

Subgrid modeling for relativistic magnetohydrodynamics with machine learning

William Cook[✉] and Sebastiano Bernuzzi[✉]

Theoretisch-Physikalisches Institut, Friedrich-Schiller-Universität Jena, 07743, Jena, Germany



(Received 10 July 2026; accepted 11 August 2026; published 4 September 2026)

Resolving the impact of magnetic field instabilities in triggering small scale turbulent flow and the associated rearrangement of the field is of critical importance in understanding multimessenger observables in binary neutron star mergers and angular momentum transport in neutron stars and accretion disks. Direct simulation of these instabilities are unfeasible, however large-eddy simulations can incorporate the impact of this turbulence with a subgrid model. We present the first machine-learning-based subgrid model for special relativistic magnetohydrodynamics, trained using a neural network. We demonstrate its performance in online simulations of the 3D Kelvin-Helmholtz instability through both *a priori* and *a posteriori* tests. Evaluated in a low resolution simulation, our model captures magnetic field amplification of a simulation at 4 times the resolution with a speedup factor of 44. This demonstrates the applicability of such methods in general relativistic simulations of neutron star mergers and other scenarios.

DOI: [10.1103/t81t-vx23](https://doi.org/10.1103/t81t-vx23)

I. INTRODUCTION

Astrophysical fluids typically have incredibly high Reynolds numbers of order $\text{Re} \gtrsim 10^{10}$ and are thus susceptible to the onset of turbulent flow [1]. In the context of binary neutron star (BNS) mergers, turbulent flow is triggered by the onset of fluid and magnetic field instabilities such as the Kelvin-Helmholtz instability (KHI) [2] and the magnetorotational instability (MRI) [3] triggered as the stars first merge and in the postmerger remnant or accretion disk, respectively. The latter instability also plays a key role in angular momentum transport in an accretion disk around a black hole. The precise structure of the magnetic field in the postmerger remnant is unknown, but is driven by this turbulent flow and is believed to rearrange into a large scale structure that emerges from the remnant, capable of powering the launching of an astrophysical jet [4]. This jet is believed to generate γ -ray bursts [5–9], such as that observed coincident with the BNS merger observed in the gravitational wave spectrum, GW170817 [10–14].

Including the impact of the magnetic field in numerical simulations is vital to accurately predict multimessenger observables from BNS mergers [15–24] and in isolated neutron star spacetimes [25–29]. In a numerical simulation of a BNS, fully modeling this turbulent physics and capturing the nature of the magnetic field requires resolving

a turbulent cascade that may extend down to $\mathcal{O}(1 \text{ cm})$ [1]. Such length scales are impractical to resolve in direct numerical simulations (DNSs) that also hope to capture ejecta and gravitational waves on length scales of $\mathcal{O}(1000 \text{ km})$ and also widely span the possible parameter space. Recent work has shown however that such structure may be captured by zoom-in style simulations that stitch together evolutions at multiple length scales [30] or by short-lived incredibly high resolution simulations [23].

The impact of this small scale physics can be included into these large scale simulations however through the use of large-eddy simulations (LESs). LESs treat the evolution equations of (magneto)hydrodynamics [(M)HD] in finite volume form as the equations for “filtered” quantities, with the finite discretization length and volume averaging operation providing this implicit filter. These equations capture the evolution of eddies in the turbulent fluid on length scales larger than the filter length scale; while the impact of smaller scale eddies are captured by nonclosed terms in the equations arising from the noncommutation of the filtering operation with nonlinearities in the flux functions.

LESs then require the construction of a “subgrid model” to provide a closure of the equations. Such models have been constructed in a physics motivated manner by Smagorinsky [31], developed to include dynamical information [32], and through an analytic expansion of the equations in the gradient approach [33,34]. Extensions of such approaches to include magnetic fields have also been developed [35–39].

The application of such methods to astrophysical contexts is summarized in modern reviews such as [40,41], but here we focus on the relatively few applications to our area

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article’s title, journal citation, and DOI.

of interest, compact object merger and evolution. Such methods have been applied in the context of BNS mergers, initially by [42], utilizing the Smagorinsky subgrid model [31], with subsequent relativistic models based on the gradient approach developed [43–45]. A complementary approach is taken by the inclusion of viscous General Relativistic Hydrodynamics with a simplified Israel-Stewart method [46] or through the Bemfica-Disconzi-Noronha-Kovtun approach [47,48]. Effective models of magnetic field growth have also been incorporated through dynamo modeling [49].

Within the fluid dynamics community machine-learning (ML) techniques have been used to model turbulence [50] and build such subgrid models in nonrelativistic contexts [51,52]. It has been proposed recently by [53] that such techniques may also be used in magnetohydrodynamics (MHD), which may be applied to relativistic physics, and it is this approach that we follow in this work.

ML techniques have already been leveraged in the context of relativistic physics, with a recent comprehensive review of the field given in [54]. Applications of these techniques have come in detecting compact binary mergers and performing gravitational wave parameter estimation [55–57] and in building physics informed neural operators to solve the MHD equations [58–60]. Recent work has also demonstrated the first application of neural networks (NNs) in an online simulation of relativistic hydrodynamics, used to evaluate both the fluid equation of state and the conserved-to-primitive variable conversion [61]. Such methods have subsequently been tested for more complicated equations of state [62] and in a BNS simulation [63]. Recent work has also explored the use of ML techniques in building subgrid models to capture scale separation in galaxy evolution around supermassive black holes [64].

In this paper we follow the strategy of [53] and perform high resolution simulations of the KHI, in both Newtonian and special relativistic (SR) MHD, and train NNs on these data to provide a subgrid model for MHD turbulence. We then demonstrate that these models can be evaluated *a posteriori* within low resolution simulations of the KHI to match the behavior of high resolution simulations.

This paper is structured as follows. In Sec. II we discuss the filtered equations that we solve and the subgrid terms in Newtonian MHD and SR MHD. We then discuss the application of ML methods to the problem and discuss the structure of the NNs employed. In Sec. III we discuss the performance of our subgrid model, both in *a priori* tests of the accuracy of the NN in reproducing test data and in *a posteriori* studies, showing the ability of the model to accurately model the development of turbulence in online simulations. We show these tests first for Newtonian and then SR simulations. We summarize our findings in Sec. IV.

In the Newtonian case we work in units where $\epsilon_0 = \mu_0 = 1$ and absorb a factor of 4π into the magnetic field

definition, which are the standard units used in Athena++ [65]. In special relativity we additionally set $c = 1$.

II. METHOD

A. Filtering

When evolving equations using a finite volume approach the evolved variables are volume averaged, equivalently “filtered,” over a given computational cell. The size of this cell provides the smallest length scale that can be directly resolved in the simulation. In a turbulent flow, where eddies may cascade in size to smaller and smaller scales, only the eddies larger than this length scale can be resolved. In the LES approach, rather than viewing the implicit filtering of the fluid as an artifact of the numerical approach, we ask what are the exact equations that the filtered fluid obey. By filtering the evolution equations we obtain a system that describes the exact evolution of the fluid on length scales longer than the discretization length of our simulation, before any numerical approximations are made. The filtered fluid variables then directly correspond to the evolved variables in a finite volume simulation. However, the filtration operation does not commute with nonlinearities in the fluxes of our evolution equations and the fluxes will depend on the filtration of product terms. These are not evolved variables and require some knowledge of the solution to the equations on the subfilter scale. The system is therefore not closed and requires the prescription of these extra “subgrid terms” through a “subgrid model.” Below we derive the filtered evolution equations and associated subgrid terms for both Newtonian and SR MHD, before proceeding to generate such a subgrid model.

Let us first define the concept of filtering. We follow the standard approach for filtering in turbulent flows, as detailed in, e.g., [66–68]. We define a filtering kernel $G(r, x)$, where G is normalized such that $\int G(r, x) dr = 1$.

A filtered variable is defined as the convolution with the filtration kernel, i.e.,

$$\bar{f}(x, t) = \int G(r, x) f(x - r, t) dr. \quad (1)$$

Convolution with the filtration kernel commutes with t and x derivatives, as long as G does not depend on x , i.e., it is homogeneous. This is true, for instance, of a top hat box filter,

$$G(r) = \frac{1}{\Delta x} H\left(\frac{\Delta x}{2} - |r|\right), \quad (2)$$

where H is the Heaviside step function. We can consider this filtering operation to be the same as that which is applied when volume averaging a continuous variable over

a computational cell of width Δx , as in the standard finite volume approach to computational fluid dynamics.

By redefining $r = x - x'$, we recover the slightly more common expression for a filtered variable,

$$\tilde{f}(x, t) = \int_{x'_{\min}}^{x'_{\max}} G(x - x') f(x', t) dx'. \quad (3)$$

We define a second filtering operation, the Favre filter. A quantity filtered by the Favre filter is denoted with a tilde and defined as

$$\tilde{f} = \frac{\overline{\rho f}}{\bar{\rho}}, \quad (4)$$

where ρ is the fluid density. We note therefore trivially that $\overline{\rho f} = \bar{\rho} \tilde{f}$. We will employ this filtering operation in the Newtonian equations to follow and define the evolved velocity as the Favre filtered velocity. Consequently, there will be no subgrid terms in the density evolution equation.

B. Newtonian MHD

In Newtonian MHD the evolution equations for a magnetized fluid read

$$\partial_t \rho + \partial_i(\rho v^i) = 0, \quad (5)$$

$$\begin{aligned} \partial_t(\rho v^j) + \partial_i(\rho v^i v^j - B^i B^j \\ + \delta^{ij}(p + B^2/2)) = 0, \end{aligned} \quad (6)$$

$$\partial_t U + \partial_i \left(\left(U + p + \frac{B^2}{2} \right) v^i - (v_j B^j) B^i \right) = 0, \quad (7)$$

where we define

$$\tau_S^{ij} = \left(\bar{\rho} \tilde{v}^i \tilde{v}^j - \overline{\rho v^i v^j} \right) - \left(\overline{B^i B^j} - \overline{B^i B^j} \right) + \delta^{ij} \left(\bar{p} - \hat{p} + \frac{\bar{B}^2 - B^2}{2} \right), \quad (18)$$

$$= \bar{\rho} \left(\tilde{v}^i \tilde{v}^j - \overline{v^i v^j} \right) - \left(\overline{B^i B^j} - \overline{B^i B^j} \right) + \delta^{ij} \left(\bar{p} - \hat{p} + \frac{\bar{B}^2 - B^2}{2} \right), \quad (19)$$

$$\tau_U^i = \left(\bar{U} \tilde{v}^i - \overline{U v^i} \right) + \left(\hat{p} \tilde{v}^i - \overline{p v^i} \right) + \left(\bar{B}^2 \tilde{v}^i - \overline{B^2 v^i} \right) - \left(\overline{B^i \tilde{v}_j B^j} - \overline{B^i v_j B^j} \right), \quad (20)$$

$$\tau_B^{ik} = \left(\overline{B^k \tilde{v}^i} - \overline{B^k v^i} \right) - \left(\overline{B^i \tilde{v}^k} - \overline{B^i v^k} \right), \quad (21)$$

where in the second line we have applied the definition of the Favre filter to the product of velocities.

