \|b(t) - \mathbf{B}(t)\|_{L^2}^2 \leq e^{-(\mu-c)t} (\|u_0\|_{L^2}^2 + \|b_0\|_{L^2}^2) C_{\nu, \lambda, G}. \quad (8)$$

For Algorithm 2.3 and Algorithm 2.4,

$$c = \mathcal{O}(G^6), \quad (9)$$

and

$$\|u(t) - \mathbf{U}(t)\|_{L^2}^2 + \|b(t) - \mathbf{B}(t)\|_{L^2}^2 \leq e^{-\min\{\nu, \lambda, \mu-c\}t} (\|u_0\|_{L^2}^2 + \|b_0\|_{L^2}^2) C_{\nu, \lambda, G}. \quad (10)$$

We now test these algorithms to see how well they perform in practice. The rigorous results above ensure that Algorithms 2.2 – 2.4 will work provided that the appropriate conditions on  $\mu$  and  $h$  are satisfied, and that the data and solutions to the systems in the algorithms are exact. Those estimates, if sharp, would require a prohibitive amount of data:  $N \sim G$  ((6) and (7)) or worse  $N \sim G^3$  ((6) and (9)). We want to demonstrate that the algorithms are effective using data that is much more coarse than the estimates require.

**3. Computational results.** All of our computations were performed on the supercomputers at Indiana University, using dedalus [8], an open source pseudo-spectral package (see <http://dedalus-project.org/>). An implicit/explicit Runge Kutta 222 time stepping scheme was used (except in Section 5); the linear terms were solved implicitly and the nonlinear terms explicitly.

**3.1. Reference solution.** To test the data assimilation algorithms, we first compute a reference solution which is to be recovered using only coarse projections. The desired reference solution should exhibit some nontrivial time-dependent behavior to adequately test the performance of the algorithm. In addition, for practical matters, we would like to have a relatively simple force, which can be accurately represented in low resolution simulations.

We choose a two-mode force for each equation, which generates some interesting dynamics. Specifically, we define the forces,  $f, g : \Omega \rightarrow \mathbb{R}^2$ , by

$$f(x, y) = 2\Re \left( \left[ \begin{array}{c} 2 + 2i \\ (2 + 2i)/2 \end{array} \right] e^{i(x-2y)} + \left[ \begin{array}{c} -6 \\ 0 \end{array} \right] e^{i(3y)} \right) / M_f, \quad (11)$$

$$g(x, y) = 2\Re \left( \left[ \begin{array}{c} 4 - 3i \\ -2(4 - 3i)/3 \end{array} \right] e^{i(2x+3y)} + \left[ \begin{array}{c} -3 + 7i \\ (-3 + 7i)/5 \end{array} \right] e^{i(x-5y)} \right) / M_g, \quad (12)$$

where  $M_f$  and  $M_g$  are constants chosen so that  $\|f\|_{L^2} = \|g\|_{L^2} = 10$ . The modes were chosen essentially arbitrarily, as were the first components of the coefficients (the second components were then chosen to make the forces divergence free).

We next adjust  $\nu$  and  $\lambda$  to produce an interesting reference solution. We expect the flow to become more turbulent the smaller we take  $\nu$  and  $\lambda$ . However, decreasing  $\min\{\nu, \lambda\}$  may necessitate increasing the resolution of the computational grid, and in turn taking a smaller time step. As a compromise we take

$$\nu = \lambda = .01.$$

With the forces in (11) and (12), this yields the Grashof number

$$G = 10^5.$$

For simplicity, we set

$$\rho = \mu_0 = 1.$$

We define  $\|(u, b)\|$  for any semi-norm  $\|\cdot\|$  as

$$\|(u, b)\| := \sqrt{\|u\|^2 + \|b\|^2}. \quad (13)$$

We consider sufficient resolution to be achieved using  $256^2$  dealiased Fourier modes, because when computed with  $512^2$  modes, the qualitative features of the chaotic solution are essentially unchanged. All computations were performed using the time step  $dt = 0.0001$ .

Figure 1 shows some properties of the computed reference solution,  $(u, b)$ . The time series plotted in (a) and (b) shows the solution settling quickly upon a chaotic solution. In (c),(d) the projection in the  $L^2, H^1$ -plane of the solutions after the transient period indicates a qualitatively similar attractor for the two resolutions  $256^2$  and  $512^2$ . This projection is chosen to get a visual sense of the Taylor wave number  $\kappa_\tau^2 = \langle \|\nabla \cdot\|_{L^2}^2 \rangle / \langle \|\cdot\|_{L^2}^2 \rangle$ , and because there are explicit curves in this plane that bound the global attractor of the Navier-Stokes and related fluid systems (see [11]).