Explicitly, we consider a Gamma law equation of state, with $\Gamma = 4/3$, and so the pressure as calculated from the filtered variables is

$$\partial_t B^k + \partial_i(B^k v^i - B^i v^k) = 0. \quad (8)$$

Recall also the definition of U ,

$$U = e + \frac{\rho v^2}{2} + \frac{B^2}{2}. \quad (9)$$

These are the continuum equations, with no filtering or discretization procedures applied. We now filter these equations, integrating against the filtration kernel $G(x - x')$,

$$\partial_t \bar{\rho} + \partial_i(\overline{\rho v^i}) = 0, \quad (10)$$

$$\partial_t(\overline{\rho v^j}) + \partial_i \left(\overline{\rho v^i v^j} - \overline{B^i B^j} + \delta^{ij}(\overline{p + B^2/2}) \right) = 0, \quad (11)$$

$$\partial_t \bar{U} + \partial_i \left(\left(\overline{U + p + \frac{B^2}{2}} \right) v^i - \overline{(v_j B^j) B^i} \right) = 0, \quad (12)$$

$$\partial_t \overline{B^k} + \partial_i(\overline{B^k v^i} - \overline{B^i v^k}) = 0. \quad (13)$$

Now let us rearrange to form the LES equations, using the Favre filter for the velocity,

$$\partial_t \bar{\rho} + \partial_i(\bar{\rho} \tilde{v}^i) = 0, \quad (14)$$

$$\partial_t(\bar{\rho} \tilde{v}^j) + \partial_i(\bar{\rho} \tilde{v}^i \tilde{v}^j - \overline{B^i B^j} + \delta^{ij}(\hat{p} + \bar{B}^2/2) - \tau_S^{ij}) = 0, \quad (15)$$

$$\partial_t \bar{U} + \partial_i((\bar{U} + \hat{p} + \bar{B}^2) \tilde{v}^i - (\tilde{v}_j \bar{B}^j) \bar{B}^i - \tau_U^i) = 0, \quad (16)$$

$$\partial_t \overline{B^k} + \partial_i(\overline{B^k \tilde{v}^i} - \overline{B^i \tilde{v}^k} - \tau_B^{ik}) = 0, \quad (17)$$

$$\hat{p} = (\Gamma - 1)\hat{e}, \quad (22)$$

$$\hat{e} = \bar{U} - \frac{\bar{\rho}\bar{v}^2}{2} - \frac{\bar{B}^2}{2}. \quad (23)$$

C. Special relativistic MHD

For special relativity, conservation of mass, energy momentum, and the Maxwell equations in the ideal MHD approximation give the evolution equations for a magnetized fluid,

$$\partial_t D + \partial_i (D v^i) = 0, \quad (24)$$

$$\partial_t S_j + \partial_i \left(S_j v^i + \delta_j^i \left(p + \frac{b^2}{2} \right) - \frac{b_j B^i}{W} \right) = 0, \quad (25)$$

$$\partial_t \tau + \partial_i \left(\left(\tau v^i + \left(p + \frac{b^2}{2} \right) v^i - \frac{b^0 B^i}{W} \right) \right) = 0, \quad (26)$$

$$\partial_t B^k + \partial_i ((B^k v^i - B^i v^k)) = 0. \quad (27)$$

Here the conserved variables can be written in terms of the primitive variables as

$$D = \rho W, \quad (28)$$

$$S_j = (\rho h + b^2) W^2 v_j - b^0 b_j, \quad (29)$$

$$\tau = (\rho h + b^2) W^2 - \rho W - (p + b^2/2) - (b^0)^2, \quad (30)$$

where the Lorentz factor $W = \frac{1}{\sqrt{1-v^2}}$, and $h = 1 + \epsilon + p/\rho$ is the fluid enthalpy. The magnetic field in the comoving fluid frame b is related to the magnetic field as seen by the Eulerian observer B by

$$b^0 = W B^i v_i, \quad (31)$$

$$b^i = \frac{B^i}{W} + W B^j v_j v^i, \quad (32)$$

$$b^2 = \eta_{\mu\nu} b^\mu b^\nu = (B^j v_j)^2 + \frac{B^j B_j}{W^2}. \quad (33)$$

We filter these equations as above, obtaining

$$\partial_t \bar{D} + \partial_i (\bar{D} \hat{v}^i) = 0, \quad (34)$$

$$\partial_t \bar{S}_j + \partial_i \left(\bar{S}_j \hat{v}^i + \delta_j^i \left(\bar{p} + \frac{\bar{b}^2}{2} \right) - \left(\frac{\hat{b}_j \bar{B}^i}{\bar{W}} \right) \right) = 0, \quad (35)$$

$$\partial_t \bar{\tau} + \partial_i \left(\left(\bar{\tau} \hat{v}^i + \left(\bar{p} + \frac{\bar{b}^2}{2} \right) \hat{v}^i - \left(\frac{\hat{b}^0 \bar{B}^i}{\bar{W}} \right) \right) \right) = 0, \quad (36)$$

$$\partial_t \bar{B}^k + \partial_i \left((\bar{B}^k \hat{v}^i - \bar{B}^i \hat{v}^k) \right) = 0. \quad (37)$$

We can then rearrange to form the LES equations,

$$\partial_t \bar{D} + \partial_i (\bar{D} \hat{v}^i - \tau_D^i) = 0, \quad (38)$$

$$\partial_t \bar{S}_j + \partial_i \left(\bar{S}_j \hat{v}^i + \delta_j^i \left(\bar{p} + \frac{\hat{b}^2}{2} \right) - \left(\frac{\hat{b}_j \bar{B}^i}{\bar{W}} \right) - \tau_S^{ij} \right) = 0, \quad (39)$$

$$\partial_t \bar{\tau} + \partial_i \left(\bar{\tau} \hat{v}^i + \left(\bar{p} + \frac{\hat{b}^2}{2} \right) \hat{v}^i - \frac{\hat{b}^0 \bar{B}^i}{\bar{W}} - \tau_\tau^i \right) = 0, \quad (40)$$

$$\partial_t \bar{B}^k + \partial_i \left((\bar{B}^k \hat{v}^i - \bar{B}^i \hat{v}^k) - \tau_B^{ik} \right) = 0, \quad (41)$$

where

$$\tau_D^i = \bar{D} \hat{v}^i - \bar{D} v^i, \quad (42)$$

$$\begin{aligned} \tau_S^{ij} = & \bar{S}_j \hat{v}^i - \bar{S}_j v^i + \hat{p} - \bar{p} + \frac{\hat{b}^2}{2} - \frac{b^2}{2} \\ & - \left(\left(\frac{\hat{b}_j \bar{B}^i}{\bar{W}} \right) - \left(\frac{b_j B^i}{W} \right) \right), \end{aligned} \quad (43)$$

$$\begin{aligned} \tau_\tau^i = & \bar{\tau} \hat{v}^i - \bar{\tau} v^i + \hat{p} \hat{v}^i - \bar{p} v^i + \frac{\hat{b}^2 \hat{v}^i}{2} - \frac{b^2 v^i}{2} \\ & - \left(\frac{\hat{b}^0 \bar{B}^i}{\bar{W}} - \left(\frac{b^0 B^i}{W} \right) \right), \end{aligned} \quad (44)$$

$$\tau_B^{ik} = \bar{B}^k \hat{v}^i - \bar{B}^k v^i - (\bar{B}^i \hat{v}^k - \bar{B}^i v^k). \quad (45)$$

We note that, by rearranging the flux terms in the energy equation for τ , one can eliminate the subgrid term τ_τ^i and write this in terms of the subgrid term appearing in the equation for mass density τ_D^i . We have experimented with this option in our simulations and find no overall difference between these two implementations of the subgrid terms.

D. Model training with machine learning

1. Machine-learning approach

The subgrid tensor terms demonstrated in Eqs. (19)–(21) and Eqs. (42)–(45) for Newtonian and SR MHD, respectively, are not closed. While filtered individual conserved

variables are directly accessible in the simulation, since they are evolved variables, filtered products of variables are not. We therefore provide a closure, specifying these terms as functions of the fluid state. We take the approach suggested in [53] and widely applied in the fluid dynamics community to train a NN on the relationship between the state of the fluid (as inputs to the network) and the subgrid tensors (the outputs). We now outline the explicit procedure we follow and give concrete examples for the induction equation and associated subgrid tensor τ_B , specifically for Newtonian physics.

We initially perform a high resolution simulation of turbulent flow, here specifying to a local simulation of the KHI (see initial data below). We denote the conservative and primitive variables obtained in this simulation $\mathbf{q}^{\text{HR}}$, $\mathbf{w}^{\text{HR}}$. We treat this as the continuum solution and calculate the product terms that appear in subgrid tensors, e.g., $B^{i,\text{HR}} v^{k,\text{HR}}$. We now filter the high resolution data down to low resolution, $\bar{q} = \bar{\mathbf{q}}^{\text{HR}}$, by averaging neighboring cells, and filter the product terms also, $\overline{B^i v^k} = \overline{B^{i,\text{HR}} v^{k,\text{HR}}}$. Now we can construct the subgrid tensor at low resolution $\tau_B = (\bar{B}^k \tilde{v}^i - \bar{B}^i \tilde{v}^k) - (\bar{B}^i \tilde{v}^k - \bar{B}^k \tilde{v}^i)$. This is then the output on which we train our model, with the inputs given by the filtered individual terms (along with their spatial derivatives). Note the unique feature to Newtonian MHD, that the Favre filter is applied to the individual velocity term v . Our NNs are fully connected multilayer perceptrons, which we train using the PyTorch library [69].

Once our dataset is constructed we separate off 10% to be used as a validation dataset to track the loss function during training. We use a mean squared error loss function for all models. We perform our training for a maximum of 200 epochs and implement an early stopping criterion, such that, if for 5 epochs the validation error does not improve by more than 10^{-4} , the training terminates. The model at the epoch with the lowest validation error is then picked.

2. Conserved-to-primitive

In SR MHD the above algorithm must be slightly modified, since the conserved-to-primitive variable inversion is not linear and hence does not commute with the filtering operation. The primitive variables that are evaluated in the fluxes of a numerical simulation are calculated from the conserved variables and are not evolved variables themselves. When we come to discuss filtering therefore, they must be treated differently, since, when fluxes are evaluated within our evolution, we are using not the filtered version of the true primitives, but instead, the primitives that are calculated from the filtered conservatives. To make this explicit, let the set of conservatives be denoted q and the primitives w . w can be calculated from q through the nonlinear relationship $w = g(q)$. Define $\hat{w} = g(\bar{q})$, i.e., not the directly filtered primitives [which would be $\bar{w} = \overline{g(q)}$], but the primitives calculated from the filtered conservatives.

The nontrivial conserved-to-primitive inversion makes assembling the training data for the NN slightly more complicated than for the Newtonian case. Explicitly, returning to the above example of the induction equation, at high resolution the product term is calculated as above and filtered in the same way. We then filter the conserved variables from our high resolution simulation and perform a conserved-to-primitive conversion on those filtered variables to obtain the “low resolution” primitives, rather than directly filtering the “high resolution” primitives from the simulation. Hence the subgrid tensor $\tau_B^{ik} = \bar{B}^k \hat{v}^i - \bar{B}^i \hat{v}^k - (\bar{B}^i \hat{v}^k - \bar{B}^k \hat{v}^i)$.

3. Choice of input and outputs for NNs

Following the approach of [53], we use as inputs for the NNs the state of the fluid, as well as its spatial derivatives at a given spatial point, with the corresponding output, the subgrid tensor at that point. In [53], the inputs passed to the NN were not just these variables at a given point, but also those at neighboring grid points in a stencil of fixed size. In order to restrict the size of the models we train, and keep our online evolution of the model efficient, we do not use a stencil of neighboring points. In the Newtonian case we use only the primitive variables as input, since the conserved-to-primitive conversion is trivial, but in the SR case, we supply both the conserved and primitive variables, and their derivatives, as inputs, implicitly encoding the nature of the relationship between the two, and the equation of state, within the network. We train a separate model for each subgrid tensor, with each component of the tensor a different output of the network. During the training process both the inputs and outputs of the training dataset are normalized to a unit normal distribution.

4. Optimized evaluation

The aim of incorporating a subgrid model is to allow computationally inexpensive low resolution simulations to capture physics that would otherwise only be accessible in higher resolution, more expensive simulations. It is therefore necessary that the cost of evaluating the subgrid model is relatively cheap compared to the overall evolution speed, such that the low resolution simulation with the subgrid model activated is still cheaper than the equivalent high resolution simulation with no model. In other applications of NNs within relativistic hydrodynamics evolutions, it has been found that NNs may speed up the evolution compared to traditional methods [61], though other applications have found that in full production contexts there is no useful acceleration [63].

Here we attempt to implement our subgrid model in a manner that ensures a fast evaluation during the evolution. Our first goal is to ensure that models stay relatively small. We discuss the full range of hyperparameters we have explored in the Appendix, and here we note that we target

models with 6 or fewer hidden layers (N_L), and no more than 64 neurons per layer (N_n). This ensures that the matrix multiplications in the model evaluation are relatively cheap. We directly evaluate the models “by hand” during our evolution, loading the model weights and biases and evaluating the model using the matrix multiplication library EIGEN [70] to ensure fast, optimized evaluation of the network. We also use the mean and standard deviation of the training datasets to rescale the inputs and outputs of the model.

We evaluate the network at every substage in the multistage time evolution algorithm (RK3), and we enforce that symmetric subgrid terms are equal, i.e., we set $\tau_{ij} = \tau_{ji}$ where appropriate.

E. Initial data and numerical details

1. Numerical approach

Within this paper we consider two test problems, first a 2D KHI in Newtonian MHD and second a 3D KHI in SR. For the Newtonian problem we evolve the system using the code Athena++ [65], while the SR case is evolved with GR-Athena++ [71–75]. The numerical fluxes are constructed using a Local Lax-Friedrichs approximate Riemann solver and Piecewise Parabolic Method reconstruction, and time integration is performed with an RK3 integrator. All numerical grids considered are of uniform resolution, with a box side length of 2 in code units. For the Newtonian case, grid configurations LR and HR, corresponding to a low and high resolution respectively, have grid spacing $\Delta x = 2/(256, 512)$ while in the special relativistic case grid spacing $\Delta x = 2/(128, 256, 512)$ for grid configurations LR, SR, HR, respectively (here SR stands for standard resolution).