Figure 2 shows the curl of the velocity and magnetic fields of the computed reference solution at time  $t = 729.92$ , when the data assimilation starts. We see that there are several small eddies and complicated structures for the data assimilation algorithms to attempt to capture.

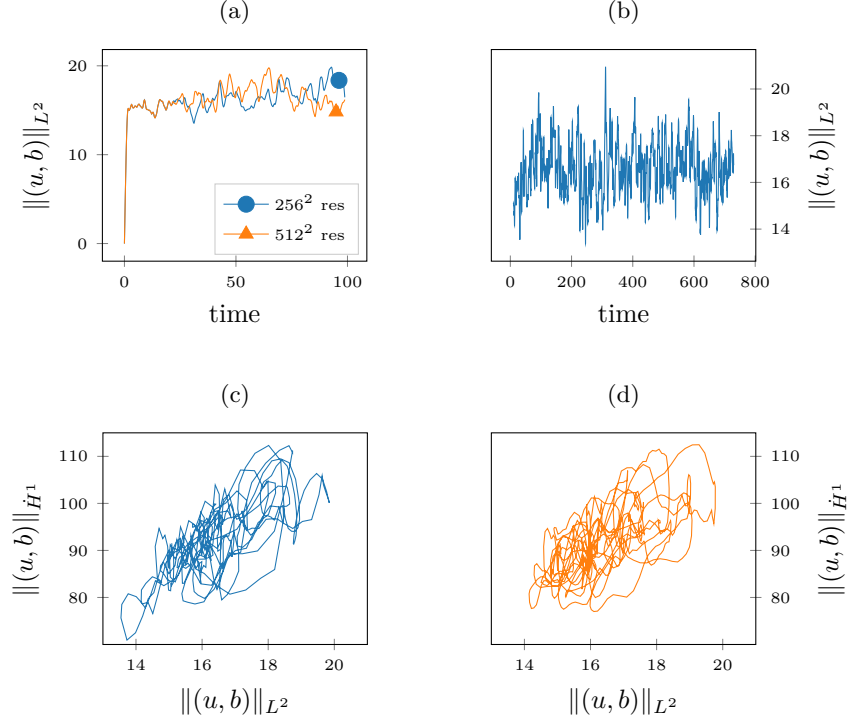


FIGURE 1. Properties of the reference solution. In (a) and (b) the evolution of the  $L^2$  norm of the reference solution computed with  $256^2$  resolution is shown for  $t \in [0, 100]$  and  $t \in [10, 792.92]$ . A comparison with  $512^2$  resolution is in (a). Plots of the  $H^1$  semi-norm vs the  $L^2$  norm of the solutions for  $t \in [10, 90]$  with  $256^2$  and  $512^2$  resolutions are in (c) and (d) respectively.

**3.2. Convergence.** All of the data assimilation simulations were performed starting from  $t_0 = 729.92$ , with initial data for the algorithm equal to 0, computing  $(U(t), B(t))$  simultaneously with  $(u(t), b(t))$  for  $t > t_0$ . We compare the resulting evolutions to see how the algorithm being tested performed in terms of the relative error, where we define the relative error as

$$\left( \|U(t) - u(t)\|_{L^2}^2 / \|u(t)\|_{L^2}^2 + \|B(t) - b(t)\|_{L^2}^2 / \|b(t)\|_{L^2}^2 \right)^{\frac{1}{2}}. \quad (14)$$

For each simulation, we choose an algorithm to generate  $(U, B)$ , and set  $N$  (so the number of modes of the reference solution for which we have data will be  $(2N+1)^2$ ), as well as  $\mu$ , which amplifies the feedback.

We would like to take  $\mu$  large, because the error in (8) and (10) decreases like  $e^{-c\mu t}$ , hence a larger  $\mu$  increases the convergence rate and therefore reduces the amount of simulation time required. However, if  $\mu$  becomes too large, it destabilizes the solution over small scales (small scales meaning finer resolutions than the larger, coarser scales captured by  $P_N$ ) by mixing in the feedback. This feedback is compensated by the dissipation, provided condition (6) holds.

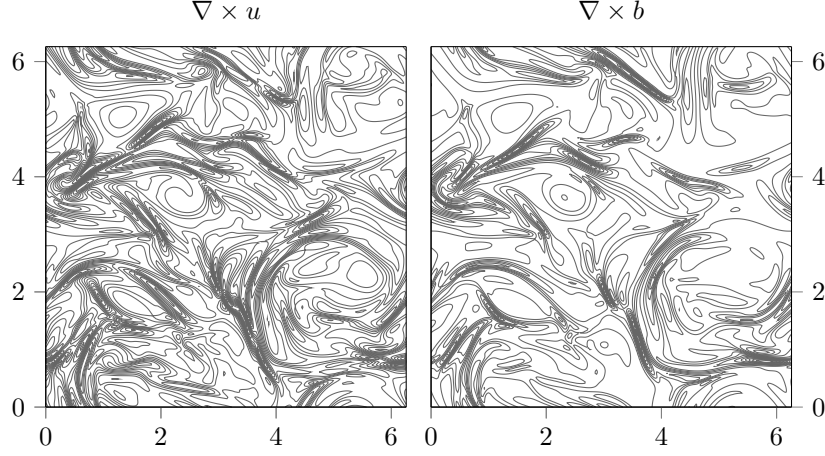


FIGURE 2. Contour lines of the curl of the computed reference solution at time  $t = 729.92$ .