As can be seen from, e.g., Eqs. (38)–(41), the subgrid terms appear as additional contributions to the fluxes. We note that there are two options for how to evaluate these terms. First one can follow the approach we take here, evaluating these terms as additional contributions to the physical fluxes, but an alternate approach is to move these terms, with their associated derivatives, to the right-hand side and to treat them as source terms.

There are advantages and disadvantages to both approaches. The mass density evolution equation is written as an exact conservation law, which preserves conservation of mass to machine precision, up to the application of a floor, or outflows from a numerical grid. Moving subgrid terms to the right-hand-side breaks this explicit framing of the conservation law and may lead to violations of mass conservation. Athena++ and GR-Athena++ employ the constrained transport algorithm [76,77] to evolve the induction equation, which relies on writing the flux for the magnetic field as an exact curl, in order to preserve the divergence-free condition on the magnetic field to machine precision. Again, introducing source terms on the right-hand-side may introduce monopoles into our evolution, and so we retain

the subgrid terms within the flux calculation, rather than introducing them as sources on the right-hand-side.

The cost to be paid for retaining the subgrid terms within the fluxes comes in constructing the inputs to the NN. The NN depends on both the fluid state and its spatial derivatives, and these must be evaluated at the face centers of the computational grid, as this is where the flux is calculated. The primitive variables are already reconstructed to the face centers as is standard for the construction of the ordinary flux terms, however, now we must also reconstruct the derivatives of the primitives to the face centers.

To do this we perform a two step procedure. First, the spatial derivatives of the primitives are calculated at the cell centers using first order finite differencing, with a minmod limiter. Then the cell centered derivatives are reconstructed to the cell interface using donor cell reconstruction. In the implementation of the constrained transport algorithm, the evolution of the induction equation requires the calculation of the electric field at two grid samplings. First, at the face centered sampling, as part of the flux calculation described above, and second, at cell centers, from $E = -v \times B$. We also correct this term with the same subgrid term that occurs in the flux of the induction equation, though the input to the NN here is given by the cell centered primitives and their already calculated derivatives, rather than the face centered quantities. Once the subgrid corrected electric field at the faces and cell centers has been calculated, the constrained transport algorithm proceeds as normal, calculating the edge centered electric field and so integrating the magnetic field in time.

2. Newtonian initial data

For initial data for the 2D Newtonian problem we follow the initial data of [53], reproduced below,

$$\rho = \rho_0 + \rho_1 \text{sgn}(y) \tanh\left(\frac{|y| - y_l}{a_l}\right), \quad (46)$$

$$v^x = v_0^x \text{sgn}(y) \tanh\left(\frac{|y| - y_l}{a_l}\right) + \delta v^x \sin(2\pi n_x y), \quad (47)$$

$$v^y = \text{sgn}(y) \delta v^y \sin(2\pi n_y x) \exp\left(-\frac{(|y| - y_l)^2}{\sigma_y^2}\right), \quad (48)$$

$$B^x = B_0, \quad (49)$$

$$B^y = 0, \quad (50)$$

$$p = p_0, \quad (51)$$

with chosen parameter values $p_0 = 1$, $a_l = 0.01$, $\sigma = 0.1$, $y_l = 0.25$, $B_0 = 0.001$, $\rho_0 = 1.5$, $\rho_1 = -0.5$, $v_0^x = 0.5$, $\delta v^x = 0.01$, $n_x = 4$, $n_y = 7$, $\delta v^y = 0.2$.

3. Special relativistic initial data

For a fully turbulent 3D KHI in special relativity we follow the initial data specified by [45]. We initialize the primitive variables as follows:

$$\rho = \rho_0 + \rho_1 \text{sgn}(y) \tanh\left(\frac{|y| - y_l}{a_l}\right), \quad (52)$$

$$v^x = v_0^x \text{sgn}(y) \tanh\left(\frac{|y| - y_l}{a_l}\right) + \delta v^x \sin\left(\frac{2\pi n_x x}{L_x}\right), \quad (53)$$

$$v^y = v_0^y \text{sgn}(y) \tanh\left(\frac{|y| - y_l}{a_l}\right) + \delta v^y \sin\left(\frac{2\pi n_y y}{L_y}\right) \text{sgn}(y) \exp\left(-\frac{(|y| - y_l)^2}{\sigma_y^2}\right), \quad (54)$$

$$v^z = v_0^z \text{sgn}(y) \exp\left(-\frac{(|y| - y_l)^2}{\sigma_y^2}\right) + \delta v^z \sin\left(\frac{2\pi n_z z}{L_z}\right), \quad (55)$$

$$B^x = B_0, \quad (56)$$

$$B^y = B^z = 0, \quad (57)$$

$$p = p_0, \quad (58)$$

where the chosen parameter values are $B_0 = 0.001$, $p_0 = 1$, $\rho_0 = 2$, $\rho_1 = 1$, $y_l = 0.25$, $a_l = 0.01$, $v_0^i = 0.5$, $\delta v^x = 0.01$, $\delta v^y = 0.1$, $\delta v^z = 0.01$, $n_x = 11$, $n_y = 7$, $n_z = 5$, $\sigma_y^2 = 0.01$, $\sigma_z^2 = 0.1$.

III. RESULTS

A. Newtonian Kelvin-Helmholtz instability in 2D

1. Training dataset construction

We evolve the initial data detailed in Sec. II E 2 up to $t = 100$, ensuring that, for an extended period in the simulation, the initial configuration has been totally washed out and that a fully turbulent evolution has developed, at two resolutions, LR and HR. We only use data from after

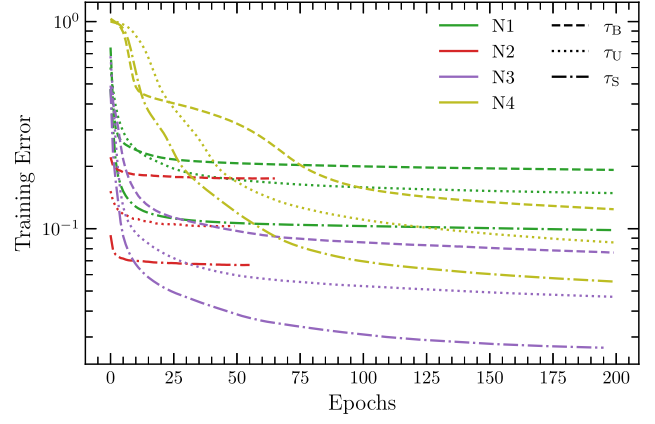


FIG. 1. The error of the NN during training. We show four models with different hyperparameters, selected by *a posteriori* criteria discussed in Sec. III A 3. Colors represent different models, while line styles represent the three networks trained for each model, corresponding to τ_B , τ_U , τ_S [Eqs. (19)–(21)].

this burn in period in the training dataset. Following the filtering procedure described in Sec. II D 1, we calculate the subgrid terms on the HR dataset and filter these data down to the LR resolution. We then perform the data augmentation procedure detailed in the Appendix to construct our training dataset of inputs (detailed in Sec. II D 3) and the corresponding subgrid terms.

2. Model training and *a priori* testing

We train a number of NNs on this training dataset, exploring a range of hyperparameters as detailed in the Appendix. In Fig. 1 we demonstrate the mean square error loss of four of our networks during training. We demonstrate model architectures which have been selected with *a posteriori* criteria defined below, with the model hyperparameters detailed in Table I. The stopping criteria for our training is detailed in Sec. II D 1, and we see that, for instance, model N2 terminates training comparatively early, after ~ 60 epochs, while all other models continue to train until the maximum 200 epochs. Despite this, the error of this model is lower than that of model N1, while the lowest error is found for model N3 across the three subgrid models constructed for the three subgrid tensors.

TABLE I. Hyperparameters for models of 2D Newtonian KHI.

Name	N_L	N_n	Act. func.	Initial LR	t_{start}	σ	Speed t/hr on 1 CPU	$B^2(t = 20)$
LR	...	...	...	...	...	...	7.32	7.58×10^{-4}
HR	...	...	...	...	...	...	0.964	3.78×10^{-3}
N1	2	16	tanh	10^{-4}	50	0	1.42	2.39×10^{-3}
N2	2	48	ReLU	10^{-3}	50	0	0.994	4.86×10^{-3}
N3	2	32	ReLU	10^{-3}	75	10	1.28	5.25×10^{-3}
N4	4	16	ReLU	10^{-4}	75	10	1.34	2.03×10^{-3}

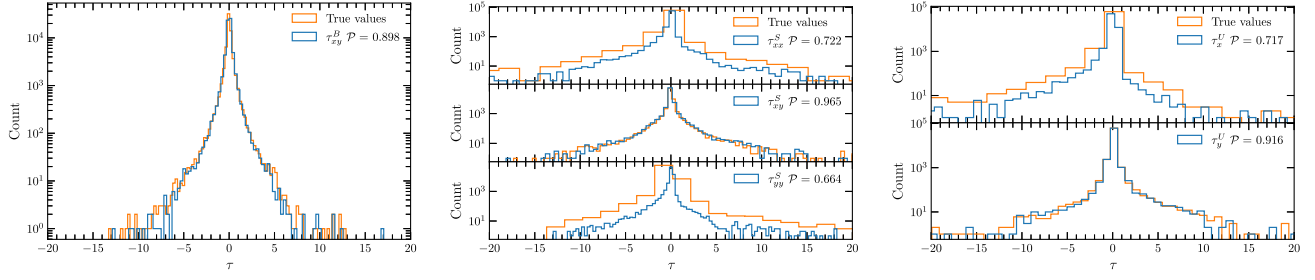


FIG. 2. Histogram of the subgrid tensors in the testing dataset (orange) and the predicted values of those tensors by the NN model N1. On the x-axis we show the value of the normalized subgrid tensor. On the y-axis we show the unnormalized count of points in the bins. Left, middle, and right panels show the subgrid tensors τ_B , τ_S , τ_U , respectively, while subpanels show different components of the tensor, indicated in the legend. In the legend we display the Pearson correlation coefficient between the two datasets.

We validate the *a priori* performance of our models by evaluating the model on a testing dataset drawn from a time slice of the HR simulation not used to generate the training dataset. The testing dataset consists of pairs of the fluid state at the filtered resolution (the input to the network) and the corresponding subgrid tensor values, calculated at higher resolution and then filtered (the outputs). We compare the model's predicted values of the subgrid tensors to the exact values in the dataset.

In Fig. 2 we show the distribution of values of each subgrid tensor in the testing dataset and the corresponding distribution predicted by network N1. We also calculate the Pearson correlation coefficient between the two datasets, denoted $\mathcal{P}$. A value of 1 corresponds to a perfect correlation, which we would expect for a model that perfectly generated the correct subgrid tensor values, while a value of 0 corresponds to uncorrelated data, which would denote a model generating random values. We see that our model performs well in this *a priori* test, with the predicted distributions matching the expected values well. We find

also that the correlation coefficient is well above zero, and over 0.7 for all but one component. The other models considered, N2, N3, N4 show very similar performance to model N1 in all components. The worst performing component in this test is τ_{yy}^S , which we ascribe to the comparatively small value of the momentum in the y direction, compared to the x direction due to the setup of our test problem. We may be able to further reduce this error in the future by training separate models for each component of the subgrid tensor, though the impact on performance in online evaluation of the network may make this improvement unfeasible.

3. *A posteriori* testing

To validate the *a posteriori* performance of the models, we perform simulations at LR with the subgrid terms activated, provided by a variety of models. In Fig. 3 we demonstrate the magnetic field strength at the end of the simulation at $t = 20$ for our fiducial LR and HR

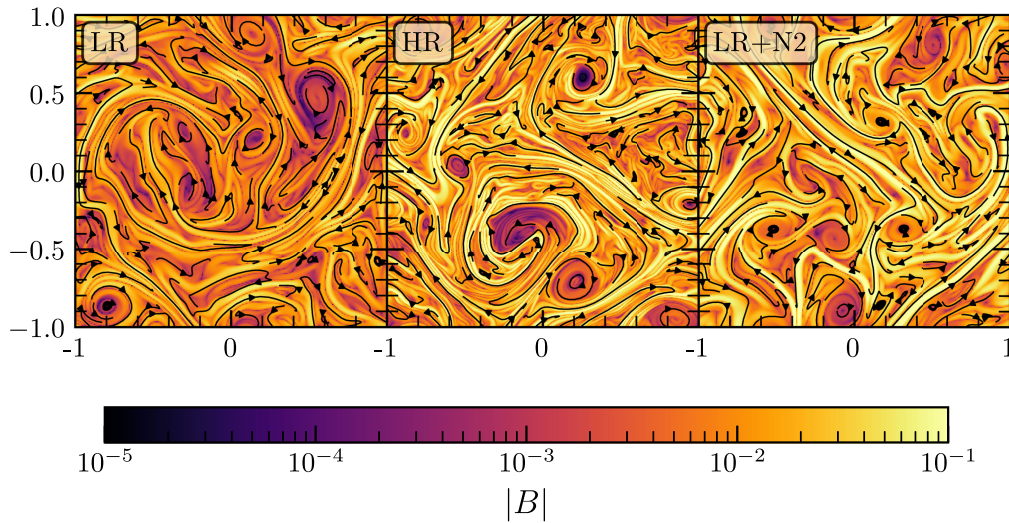


FIG. 3. The magnetic field strength at the end ($t = 20$) of the KHI simulation. Left and central panels denote the fiducial low and high resolution runs respectively, without a subgrid model. Right panel shows low resolution configuration with N2 model activated (hyperparameters in Table I). Streamlines of the magnetic field are shown in black.