Hence, given a value for  $N$ , we have an upper bound for  $\mu$  in the sense that, with  $\nu = \lambda = .01$ , (6) gives  $\mu \lesssim N^2/100$ . However, it is unclear what the optimal value for  $\mu$  is because of the constants involved and because (6) is a sufficient condition for convergence rather than a necessary condition. Figure 3 shows the resulting error after simulating Algorithms 2.2 – 2.4 for 5 simulation seconds, using different values of  $\mu$  and  $N$ .

We see that the benefit of increasing  $\mu$  (which is considerable initially) quickly diminishes in all cases. Furthermore, it appears that each algorithm performs as well with  $N = 32$  as with  $N = 128$  (note that  $N = 128$  constitutes using data on all the computational modes, as our  $256^2$  resolution is supported on a  $[-128, 128] \times [-128, 128]$  grid in Fourier space). Based on these results, we decide that  $\mu = 20$  is a reasonable value to use in longer simulations.

Figure 4 shows the convergence results we obtain with  $\mu = 20$  for Algorithms 2.2, 2.3, and 2.4, when  $N = 32$ , as well as the convergence of Algorithm 2.2 when  $N = 3$ . As expected, Algorithm 2.2 performs much better than Algorithm 2.3 and Algorithm 2.4. Perhaps surprisingly, Algorithm 2.4 performs noticeably better than Algorithm 2.3. Algorithm 2.3, though it does not perform as well, still shows near monotonic convergence, and it is reasonable to speculate that if the simulation time were extended, the error would reach  $10^{-9}$ , as do the other algorithms.

**4. Other algorithms.** We now test additional algorithms, which, unlike those in Section 3.2, have no rigorous estimates to support them. In each case we solve the nudged system for  $(U, B)$  on  $[t_0, t_0 + T]$ , with the initial condition  $U(t_0), B(t_0) \equiv 0$ .

**4.1. Different abridged algorithms.** In the following algorithm, we collect data on only the velocity field.

**Algorithm 4.1.** *Nudge*  $(u_1, u_2)$

$$\begin{aligned} \partial_t U - \nu \Delta U + (U \cdot \nabla) U - \frac{1}{\rho \mu_0} (B \cdot \nabla) B &= -\frac{1}{\rho} \nabla \tilde{p} + f + \mu P_N(u - U), \\ \partial_t B - \lambda \Delta B + (U \cdot \nabla) B - (B \cdot \nabla) U &= -\nabla q + g, \\ \nabla \cdot U &= 0, \nabla \cdot B = 0, \end{aligned}$$



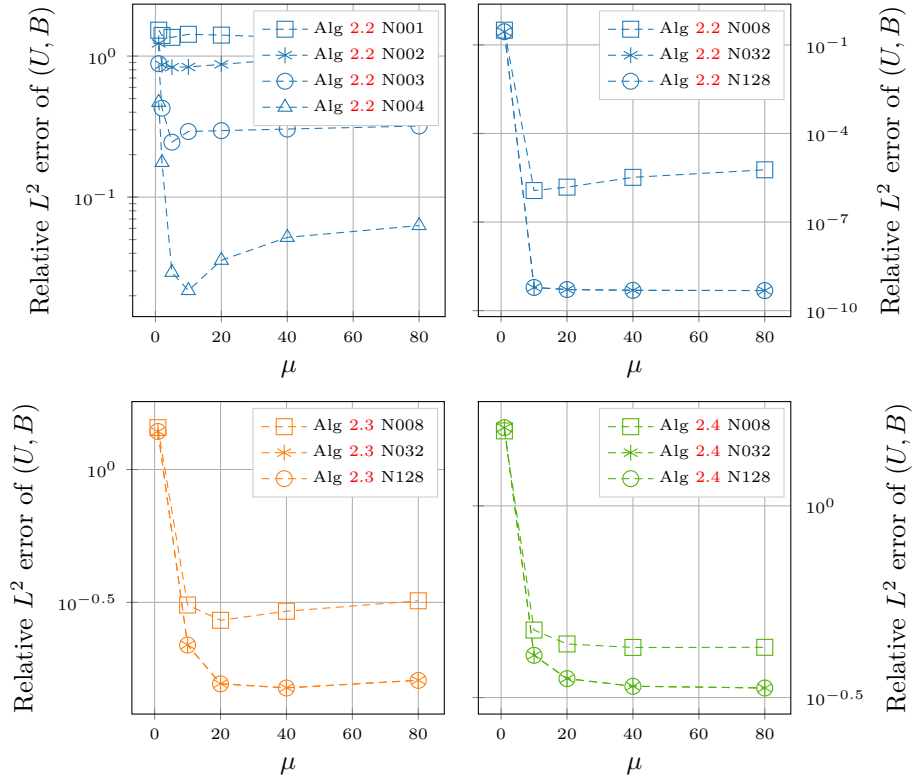


FIGURE 3. Dependence of the error (14) on  $\mu$  and  $N$ . The solutions were computed over the time interval  $[729.9, 734.9]$ , and the error is at time  $t = 734.9$ .

The next algorithm requires collecting data on only the magnetic field. Since it differs from Algorithm only in which variables are nudged, we specify simply by

**Algorithm 4.2.** *Nudge*  $(b_1, b_2)$

Note that because the evolution equation for  $B$  depends on  $U$  only through products with  $B$ , we might suspect that this algorithm will fail to recover the reference velocity  $u$ , as it would be impossible for the case that  $g \equiv b_0 \equiv 0$  (so  $b(t) \equiv 0 \forall t$ ), as was pointed out in [7]. However, we can still consider the possible success of this algorithm when the reference magnetic field is nonzero, which is our present situation.

We also consider how much of an improvement using data from 3 out of the 4 fields might yield compared to only using data from 2 fields. With this in mind, we define the following two algorithms.

**Algorithm 4.3.** *Nudge*  $(u_1, u_2, b_1)$

**Algorithm 4.4.** *Nudge*  $(u_1, b_1, b_2)$

Figure 5 shows the convergence results we obtain for the above four algorithms, using the same reference solution as before and  $\mu = 20$ . We see that Algorithm 4.2 shows no sign of converging, even with data on all the computational modes, i.e.

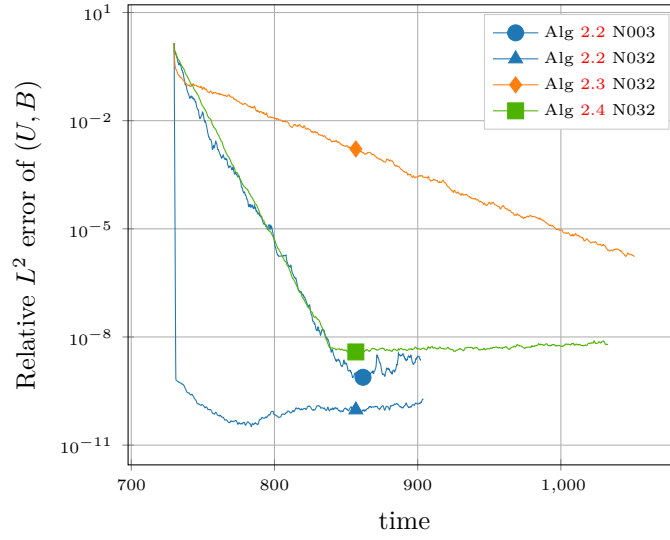


FIGURE 4. Convergence results for Algorithms 2.2, 2.3, and 2.4, with damping  $\mu = 20$ .

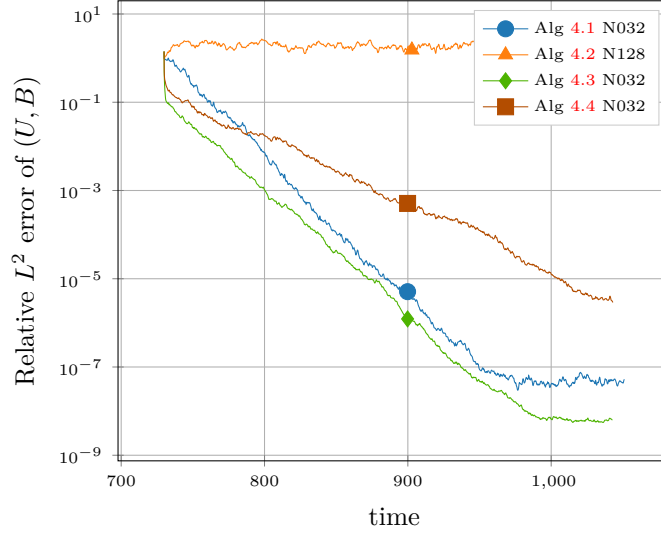


FIGURE 5. Convergence results for Algorithms 4.1-4.4 with  $\mu = 20$ .