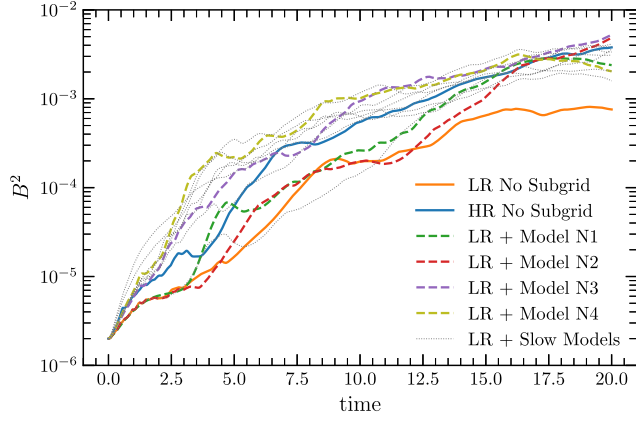


FIG. 4. Amplification of magnetic field energy with subgrid models. Solid lines represent reference runs without subgrid models, dashed lines represent runs with subgrid models that are faster than run HR. In dotted gray we demonstrate a selection of other models which successfully reproduce the amplification of run HR, but are slower to evaluate.

simulations, as well as one representative configuration at LR with a subgrid model activated, model N2. We note that the development of the KHI and associated turbulence is highly chaotic, and so we do not expect to see the low resolution run with subgrid model to exactly match the high resolution picture. We however do see that the amplification of the magnetic field, shown by the color map in Fig. 3, is of a comparable strength between the HR and LR+N2 models, both much stronger than the LR configuration, and that small scale turbulent features have developed in the magnetic field structure. We subsequently demonstrate the similarity between the HR and LR+N models with more detailed diagnostic quantities.

In Fig. 4 we demonstrate the amplification of the magnetic field due to the KHI for the two reference configurations with no subgrid model, LR and HR, compared with four models for different choices of hyperparameter.

From our search of the hyperparameter space we reproduce the four best performing models in the *a posteriori* test that are faster to execute than the HR configuration. Here we select models based on their ability to reproduce the final magnetic field amplification at $t = 20$. In gray we demonstrate a further 7 models that perform well in reproducing the magnetic field amplification, but which are slower to execute than model HR, thus giving no gain in performance. In Table I we detail the final magnetic field strength reached which consistently exceeds that of the low resolution run by at least a factor of 2.7 across the models considered, as well as the relative speeds of the simulations. These models are stochastically trained, and so we should expect that different models with different hyperparameters, and even different realizations of models with the same hyperparameters, should perform differently. Nevertheless, we find a range of hyperparameters that

successfully enhance the amplification of the magnetic field for resolution LR to a level comparable with that of resolution HR. Further, we find that a nontrivial subset of these hyperparameters gives us useful models that are sufficiently fast to evaluate that the LR run with subgrid model is faster than the corresponding HR run. This confirms that we can successfully use subgrid models from NNs to more efficiently capture the turbulent physics of the KHI than with direct high resolution simulations.

The speedups over configuration HR are, for our largest models, fairly small. For this 2D Newtonian problem the cost of evaluating the right-hand side of the evolution equations is relatively small, and so the impact of adding the NN evaluation has a large impact on the run-time especially for larger models. We test the impact of various hyperparameters on the run-time of the model and find two major observations. First, as expected, the evaluation of a more complex activation function slows the code down notably. The $\tanh(x)$ activation function (as taken from the standard C++ math library) is consistently $\sim 1.5\times$ slower to evaluate than the rectified linear unit (ReLU) activation function, and so only the smallest models are feasible with this activation function. Unsurprisingly N_n has the most significant impact on speed of execution, and of the model architectures we consider, it is clear that only models with 48 or fewer neurons per layer can give a faster performance than the HR configuration, with this threshold decreasing as the number of hidden layers increases.

We demonstrate also the spectrum of the magnetic field that develops by the end of the simulation, in Fig. 5. Clearly, the HR configuration is capable of capturing small scale magnetic field structure that is impossible to resolve in a LR run, no matter the presence of a subgrid model, due to the discretization length. However, we note, in line with other similar studies [43], that the addition of the subgrid terms successfully captures the growth of the magnetic field

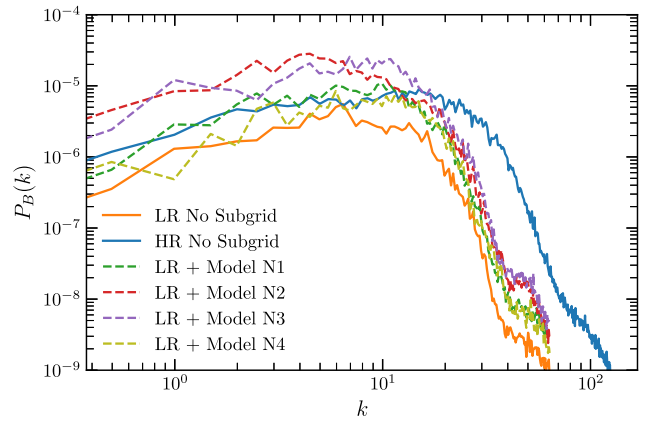


FIG. 5. Magnetic field spectrum at $t = 20$ in 2D Newtonian KHI. Solid lines represent reference runs without subgrid models, dashed lines represent runs with subgrid models, detailed in Table I.

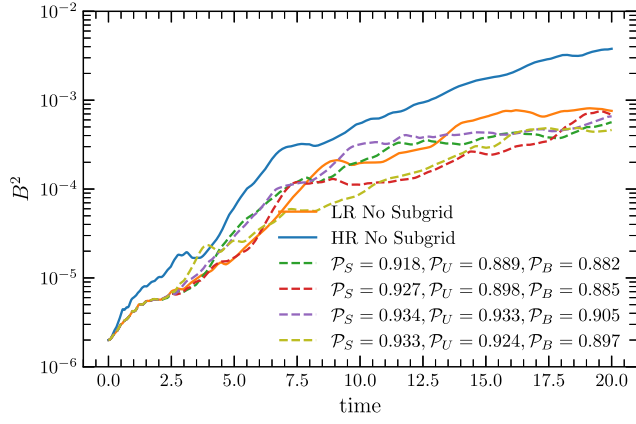


FIG. 6. Amplification of magnetic field energy with subgrid models with good *a priori* performance, but poor *a posteriori* performance. Solid lines represent reference runs without subgrid models, dashed lines represent runs with subgrid models. The average Pearson correlation coefficient for the components of each subgrid tensor are shown in the legend.

at longer wavelengths, present in the HR run and in the N1, N2, and N3 subgrid models.

The Pearson correlation coefficients demonstrated above for the models we have considered are less accurate than, for instance, those presented for the models considered in [53], or for the gradient models considered in [43]. We emphasize however that our selection criterion for models has been their *a posteriori* performance, not *a priori* performance. As an example of the relative importance of both of these tests, we demonstrate a second set of models in Fig. 6.

Here we see that for all models considered the amplification of the magnetic field energy is consistent with that of the LR configuration, and the expected amplification of the magnetic field to the HR level is absent. Compared to the models considered above, however, the average value of the Pearson correlation coefficient is much higher. A purely *a priori* selection criterion might select such models, however, it is clear that they do not capture the physical effect desired. This failure may be due to the models being well trained on the nonturbulent bulk of the training set, leading to a good performance in the *a priori* test, where the majority of points have small values of the subgrid tensor, which are accurately predicted; but poor performance in the tail of the dataset, where subgrid tensor values are large. This failure is not identified by the *a priori* test, but is by the *a posteriori* test. Since the *a priori* test is constructed from a dataset with no subgrid model present, only the *a posteriori* test can validate whether the model is capable of accurately evolving the fluid state that incorporates the model's own feedback into the evolution.

When compared to the gradient model implemented by Viganò *et al.* [43], our models perform well in *a posteriori* tests. In [43] the gradient model (applied to the 3D Newtonian KHI) can reproduce the evolution of the

magnetic field at double the considered resolution, with the addition of an extra multiplicative factor in front of the subgrid term, of ~ 16 . The magnetic amplification itself oscillates around the target amplification at double the resolution, much as the amplification shown in Fig. 4 does, with no extra factor included.

4. The role of individual subgrid tensors

In the results above we have looked at modeling all the subgrid terms that arise from noncommutation of the filtering operation with the nonlinear fluxes in the MHD equations. In contrast, other works constructing such models have focused on the impacts of specific terms. For instance in [42] only the subgrid term in the momentum equation is incorporated, while in [53] pressure-dependent terms are neglected. In the gradient approach of [45] all terms are considered, however, in production applications such as [15], only the term in the induction equation is incorporated, since in [78] it is argued that the most important term to include in the evolution to capture magnetic field amplification at the merger of a BNS system is the subgrid term in the induction equation, while other terms can suppress this amplification. We here show the role of each term, one by one in the evolution.

In order to determine the relative importance of these different terms we take our best performing models and run each simulation again, with only one subgrid term present at a time. We show the obtained amplification in Fig. 7. We make the following observations. Generally, the magnetic field amplification is largest for configurations with only the subgrid term in the induction equation present (only τ_B)

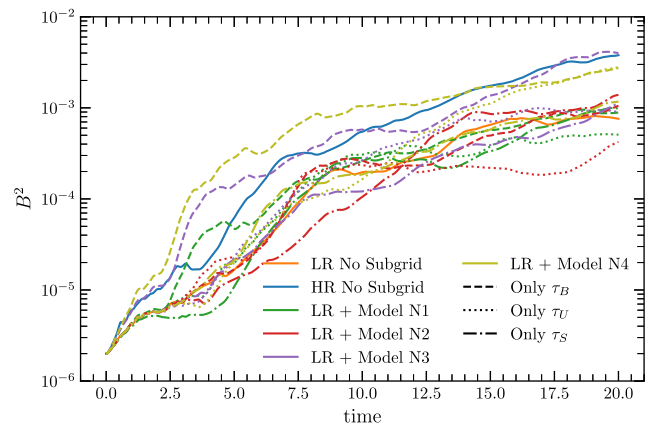


FIG. 7. Amplification of magnetic field energy with subgrid models only active in one evolution equation at a time. Solid lines represent reference runs without subgrid models, dashed lines represent runs with subgrid models only active in the induction equation [Eq. (21)], dotted lines represent runs with subgrid models only active in the energy equation [Eq. (20)], and dot-dashed lines represent runs with subgrid models only active in the momentum equation [Eq. (19)]. Colors represent the different models detailed in Table I.

configurations), as opposed to any other single term, supporting the suggestion of [78] that this is the term most important for the amplification. However, we also note that, for all models but model N3, it is possible to achieve amplification commensurate with the “only τ_B ” evolution by incorporating a different term, for instance, we see for model N4 that the final amplification for the only τ_B configuration is approximately identical to the “only τ_U ” configuration. It is also clear that for models N1 and N2, that, while the evolution with all subgrid terms active presented in Fig. 4 shows a large amplification consistent with the HR no-subgrid run, none of the evolutions with only one subgrid term presented in Fig. 7 show an amplification much larger than the LR no-subgrid evolution. From the array of models we consider therefore, it is difficult to conclude that one subgrid term can be prioritized over all other terms, and we suggest that all terms must be included to fully capture the turbulent physics in play.

A note of caution on our findings. We have presented a set of models picked due to *a posteriori* criteria, that is, models which, with all subgrid terms included, reproduce the amplification of the HR no-subgrid configuration. We enforce no selection criterion on the individual performances of the three NNs that comprise the overall model. It is therefore possible that, for any given model N?, the error budget is unevenly spread among the three networks, and only the combination of all networks gives an accurate result. However, when analyzing the errors demonstrated in Fig. 1, we can see that, while the magnitude of errors may differ between the models, the hierarchy of errors between the three networks is identical between the models, suggesting the partitioning of errors may be relatively consistent. We also note that model N3 has the lowest training error by the end of the training process. These facts lend slightly more robustness to the idea that τ_B may be the more important subgrid term, though again, it seems difficult to conclude that ignoring the other terms does not miss key physics, especially in the transport of angular momentum.