$N = 128$ . Also, while Algorithm 4.4 shows little improvement over Algorithm 2.3, Algorithm 4.3 performs much better, although still worse than Algorithm 2.4. This is somewhat surprising, as Algorithm 2.4 performs better while requiring fewer measurements. We also note that Algorithm 4.1 seems to be converging. Although it was not proved, it was conjectured in [7] that such an algorithm would converge.

**4.2. Nonlinear nudging.** We demonstrated in Figure 3 how to empirically find the optimal value for  $\mu$  for a given amount of data. Thanks to the exponential rate of convergence, this can be done in relatively short computational time. Yet we can

avoid this by using an adaptive method which automatically adjusts the amplification of the nudging. In [27], Larios and Pei introduced the following nonlinear variation on (2):

$$\frac{d}{dt}Y = F(Y) - \mu \mathcal{N}(I_h(Y - y)), \quad (15a)$$

$$Y(0) = 0, \quad (15b)$$

where  $\mathcal{N}$  took several different forms, one being

$$\mathcal{N}(x) = \mathcal{N}_2(x) := \begin{cases} \|x\|^\gamma x, & \text{if } x \geq 1 \\ \|x\|^\gamma x, & \text{if } 0 < x < 1 \end{cases},$$

for some  $\gamma \in (0, 1)$  and some norm  $\|\cdot\|$ . In addition, they demonstrated for the Kuramoto–Sivashinsky equation that their approach achieved synchronization with the reference solution at a super-exponential rate.

We will now adapt their approach for the MHD using a modification of Algorithm 2.2 as an illustration. Each of the next four algorithms amount to solving

$$\begin{aligned} \partial_t \mathbf{U} - \nu \Delta \mathbf{U} + (\mathbf{U} \cdot \nabla) \mathbf{U} - \frac{1}{\rho \mu_0} (\mathbf{B} \cdot \nabla) \mathbf{B} &= -\frac{1}{\rho} \nabla \tilde{p} + f + \mu \mathcal{N}(u, \mathbf{U}), \\ \partial_t \mathbf{B} - \lambda \Delta \mathbf{B} + (\mathbf{U} \cdot \nabla) \mathbf{B} - (\mathbf{B} \cdot \nabla) \mathbf{U} &= -\nabla q + g + \mu \mathcal{N}(b, \mathbf{B}), \\ \nabla \cdot \mathbf{U} &= 0, \nabla \cdot \mathbf{B} = 0, \end{aligned} \quad (16)$$

for  $(\mathbf{U}, \mathbf{B})$  on  $[t_0, t_0 + T]$ , with the initial condition  $\mathbf{U}(t_0), \mathbf{B}(t_0) \equiv 0$ . The first nonlinear function we test closely resembles  $\mathcal{N}_2$ .

**Algorithm 4.5.** Solve (16) with  $\mathcal{N}$  defined by

$$\mathcal{N}(u, \mathbf{U}) = \begin{bmatrix} \psi(\|P_N(u_1 - \mathbf{U}_1)\|_{L^2} / \|P_N(u)\|_{L^2}) P_N(u_1 - \mathbf{U}_1) \\ \psi(\|P_N(u_2 - \mathbf{U}_2)\|_{L^2} / \|P_N(u)\|_{L^2}) P_N(u_2 - \mathbf{U}_2) \end{bmatrix}$$

(and  $\mathcal{N}(b, \mathbf{B})$  defined similarly), where

$$\psi(x) := \begin{cases} x^{0.1}, & \text{if } x \geq 1 \\ x^{-0.1}, & \text{if } 0 < x < 1 \end{cases}.$$

We note that the use of the  $L^2$ -norm in the nonlinear factor interferes with running the code in parallel. To address this we consider one which uses only Fourier modes and has the advantage then of being adaptive in both time and space. Again we specify in terms of velocity, since the expression for the magnetic field should be clear.

**Algorithm 4.6.** Solve (16) with  $\mathcal{N}$  defined by

$$\mathcal{F}(\mathcal{N}(u, \mathbf{U}))(k) = \begin{bmatrix} \psi(|\hat{u}_1(k) - \hat{\mathbf{U}}_1(k)| / |\hat{u}_1(k)|) (\hat{u}_1(k) - \hat{\mathbf{U}}_1(k)) \\ \psi(|\hat{u}_2(k) - \hat{\mathbf{U}}_2(k)| / |\hat{u}_2(k)|) (\hat{u}_2(k) - \hat{\mathbf{U}}_2(k)) \end{bmatrix} \quad \text{for } |k| \leq N,$$

and  $\mathcal{F}(\mathcal{N}(u, \mathbf{U}))(k) = 0$  for  $|k| > N$ , where  $\mathcal{F}$  denotes the Fourier transform and  $\psi$  is defined as in Algorithm 4.5.

**Remark 2.** We also considered variations on Algorithm 4.5 and Algorithm 4.6 which use a continuously adapted exponent determined by velocity, magnetic field, and even pressure data. Preliminary efforts however, have produced only schemes that seem unstable. This approach will be further studied in a future work, along with other schemes which make use of pressure data.

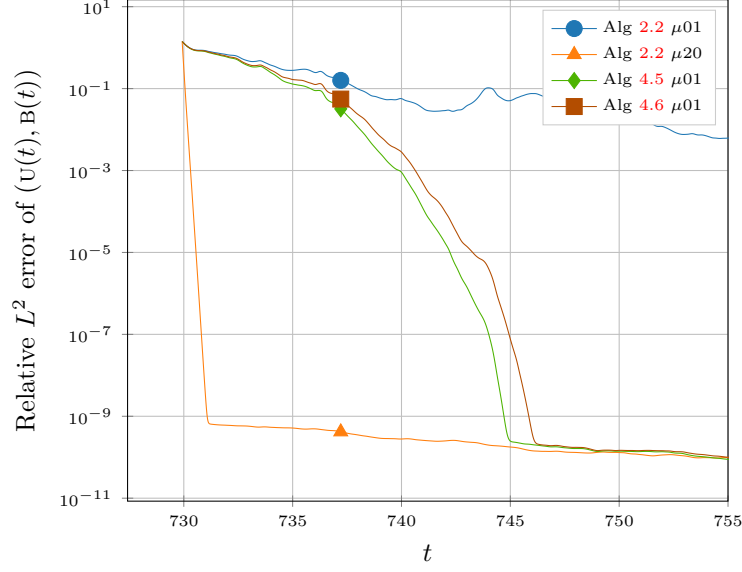


FIGURE 6. The evolution of the  $L^2$  error is shown for solutions of nonlinear modifications of Algorithm 2.2. Each simulation was performed with  $N = 32$ .

The results of using Algorithms 4.5 and 4.6 are shown in Figure 6, and compared to the rates of convergence we obtained previously with a static nudging constant  $\mu$ ; specifically, we compare to the cases where  $\mu = 1$  and  $\mu = 20$ . We see that the nonlinear schemes converge at a super-exponential rate, and are able to successfully synchronize with the reference solution without prior knowledge of a near optimal value for  $\mu$ .

**5. Measurement noise.** In most practical situations, measurement noise is inevitable. In this section we simulate the impact of such noise in the case of Algorithm 2.3. We do this by replacing each occurrence of  $P_N(u_i)$  and  $P_N(b_j)$  with the interpolation of the measured values including noise, which, in our case, are Fourier coefficients  $\mathcal{F}(P_N(u_i))(k) + N(0, \sigma_{u_i})$  and  $\mathcal{F}(P_N(b_j))(k) + N(0, \sigma_{b_j})$  respectively (for  $i, j \in \{1, 2\}$ ), where the standard deviations  $\sigma_{u_i}$  and  $\sigma_{b_j}$  gauge the precision of the instrumentation taking the measurements.

To simulate the noise, at each time step and for each measurement, we generate a random number from the normal distribution with mean zero and variance  $\sigma^2$ , divide it by the square root of the time step ( $\sqrt{10^{-4}} = 0.01$ ), and add the quotient to the exact value of the computed solution before using it in the data assimilation equation. For each evaluation we sample an independent random number  $\epsilon \sim N(0, \sigma)$  for each  $k \in \mathbb{Z}^2$ ,  $|k| \leq N = 32$ , set  $\tilde{u}_1(k) = \hat{u}_1(k) + \frac{\epsilon}{\sqrt{dt}}$ , and then use

$$x \mapsto \sum_{|k| \leq N} \tilde{u}_1(k) e^{ik \cdot x}$$

in the data assimilation equations instead of  $P_N(u_1)$ . The same is also done for each of the other fields:  $u_2, b_1$  and  $b_2$ . To ensure the integrity of the integration, we use a first order implicit/explicit scheme for the ODE solver at each time step.

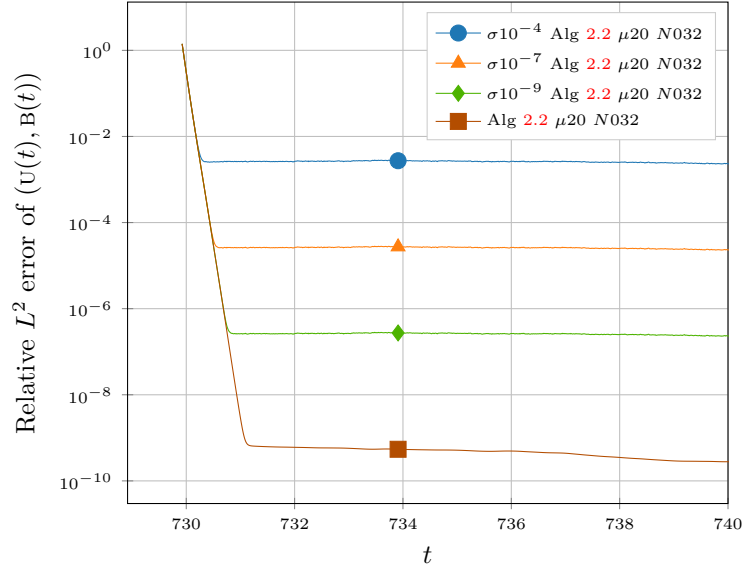


FIGURE 7. The evolution of the  $L^2$  error is shown for solutions of Algorithm 2.2 when subject to simulated measurement noise.