B. Special relativistic Kelvin-Helmholtz instability in 3D

We now repeat the above analysis generalizing to 3D and moving from Newtonian to SR physics.

1. Training dataset construction

We evolve the initial data detailed in Sec. II E 3 using the code GR-Athena++, up to $t = 40$ at resolutions LR, SR, and HR, corresponding to 128, 256, and 512 points in each direction over the domain. As in Sec. III A, we take the high resolution data, calculate the subgrid terms appearing in Eqs. (42)–(45), and filter this down to a lower resolution, along with the primitive variables and their spatial derivatives. These will respectively be used as the outputs and inputs of the NNs. We train using the same loss function,

stopping criteria, and maximum number of epochs as in the Newtonian case. Here we explore two types of models, based on different filtering factors. Models denoted f2 are constructed by treating the SR data as our continuum solution where the subgrid tensors are calculated, and then filtering these data down by a factor of 2 to the LR resolution, where the models are trained. Models denoted f4 are constructed by filtering instead the HR data down by a factor of 4 to the LR resolution. Practically, this filtering operation means that a neighboring cube of 8 (64) computational cells is averaged to a single cell for constructing the training dataset for the f2 (f4) models. In our *a posteriori* tests the f2 models should reproduce the behavior of the SR simulation, while the f4 models should reproduce that of the HR simulation.

We note a key difference between the Newtonian and special relativistic problems we consider, that the first is 2D and the latter 3D. Since a given time slice of data from our simulations is much larger in terms of computational cells in 3D, we have much more training data available to us, and therefore our dataset construction and data augmentation strategies are different compared to the 2D Newtonian case. As in the Newtonian case, we construct our training set from time slices taken after an initial burn in time t_{start} , in order to train on purely turbulent data, rather than data that carry the imprint of the initial conditions. Since the models we train are relatively small in terms of trainable parameters, we then reduce the datasets further, by taking a spatial subsample of each time slice in our training simulation after t_{start} , by selecting N_{sample} meshblocks at random from the full 3D data at each time slice. We refer the reader to [65,71] for a detailed description of the grid structure of GR-Athena++, but note that a meshblock is a spatially contiguous, logically rectangular spatial subdomain of the overall computational grid, which is, for all problems considered in this paper, 16^3 grid cells in size. We find that the size of these datasets removes the need to augment the datasets as we found in the 2D case. As before, we explore hyperparameter space, as described in the Appendix, training models that we will subsequently select based on their *a posteriori* performance.

2. Model training and *a priori* testing

Following the same training procedure as in the Newtonian case we monitor the online training error with a validation dataset separated from our training dataset. We demonstrate the model training error in Fig. 8, for a subset of successful models (according to the same *a posteriori* criterion as before) that we focus our analysis on. The hyperparameters of these models are presented in Table II.

We note that the training error in the energy term τ_ϵ , is consistently larger than that of the magnetic field, momentum, and density terms. Nevertheless, we find that in our *a posteriori* tests all models show an excellent conservation of energy, at least to a factor of one part in 10^3 , suggesting

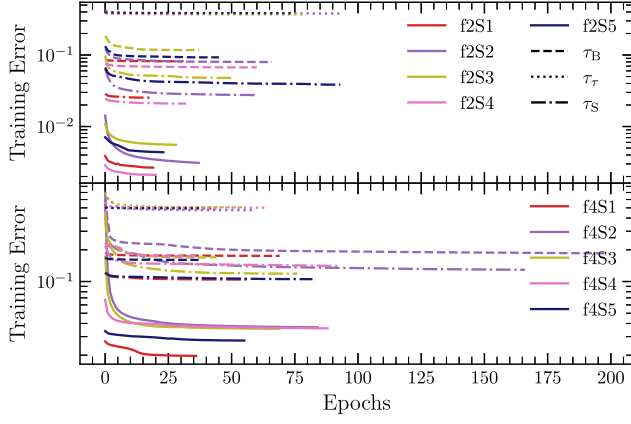


FIG. 8. Online training error given by mean squared error loss function on training dataset. Colors represent different models, while line styles represent models trained to reproduce different subgrid terms. Upper: f2 models. Lower: f4 models.

that this limitation to our model is not a significant source of error in our simulations. f2 models also show a lower error than their f4 counterparts, in agreement with their overall performance in the *a priori* test.

We demonstrate also the *a priori* performance of two of the trained models on testing datasets in histograms in Figs. 9 and 10. As above, we take a set of testing data that is independent of the training dataset, taken from a separate time slice within the same simulation, and demonstrate the ability of the model to accurately infer the subgrid term.

As in the Newtonian case, while we only show one case for each set of models f2 and f4, the other models within these families show similar behavior. First we note that, in comparison to the Newtonian case, many Pearson coefficients are much higher, in excess of 0.99, with all models other than τ_τ exceeding 0.83. We attribute this to the

different nature of the training dataset, with the large dataset no longer requiring a suppression of the mean and enhancement of the tails as detailed in the Appendix. As in the Newtonian case, the worst performing components are associated with the coordinate directions in which there are the smallest overall perturbations in the initial data, especially the τ_τ^z . As noted above, the worse performance of this term does not impact energy conservation. In comparison to the f2 models, the f4 models perform notably worse in the *a priori* test. This is consistent with results seen in the gradient model and for other NN implementations [43,53], in which the larger the filtering factor used, the worse the model performs in *a priori* tests. Since we select our models based on the *a posteriori* performance, these models are therefore not necessarily the best performers in the *a priori* test.

We note that the performance of the model is broadly consistent with similar plots produced in [53], where correlation coefficients are seen consistently in excess of ~ 0.95 for a filtering factor of 2 and in excess of ~ 0.85 for a factor of 4, though our worst performing components fall significantly below this threshold, a point which holds also for the earlier Newtonian models.

We attribute this worse performance to two reasons. First, in [53] inputs for a particular model are given by not only the primitives and their first derivatives, as employed here, but also the second derivatives, as well as all of these variables on a stencil of neighboring points. Taking this approach would increase the size of the input layer by a factor of 23, making the inference of the NN much more expensive. This would also require higher derivative terms and more intricate interpolation stencils to reconstruct these derivatives to the cell faces for flux construction within the numerical evolution code. In addition several terms are

TABLE II. Hyperparameters for models of 3D special relativistic MHD KHI. Models labeled f2 are trained on SR data, and f4 on HR data. Speed up to target is measured against SR for f2 models and against HR for f4 models.

Name	N_L	N_n	Act. func.	Initial LR	t_{start}	N_{sample}	Speed on 1 CPU (t/hr)	$B^2(t=20)$	Speed up to target	$N_{\text{mesh,eff}}$
LR	...	...	...	...	...	...	3.80×10^{-2}	2.43×10^{-4}	...	128
SR	...	...	...	...	...	...	0.333×10^{-2}	1.48×10^{-3}	...	256
HR	...	...	...	...	...	...	0.0296×10^{-2}	4.31×10^{-3}	...	512
f2S1	2	32	ReLU	10^{-3}	30	1000	1.39×10^{-2}	1.88×10^{-3}	4.17	270
f2S2	2	64	ReLU	10^{-3}	35	200	0.931×10^{-2}	1.75×10^{-3}	2.80	288
f2S3	3	16	ReLU	10^{-3}	20	50	1.54×10^{-2}	1.62×10^{-3}	4.63	159
f2S4	4	16	ReLU	10^{-3}	20	100	1.34×10^{-2}	1.26×10^{-3}	4.02	175
f2S5	4	16	ReLU	10^{-3}	30	200	1.31×10^{-2}	1.52×10^{-3}	3.94	285
f4S1	2	32	ReLU	10^{-3}	20	2000	1.20×10^{-2}	4.02×10^{-3}	40.5	508
f4S2	3	16	ReLU	10^{-3}	35	100	1.29×10^{-2}	4.75×10^{-3}	43.5	541
f4S3	3	32	ReLU	10^{-3}	35	100	0.986×10^{-2}	3.81×10^{-3}	33.3	528
f4S4	4	16	ReLU	10^{-3}	35	500	1.14×10^{-2}	4.46×10^{-3}	38.6	428
f4S5	4	32	ReLU	10^{-3}	30	50	0.840×10^{-2}	5.46×10^{-3}	28.4	409

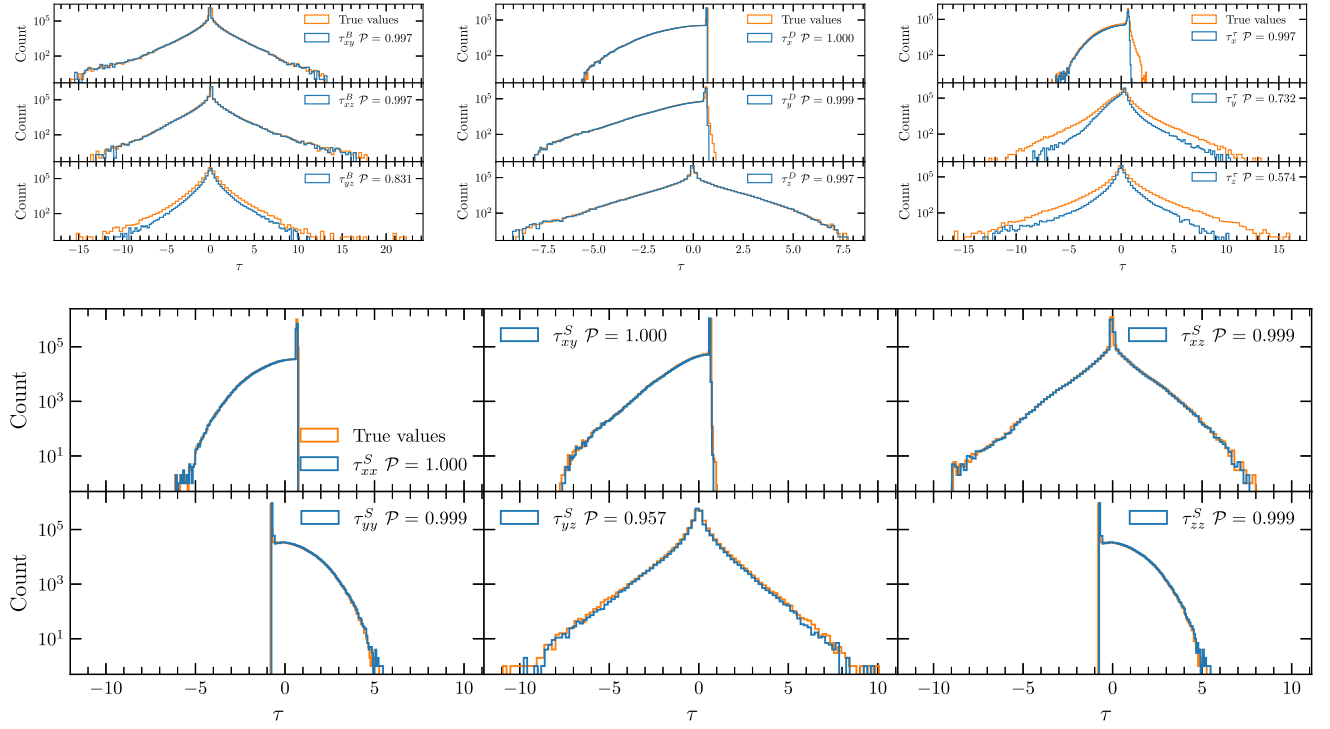


FIG. 9. Histogram of the subgrid tensors in the testing dataset (orange) and the predicted values of those tensors by the NN model f2S1. On the x-axis we show the value of the normalized subgrid tensor. On the y-axis we show the unnormalized count of points in the bins. Left, middle, and right upper panels show the subgrid tensors τ_B , τ_D , τ_τ , respectively, while the lower panel shows τ_S . Subpanels show different components of the tensor, indicated in the legend. In the legend, we display the Pearson correlation coefficient between the two datasets.