The results of including measurement noise are shown in Figure 7. The approximations successfully synchronize with the reference solution, down to a saturation level of noise that appears to be linear in the standard deviation of the measurement noise.

**6. Summary.** We have seen that data assimilation by nudging is highly effective for the MHD equations. In fact, it works with extremely coarse data when measurements are taken on all of the fields, even in the presence of measurement noise. Our results indicate that the Elsässer variable formulation is the most efficient way to use measurement data (without measuring every field), which leads us to suspect it may have deeper meaning. All of the abridged algorithms seem to work in our study, with the exception of the one which did not incorporate any measurements of the velocity field, as was conjectured in [7]. Thus, the vast collection of geomagnetic data alone seems insufficient for assimilation by nudging. On the other hand, an algorithm which incorporates only velocity field measurements is quite effective.

We also tested new algorithms which adaptively adjust the relaxation factor on the nudging term to amplify it as the error gets small. The straightforward adaptation of nonlinear nudging in [27] to the MHD system results in super exponential convergence, but due to a norm calculation, does not run well in parallel. We then demonstrated that a nonlinear nudging approach in terms of each Fourier mode converges at nearly the same super exponential rate. This final algorithm is not only adaptive in both time and space, but also well-suited to run in parallel.

**Acknowledgments.** The work of M.J. was supported in part by National Science Foundation (NSF) Grant DMS-1418911 and DMS-1818754, that of J.H. by NSF Grant DMS-1517027. Computational resources were provided by Lilly Endowment, Inc., through its support for the Indiana University Pervasive Technology Institute,

and in part by the Indiana METACyt Initiative. The Indiana METACyt Initiative at IU was also supported in part by Lilly Endowment, Inc.

## REFERENCES

- [1] D. A. Albanez, H. J. Nussenzveig Lopes and E. S. Titi, [Continuous data assimilation for the three-dimensional Navier–Stokes- \$\alpha\$  model](#), *Asymptotic Anal.*, **97** (2016), 139–164.
- [2] M. U. Altaf, E. S. Titi, T. Gebrael, O. M. Knio, L. Zhao, M. F. McCabe and I. Hoteit, [Downscaling the 2D Bénard convection equations using continuous data assimilation](#), *Comput. Geosci.*, **21** (2017), 393–410.
- [3] A. Azouani, E. Olson and E. S. Titi, [Continuous data assimilation using general interpolant observables](#), *J. Nonlinear Sci.*, **24** (2014), 277–304.
- [4] A. Azouani and E. S. Titi, [Feedback control of nonlinear dissipative systems by finite determining parameters—a reaction-diffusion paradigm](#), *Evol. Equ. Control Theory*, **3** (2014), 579–594.
- [5] H. Bessaih, E. Olson and E. S. Titi, [Continuous data assimilation with stochastically noisy data](#), *Nonlinearity*, **28** (2015), 729–753.
- [6] A. Biswas, C. Foias, C. F. Mondaini and E. S. Titi, [Downscaling data assimilation algorithm with applications to statistical solutions of the Navier-Stokes equations](#), *Ann. Inst. H. Poincaré Anal. Non Linéaire*, **36** (2019), 295–326.
- [7] A. Biswas, J. Hudson, A. Larios and Y. Pei, [Continuous data assimilation for the 2D magneto-hydrodynamic equations using one component of the velocity and magnetic fields](#), *Asymptot. Anal.*, **108** (2018), 1–43.
- [8] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet and B. Brown, Dedalus: Flexible framework for spectrally solving differential equations, Astrophysics Source Code Library, 2016.
- [9] E. Celik, E. Olson and E. S. Titi, [Spectral Filtering of Interpolant Observables for a Discrete-in-Time Downscaling Data Assimilation Algorithm](#), *SIAM J. Appl. Dyn. Syst.*, **18** (2019), 1118–1142.
- [10] J. Charney, M. Halem and R. Jastrow, [Use of incomplete historical data to infer the present state of the atmosphere](#), *Journal of the Atmospheric Sciences*, **26** (1969), 1160–1163.
- [11] R. Dascaliuc, C. Foias and M. Jolly, [Relations between energy and enstrophy on the global attractor of the 2-d Navier-Stokes equations](#), *J. Dynamics Differential Equations*, **17** (2005), 643–736.
- [12] D. Drake, C. Kletzing, F. Skiff, G. Howes and S. Vincena, Design and use of an Elsässer probe for analysis of Alfvén wave field according to wave direction, *The Review of Scientific Instruments*, **82** (2011), 103505.
- [13] G. Evensen, [Data Assimilation](#), 2nd edition, Springer-Verlag, Berlin, 2009, The ensemble Kalman filter.
- [14] A. Farhat, H. Johnston, M. Jolly and E. S. Titi, [Assimilation of nearly turbulent Rayleigh–Bénard flow through vorticity or local circulation measurements: a computational study](#), *J. Sci. Comput.*, **77** (2018), 1519–1533.
- [15] A. Farhat, M. S. Jolly and E. S. Titi, [Continuous data assimilation for the 2D Bénard convection through velocity measurements alone](#), *Phys. D*, **303** (2015), 59–66.
- [16] A. Farhat, E. Lunasin and E. S. Titi, [Abridged continuous data assimilation for the 2D Navier–Stokes equations utilizing measurements of only one component of the velocity field](#), *J. Math. Fluid Mech.*, **18** (2016), 1–23.
- [17] A. Farhat, E. Lunasin and E. S. Titi, [Data assimilation algorithm for 3D Bénard convection in porous media employing only temperature measurements](#), *Journal of Mathematical Analysis and Applications*, **438** (2016), 492–506.
- [18] A. Farhat, E. Lunasin and E. S. Titi, [On the Charney conjecture of data assimilation employing temperature measurements alone: The paradigm of 3d planetary geostrophic model](#), [arXiv:1608.04770](#).
- [19] A. Farhat, E. Lunasin and E. S. Titi, [Continuous data assimilation for a 2D Bénard convection system through horizontal velocity measurements alone](#), *J. Nonlinear Sci.*, **27** (2017), 1065–1087.
- [20] M. Gesho, E. Olson and E. S. Titi, [A computational study of a data assimilation algorithm for the two-dimensional Navier-Stokes equations](#), *Commun. Comput. Phys.*, **19** (2016), 1094–1110.