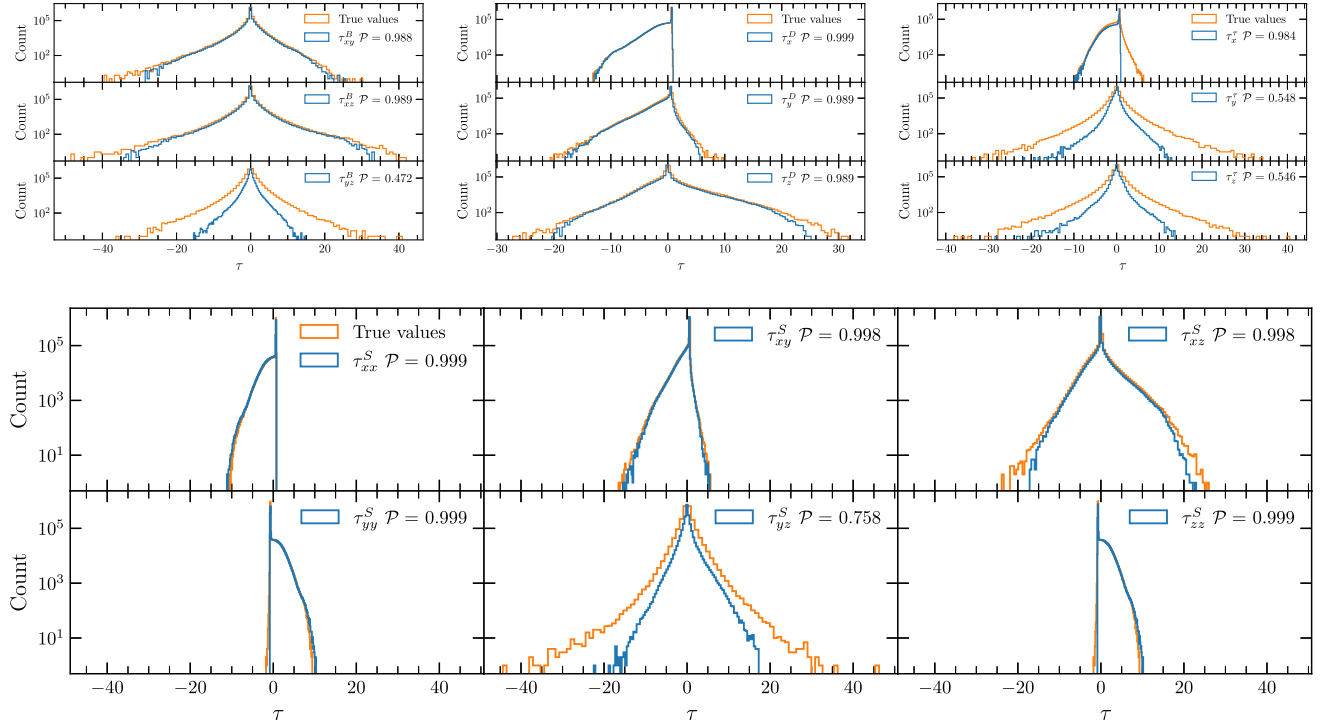


FIG. 10. Histogram of the subgrid tensors in the testing dataset (orange) and the predicted values of those tensors by the NN model f4S1. On the x-axis we show the value of the normalized subgrid tensor. On the y-axis we show the unnormalized count of points in the bins. Left, middle, and right upper panels show the subgrid tensors τ_B , τ_D , τ_τ , respectively, while the lower panel shows τ_S . Subpanels show different components of the tensor, indicated in the legend. In the legend, we display the Pearson correlation coefficient between the two datasets.

neglected in the subgrid tensors constructed in [53], for instance, all terms with a pressure dependence.

Here we prioritize generating small models, capturing all of the relevant physics, and tolerate the associated error in order to achieve performant evolutions that can capture the high resolution physics more efficiently than by performing the higher resolution DNS. We note however that our current simulations are in special relativity. In future applications, in general relativity, a given time step of our code will be much more expensive due to the requirement to solve the Einstein equations. This extra cost will mask the cost of the NN evaluation much more, allowing us to train much larger models, without risking making the code too inefficient. In this scenario it may be that using a wider stencil and higher derivatives gives a valuable accuracy improvement with minimal performance impact. Again, however, we reemphasize the point of Fig. 6, that merely increasing the *a priori* correlation is not a guarantee of successful performance in the *a posteriori* test.

3. *A posteriori* testing

We perform simulations at resolution LR with the trained models from both the f2 and f4 families. We then select the best performing models from our *a posteriori* test in terms of their ability to reproduce the final magnetic field

amplification of either the SR or HR no-subgrid run. The best performing models and their hyperparameters are listed in Table II.

In Figs. 11 and 12 we show 2D slices of the full 3D evolution for models LR, SR, and HR and two subgrid models f2S1, f4S1 at three characteristic time slices. These are spatial slices taken at $x = 0$ to show the full 3D evolution.

In Fig. 11 we see the evolution of the density. By the middle time slice $t = 10$ we see that the initial conditions have been washed out and that a turbulent state has developed. By the end of the simulation we see that structure cascades to smaller length scales, best captured at high resolution.

In Fig. 12 we demonstrate the evolution of the magnetic field. We see that, while the SR and HR simulations are clearly capable of capturing the small scale features in the magnetic field that cannot be directly resolved in any of the LR simulations, the magnitude of the magnetic field, as shown by the color scale, is comparable between the SR and f2S1 runs and between the HR and f4S1 runs.

In Fig. 13 we directly demonstrate the amplification of the magnetic field. In the upper panel we demonstrate the performance of the f2 models, which aim to reproduce the behavior of the SR run. We see that the four selected models all demonstrate qualitatively similar behavior.

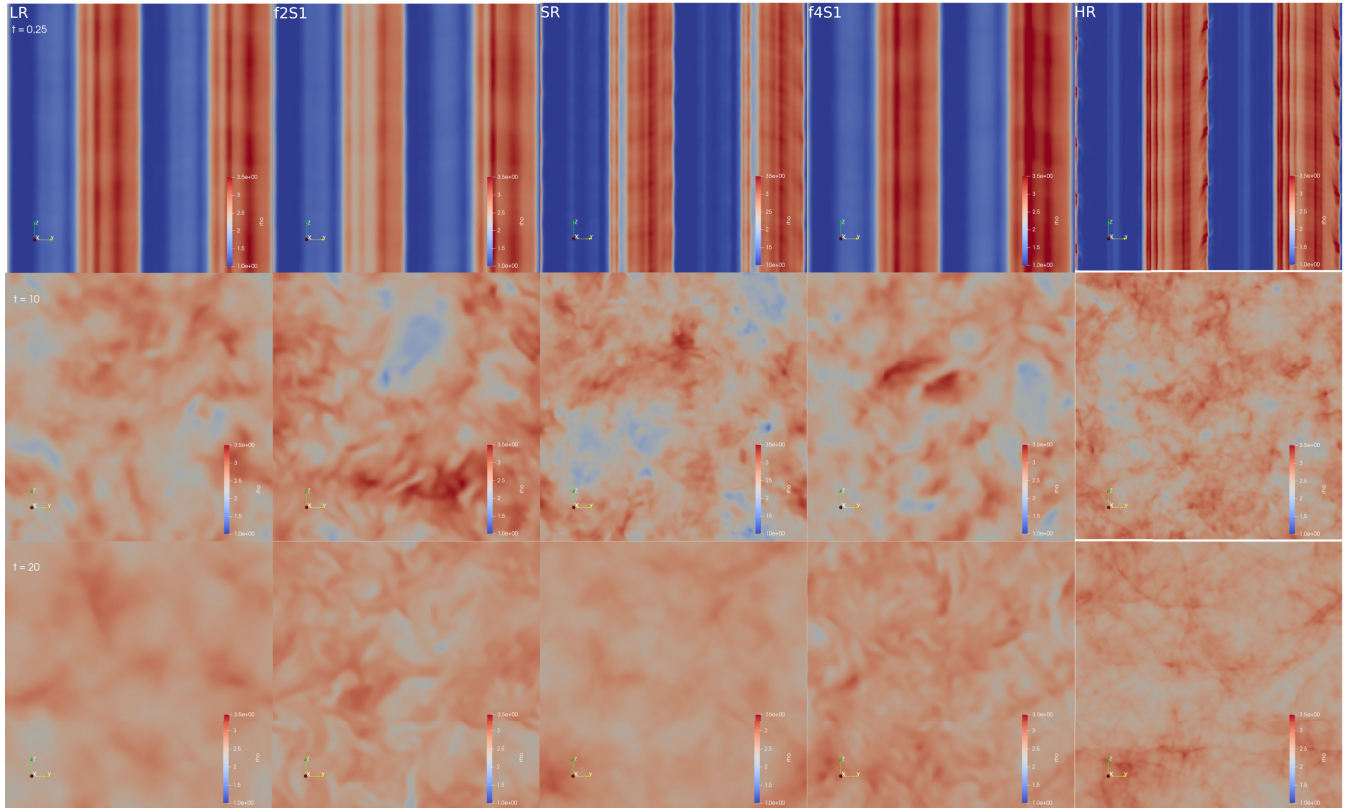


FIG. 11. 2D slices of 3D evolution of ρ for special relativistic KHI evolution. The three rows demonstrate different time slices $t = (0.25, 10, 20)$. The five columns show the models LR, f2S1, SR, f4S1, HR, respectively.

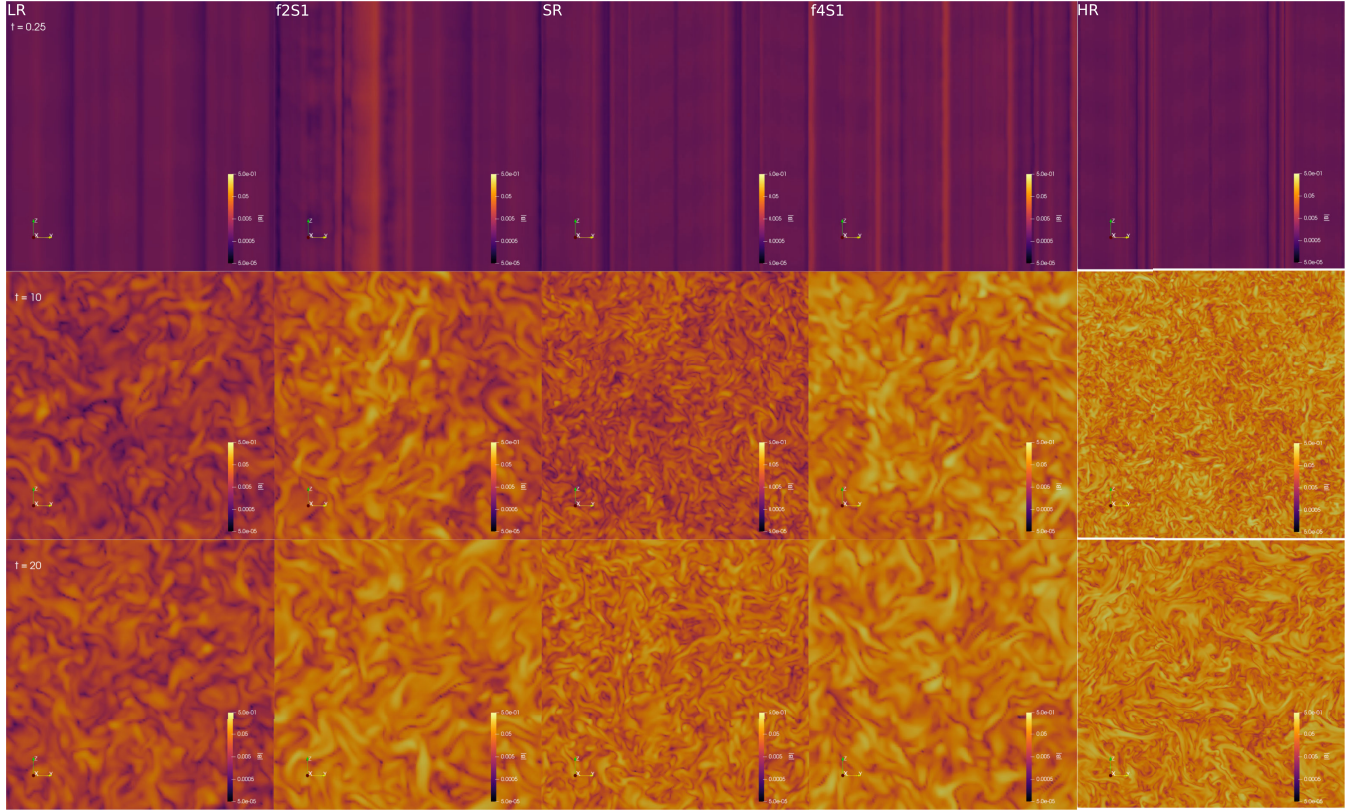


FIG. 12. 2D slices of 3D evolution of $|B|$ for special relativistic KHI evolution. The three rows demonstrate different time slices $t = (0.25, 10, 20)$. The five columns show the models LR, f2S1, SR, f4S1, HR, respectively.

At early times the models amplify faster than their target run, which is still dominated by the impact of the initial data, but that between $t \sim 2.5$ and 10 the simulations evolve with a similar growth rate of the magnetic field. At $t \sim 10$ the simulations saturate at a similar magnetic field strength.