- [21] M. Ghil, M. Halem and R. Atlas, Time-continuous assimilation of remote-sounding data and its effect on weather forecasting, *Mon. Wea. Rev.*, **107** (1978), 140–171.
- [22] K. Hayden, E. Olson and E. S. Titi, [Discrete data assimilation in the Lorenz and 2D Navier-Stokes equations](#), *Phys. D*, **240** (2011), 1416–1425.
- [23] A. Jackson, A. R. T. Jonkers and M. R. Walker, [Four centuries of geomagnetic secular variation from historical records](#), *Phil. Trans. R. Soc. Lond.*, **358** (2000), 957–990.
- [24] M. S. Jolly, V. R. Martinez and E. S. Titi, A data assimilation algorithm for the subcritical surface quasi-geostrophic equation, *Adv. Nonlinear Stud.*, **17** (2017), 167–192.
- [25] M. S. Jolly, T. Sadigov and E. S. Titi, [Determining form and data assimilation algorithm for weakly damped and driven Korteweg–de Vries equation—Fourier modes case](#), *Nonlinear Anal. Real World Appl.*, **36** (2017), 287–317.
- [26] A. R. T. Jonkers, A. Jackson and A. Murray, Four centuries of geomagnetic data from historical records, *Rev. Geophys.*, **41**.
- [27] A. Larios and Y. Pei, Nonlinear continuous data assimilation, [arXiv:1703.03546](#).
- [28] D. Leunberger, [An introduction to observers](#), *IEEE Trans. Automat. Control*, **16** (1971), 596–602.
- [29] P. A. Markowich, E. S. Titi and S. Trabelsi, [Continuous data assimilation for the three-dimensional Brinkman–Forchheimer-extended Darcy model](#), *Nonlinearity*, **29** (2016), 1292–1328.
- [30] E. Olson and E. S. Titi, [Determining modes for continuous data assimilation in 2D turbulence](#), *J. Statist. Phys.*, **113** (2003), 799–840, Progress in statistical hydrodynamics (Santa Fe, NM, 2002).
- [31] S. Reich and C. Cotter, [Probabilistic Forecasting and Bayesian Data Assimilation](#), Cambridge University Press, New York, 2015.
- [32] F. E. Thau, [Observing the state of non-linear dynamic systems](#), *Int. J. Control*, **17** (1973), 471–479.

*E-mail address:* [Joshua.Hudson@jhuapl.edu](mailto:Joshua.Hudson@jhuapl.edu)

*E-mail address:* [msjolly@indiana.edu](mailto:msjolly@indiana.edu)