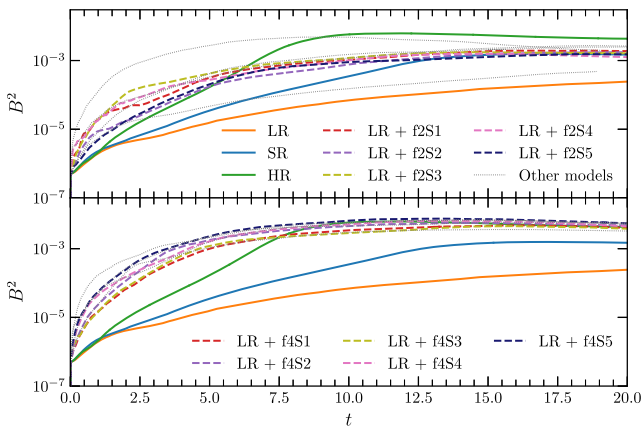


FIG. 13. Amplification of the magnetic field with subgrid model in 3D special relativistic KHI test. Solid lines denote fiducial runs with no subgrid model. Dashed lines show the subgrid models described in Table II. Gray dotted lines are other models in hyperparameter space. Upper: f2 models trained on SR data. Lower: f4 models trained on HR data.

We detail the final field strength in Table II, which we find to be consistent with the SR run to between 3% and 27%. Comparing the speed of evolution we see that these configurations are between ~ 3 and 5 times faster than the SR configuration. In the lower panel we repeat this analysis for the f4 models. The same overall behavior is seen, a fast initial growth before saturating at the target HR value. The final error in the magnetic field is also between 3% and 27% with a speedup between a factor of 28 and 44. As above, we have selected from our trained models those that perform well in the *a posteriori* testing. However, we demonstrate in dotted gray lines a broader set of other models that also perform well. This demonstrates that well performing models are not rare in hyperparameter space. Unlike in the Newtonian 2D case we do not denote these as slow models since in the more computationally taxing special relativistic 3D case, all models are faster than the SR or HR simulations.

In all models we consider, the amplification appears earlier than in the target no-subgrid run. We ascribe this behavior to the fact that we train our models using time slices from the end of the simulation, to avoid carrying the imprint of the initial conditions. Therefore the NN is biased toward generating the final turbulent state as quickly as possible. For future applications we will be interested in applying this technique in real physical scenarios, therefore

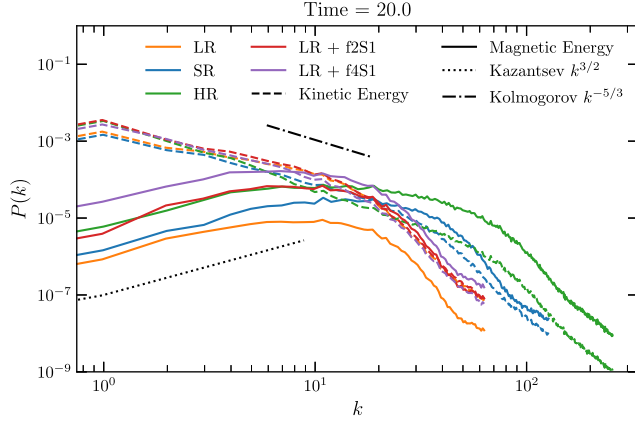


FIG. 14. Spectrum at $t = 20$ of magnetic (solid lines) and kinetic (dashed lines) energies. For visualization purposes the Kolmogorov $k^{-5/3}$ and Kazantsev $k^{3/2}$ scalings are shown with dot-dashed lines. Different models are denoted by different colors.

we are not interested in training models that perform well in the early phase where the evolution is dominated by the initial data from which the KHI develops.

To measure the accuracy of the model we must validate that it not only captures the final amplification of the magnetic field associated with the higher resolution simulation, but also that the structure of the magnetic field is consistent with this simulation. In Fig. 14 we demonstrate the power spectrum of the magnetic and kinetic energies. In the presence of the turbulence induced by the KHI, we expect to see that a Kolmogorov cascade emerges in the kinetic energy, demonstrating the well-known $k^{-5/3}$ power law scaling [79]. Associated with this cascade of kinetic energy, we expect to see a dynamo process emerging, associated with a $k^{3/2}$ power law scaling following Kazantsev [80]. We see that, by the end of our simulation, all runs demonstrate power law scaling in the kinetic spectra consistent with the Kolmogorov scaling and in the magnetic field consistent with the Kazantsev scaling. In particular, this verifies that the subgrid models successfully reproduce the dynamo process at work. Clearly, due to the smaller length scales they resolve, we see that the SR and HR simulations capture the magnetic field structure at larger values of k better than the subgrid simulations, and that the larger magnetic field strength in the SR and HR compared to LR simulation is associated mostly with the magnetic field that is captured at smaller scales and, to a lesser extent, the larger magnetic field that develops on longer wavelengths through the dynamo process. In contrast, the subgrid models cannot directly capture the magnetic field on these smaller scales as they are limited by resolution. Therefore, in order to match the magnetic amplification of SR (HR), f2(4)S1 must achieve more amplification at longer wavelengths, as seen in Fig. 14. Then in comparison to LR, model f2(4)S1 shows considerable amplification at all wavelengths, with the overall

shape of the spectra largely unchanged. We see that the kinetic spectra are largely similar between all models with, again, the subgrid models capturing more kinetic energy at long wavelengths.

Our *a posteriori* selection criterion for well performing models was based on matching the final magnetic field amplitude of the target run. Nevertheless, our subgrid models appear to show a better match to the Kolmogorov power law behavior for the kinetic energy than the LR simulation, even though no information about the kinetic energy was used to select them as well performing models. This suggests that our selection criterion focusing on magnetic field amplification is robust in choosing models that capture all aspects of subgrid physics, not just magnetic field dynamics.

We finally assess the performance of our models by analyzing the growth rate of the KHI. We have noted that our subgrid models grow their magnetic fields earlier than the fiducial runs with no subgrid model, due to the fact that we train on late time data. However, it is important that the growth rate of the instability is preserved, as this contains physically important information. In a DNS fully capturing the KHI, we expect the growth of the magnetic field energy to be exponential in time, with a growth rate related to the local behavior in the vicinity of the shear layer, $B^2 \propto e^{\gamma_{\text{KHI}} t}$. As the instability develops and a fully turbulent state is realized, it is practically difficult to analyze the width of a given shearing layer, though diagnostic proxies for this have been constructed, such as the local shear [81]. In a simulation with finite resolution that cannot directly capture the full cascade of the KHI, it has however been shown that the growth rate should be inversely proportional to the grid spacing $\gamma_{\text{KHI}} \propto \frac{1}{\Delta x}$ [82]. We model the growth of the magnetic field energy (E_B) therefore with the form

$$E_B = E_0 e^{\gamma_{\text{KHI}}(t-t_0)}, \quad (59)$$

where t_0 is the time at which the KHI begins to grow, after the initial growth associated with the initial conditions, seen uniform across all resolutions in runs without a subgrid model, has finished. We fit this functional form to the data in Fig. 13 for a range of values of t_0 to search for the optimal window during which the KHI is active for each model and extract γ_{KHI} for each simulation. We then make the assumption that $\gamma_{\text{KHI}} = \gamma/\Delta x$, where γ is a universal constant. From our three configurations without a subgrid model (LR, SR, HR) we extract an average value of $\gamma = 0.00408$. This value allows us to extract an “effective” Δx or, equivalently, an effective number of grid points $N_{\text{mesh,eff}} = 2\gamma_{\text{KHI}}/\gamma$ for each of our runs employing a subgrid model, by comparing this value of γ to the γ_{KHI} fitted for each subgrid run. This demonstrates the resolution that would be needed to obtain an equivalent growth rate of the KHI, seen instead at low

resolution with the subgrid model employed. Our results are detailed in Table II, and we see that, of the f2 models, the effective resolution largely matches the target SR resolution of 256 points, with three of our models below 12% error, with two other models performing notably worse with error at worst 38%. For the f4 models, we see slightly better performance in recovering the target growth rate with three models showing under 6% error and the remaining two showing at worst an error of 20%.

We can again compare the behavior of our simulations to the alternate gradient-model-based scheme presented in Viganò *et al.* [45]. When considering the same 3D KHI problem in special relativistic physics, the gradient model does not show the early amplification of the magnetic field demonstrated in our simulations. As mentioned above, however, it is the late time physics that we have trained our model to reproduce. As in the Newtonian case, it is found that the bare gradient model underestimates the growth rate and final amplification of the higher resolution target simulations, and an extra prefactor is required to boost the amplification to the target values. This suggests that in constructing such a subgrid model it is necessary to introduce free parameters that must be tuned to data from higher resolution simulations, validating that the machine-learning, data-driven approach we take here is well suited to the problem.

4. Robustness of models with τ_D

In the models presented above we have included the impact of all subgrid terms, including that which enters the right-hand side of the density evolution equation τ_D . In our *a posteriori* tests, we have empirically found that it is easier to find successful models in hyperparameter space that match the target magnetic field amplification by deactivating this term, and that, with this term active, there is a larger set of hyperparameters that lead to models that overamplify the magnetic field. This overamplification can drive the simulation to unphysical values and eventually lead to a sufficiently strong magnetization that the ideal MHD equations become ill conditioned, leading to the crashing of our numerical code. Nevertheless, we find a sufficiently large region of hyperparameter space containing models that give successful evolutions with this term active, to conclude that robust and accurate models and evolutions can be constructed with all subgrid terms active.

We note that these failures occur even in the presence of a regularization procedure provided by the failure response handling of our conserved-to-primitive algorithm, following [83], provided by the PRIMITIVE-SOLVER library, detailed in [72]. Should the physical state at the end of a time step leave a predefined range of predetermined values, the primitive density ρ and temperature T are set to floor values, while the velocity is set to 0. Of particular relevance to this problem, should the magnetization exceed a ceiling value (here 10^3), in a given computational cell,

extra mass is injected into this cell to bring the magnetization down to this ceiling value. The handling of the atmosphere region and regions where the conserved-to-primitive variable conversion fails in a BNS simulation can be highly sensitive, with different treatments giving a nontrivial impact on the extractable physics, as shown in [84]. It may therefore be possible to further regularize simulations with these models that show excessive amplification, with a more in-depth tuning of the failure response parameters of this algorithm.

The impact of this term in the density evolution equation is relatively rarely studied in the literature. In the case of Newtonian MHD, due to the particular choice of the Favre filter operation on the fluid velocity, this subgrid term can be removed by definition. Therefore, in the extensive literature on nonrelativistic (M)HD LES simulations, this term can be neglected. In the relativistic case, due to the nonlinear dependence of the momentum density on the fluid 3-velocity, through the presence of the Lorentz factor $W = \frac{1}{\sqrt{1-v^2}}$, this choice of filter does not allow a simple elimination of this subgrid term. Of the other authors who have focused on LES simulation in fully relativistic MHD, only in the gradient approach of [44,45] have the full spectrum of terms, including this term in the density evolution, been considered.

In their study of *a posteriori* performance (in general relativistic MHD), Viganò *et al.* [45] found that, with the addition of an extra prefactor of 8 which multiplies the overall subgrid term, tuned to data from a KHI simulation, an *a posteriori* test with all subgrid terms active reproduces the magnetic field behavior seen at double the resolution. It is shown that within this model the dominant effect on the magnetic field comes from the subgrid term in the induction equation only, with all other subgrid terms switched off. As discussed above in the Newtonian case (Sec. III A 4), we have found that, in our model, all subgrid terms are important in recovering the high resolution behavior. In full production applications, in the context of a BNS merger [78], the same group has found that magnetic field amplification is enhanced by increasing the prefactor on the subgrid term in the induction equation but is suppressed by introducing a prefactor for the terms appearing in the momentum and density equations, and that these terms can strongly influence the collapse time of a postmerger remnant to a black hole. The conclusion of their study is that to capture the behavior of the higher resolution simulations without a subgrid model, all subgrid terms other than that in the induction equation should be deactivated, with the term in the induction equation premultiplied by a factor of 8, the approach taken in subsequent works [15,85]. Here we have shown that, for the test problems considered, the high resolution behavior can be matched with all subgrid terms active, including that in the density equation, without the addition of a prefactor, though we note that our model is, similar to this prefactor,

entirely fitted to data. It is also the case that the problems we consider here are far more idealized than a full BNS simulation. In a full application of this approach to a BNS merger, it will be necessary to reexamine the relative importance of these terms and the robustness of our scheme to this problem. We may hope however, that the inclusion of terms in the momentum equation especially will allow us to accurately capture the impact of the small scale magnetic field on momentum transport in the system.

IV. CONCLUSION

In this paper we have presented the first implementation of an online subgrid model, tested *a posteriori*, for modeling turbulent flow in relativistic MHD using NNs. We have demonstrated the application of this model in two test problems of the KHI: first, a 2D problem in Newtonian MHD, and second, a fully 3D problem in SR.

In both cases we initially demonstrated that we can train relatively small NNs on high resolution data, filtered down to lower resolution, to successfully evaluate subgrid tensors, using the fluid state and its derivatives as inputs. We have shown that these models can reproduce a training dataset with good accuracy, passing the initial *a priori* test that such subgrid models have previously been subjected to in [53].

We have then gone beyond these tests to, for the first time to the best of our knowledge, show success in *a posteriori* testing. To demonstrate this we have performed low resolution simulations, evaluating the NN online in the simulation to provide the subgrid terms. We have shown that models can be constructed that successfully reproduce the high resolution total amplification of the magnetic field, Kelvin-Helmholtz growth rate, and spectral distribution of magnetic and kinetic energies in these low resolution simulations, with a speedup of a factor up to 44.

Further, we have investigated the role of individual subgrid terms in the evolution of such problems. In the Newtonian case we have seen that subgrid models that successfully give the expected magnetic field amplification in *a posteriori* tests do so due to the impact of all relevant subgrid terms, incorporating the impact on the momentum, energy, and magnetic field evolutions. This suggests that subgrid models that only include the impact on, e.g., the magnetic field evolution, may neglect important contributions to the magnetic field evolution. We have also seen in the SR case that robust models in *a posteriori* testing can be found even when the subgrid contribution to the density evolution, absent in Newtonian physics, is included, providing a model that incorporates all subgrid effects at once.

This work demonstrates the applicability of such techniques to more complicated scenarios, specifically full general relativity, in the context of isolated neutron star evolution, binary neutron star mergers, and accretion flows around a black hole. Future work will investigate applying

such techniques to these configurations and modeling the magnetic field evolution at relatively low resolution with the use of such subgrid models. Our work in this paper has focused solely on the KHI, however, other authors constructing different subgrid models have focused on constructing bespoke models for separate magnetic field instabilities that may trigger turbulent flow, specifically both the KHI [81] and the MRI and associated parasitic instabilities [86]. Future work is required on our ML-based model to understand whether bespoke NNs are required to capture each of these separate instabilities or whether a single network can be trained to reproduce all of the relevant subgrid physics at once.

The filtering operations discussed in this work break the covariance of general relativity. It has been suggested that this breaking of covariance should lead to gauge artifacts which impact the thermodynamics of the evolved system [87–89]. When considering the general relativistic extension to this problem, it may prove important to implement the above subgrid approach in a gauge invariant manner to obtain robust physical results.

ACKNOWLEDGMENTS

The authors thank Charles Board, Thomas Celora, and Prasoon Pandey for helpful conversations. S. B. acknowledges support by the EU Horizon under ERC Consolidator Grant No. InspiReM-101043372. S. B. acknowledges support by the DFG project “Magnetfelddynamik in Neutronensternen” MERLIN (MERLIN; BE 6301/6-1 Projektnummer 524726453). Simulations were performed on the national HPE Apollo Hawk (Hunter) at the High Performance Computing Center Stuttgart (HLRS). The authors acknowledge HLRS for funding this project by providing access to the supercomputer HPE Apollo Hawk (Hunter) under Grants No. INTRHYGUE/44215 and No. MAGNETIST/44288. Simulations were also performed on SuperMUC_NG at the Leibniz-Rechenzentrum (LRZ) Munich. The authors acknowledge the Gauss Centre for Supercomputing e.V. [90] for funding this project by providing computing time on the GCS Supercomputer SuperMUC-NG at LRZ (Allocations No. pn67xo and No. pn76li). Simulations and postprocessing were also performed on the ARA and DRACO clusters at Friedrich Schiller University Jena and on the Tullio INFN cluster at INFN Turin. The ARA cluster is funded in part by DFG Grants No. INST 275/334-1 FUGG and No. INST 275/363-1 FUGG and ERC Starting Grant, Grant Agreement No. BinGraSp-714626.

DATA AVAILABILITY

There are no publicly available research data or software supporting this manuscript. Requests for further information or data should be sent to the authors.

APPENDIX: HYPERPARAMETER OPTIMIZATION AND TRAINING DATASET CONSTRUCTION

In order to find performant, accurate models, we train models with a wide range of hyperparameters. We assess the performance of these models both in *a priori* and *a posteriori* tests and have presented the best performing models in the analysis above. In particular, we iterate over the number of hidden layers N_L , the number of neurons per hidden layer N_n , and the choice of nonlinear activation function. As well as the hyperparameters of the model architecture, we have also iterated over various methods of dataset construction. For the Newtonian 2D problem this entails exploring strategies for augmenting the tails of the dataset which is comparatively small due to the dimensionality of the problem, while in the 3D special relativistic case, this has involved subsampling the spatial domain included in the training set.

Our data augmentation strategy for the 2D case is driven by the fact that the region of most importance for our subgrid model is the tail of the distribution of subgrid tensor values within our training set, that is, we are most interested in accurately modeling regions where the subgrid tensor takes large values, which are rare within our training set. We therefore pursue a strategy that boosts the tails of the distribution and suppresses the mean. To do this we take our unaugmented training dataset and calculate a statistic for each value of a subgrid tensor component τ ,

$$\tau_{\text{remove}} = 1 - (1 - 10^{-6}) \exp\left(\frac{-\tau^2}{2(\sigma\sigma_\tau)^2}\right), \quad (\text{A1})$$

$$\tau_{\text{add}} = 1 - (1 - 10^{-6}) \exp\left(\frac{-\tau^2}{2(2\sigma\sigma_\tau)^2}\right), \quad (\text{A2})$$

where σ_τ is the standard deviation of the distribution of τ in the unaugmented dataset. The mean of all distributions considered is ≈ 0 . We then generate a random variable drawn from a uniform distribution $\tau_{\text{cut}} \sim \text{Unif}[0, 1]$ for each element of our dataset and compare it to these statistics. If $\tau_{\text{cut}} > \tau_{\text{remove}}$ we remove the point from our dataset, and if $\tau_{\text{cut}} < \tau_{\text{add}}$ we add N_{aug} copies of the point to the dataset; thus flattening the distribution. σ, N_{aug} are hyperparameters we search over. In Fig. 15 we show the impact of this data augmentation on the training data.

We can see that the unaugmented dataset is heavily weighted toward the mean, where the value of the subgrid tensor is small, representing nonturbulent regions within the simulation. Since we require small models for fast evaluation to retain performance, this unaugmented dataset will lead to a model well trained for nonturbulent regimes, but with poor performance in the turbulent state. These configurations can correspond to those discussed above, which perform well in *a priori* tests, but poorly in

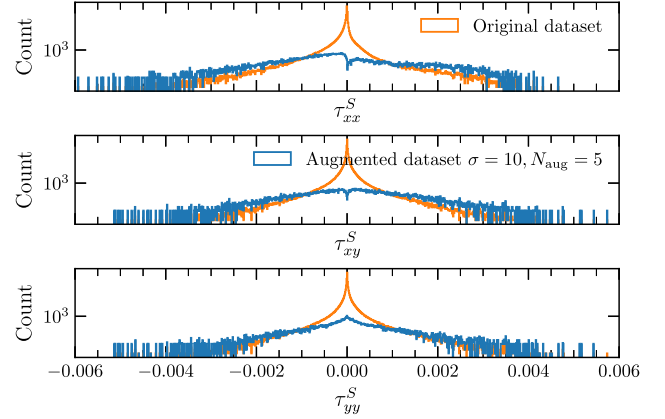


FIG. 15. Histograms of the training dataset for the components of the momentum subgrid tensor τ_{ij}^S before (orange) and after (blue) the data augmentation procedure. This dataset was used to train model N3.

a posteriori tests. In comparison, we find the flattened, augmented dataset to give models better trained in all physical regimes of our problem. This augmentation strategy is stochastic, however, we perform multiple realizations of the augmented dataset and verify that the resulting dataset does not significantly vary.

The range of hyperparameters we iterate over is given in Table III. We have selected network sizes to ensure that the evaluation time of the network online in the *a posteriori* test is not so slow that a low resolution run with the network is slower than a high resolution run without the network. In total we have explored 390 configurations for the Newtonian problem and 347 for the SR problem, run with full *a posteriori* tests. From this wide span of hyperparameter space we have shown the best five performing models in each family in *a posteriori* tests above and a small subset of the other successful models. We note that many models in the Newtonian case without data augmentation show insufficient amplification of the magnetic field, as do models trained on too early a set of time slices which do not capture the late time turbulent state sufficiently well. In Fig. 16 we show a wider subset of hyperparameters explored for the special relativistic problem, focusing on the f4 models. This is a subset of the full hyperparameter set which would be impractical to show in its entirety. Note that some simulations were terminated early if the evolution was clearly not trending toward the targeted growth of the magnetic field. We also note that some simulations not pictured show such a strong growth in the magnetic field that unphysical values are reached and the code generates “not a number” values. We see here, however, that well performing models are not a rare outcome of the training procedure, but also that a spread of outcomes can be expected in terms of growth rates and magnetic field amplifications, and that such *a posteriori* tests are necessary to select the best performing models. While the performance of an individual model can only be truly

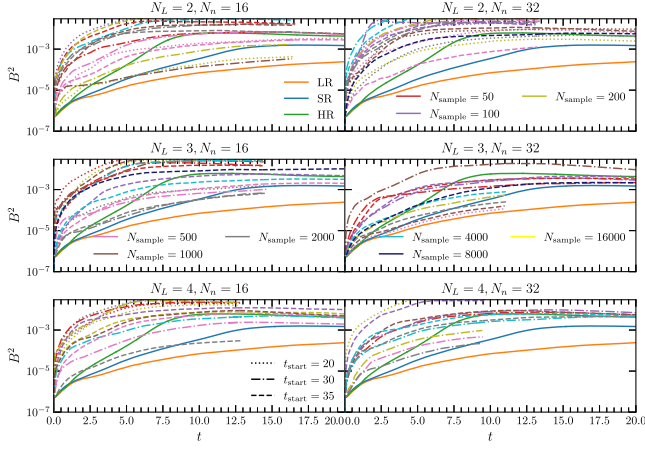


FIG. 16. Performance of a range of f4 models spanning a subset of hyperparameter space. Solid curves show the reference runs without subgrid models. Different subpanels show different choices of the number of hidden layers N_L and number of neurons N_n . Line color shows the choice of sampling parameter N_{sample} , while line style shows the choice of t_{start} .

assessed with *a posteriori* simulations, we find several broad trends in our hyperparameter search that point us toward a range of hyperparameters that generally provide well trained models. As expected, we have found that models trained on datasets that contain less of the well developed turbulent regime show a suppressed magnetic field amplification, while models trained with data that sample this phase well enhance their amplification.

TABLE III. Hyperparameter ranges. (N) denotes a parameter only relevant to the Newtonian case, (S) for the SR case.

Hyperparameter	Values
N_L	2, 3, 4, 5, 6
N_n	16, 32, 64
Act. func.	ReLU (x), tanh (x), Leaky ReLU (x)
Learning rate	$10^{-3}, 10^{-4}$
Batch size	1000
t_{start} (N)	50, 75
t_{start} (S)	20, 30, 35
σ (N)	0, 5, 10, 15, 20, 25
N_{aug} (N)	0, 5, 10, 15, 20, 25
N_{sample} (S)	50, 100, 200, 500, 1000, 2000, 4000, 8000, 16000

Therefore, we find that increasing $t_{\text{start}}, \sigma, N_{\text{aug}}$ generically provides models with a more accurate amplification. The amplification is also boosted by choosing a larger learning rate and by choosing the tanh(x) activation function over the ReLU option. For these models, however, we have sometimes found that the magnetic field amplification is strongly overestimated. In particular, the larger learning rate can lead to the model being trained too aggressively and, consequently, show lower accuracy in the amplification. As can be seen from the models we present, by investigating this high-dimensional hyperparameter space, it is possible to find models with higher learning rates or with the tanh activation function that still perform well, though generically these are rarer.

[1] D. Radice and I. Hawke, *Liv. Rev. Comput. Astrophys.* **10**, 1 (2024).
[2] D. Price and S. Rosswog, *Science* **312**, 719 (2006).
[3] S. A. Balbus and J. F. Hawley, *Astrophys. J.* **376**, 214 (1991).
[4] T. Piran, *Rev. Mod. Phys.* **76**, 1143 (2004).
[5] S. I. Blinnikov, I. D. Novikov, T. V. Perevodchikova, and A. G. Polnarev, *Sov. Astron. Lett.* **10**, 177 (1984).
[6] B. Paczynski, *Astrophys. J. Lett.* **308**, L43 (1986).
[7] J. Goodman, *Astrophys. J. Lett.* **308**, L47 (1986).
[8] D. Eichler, M. Livio, T. Piran, and D. N. Schramm, *Nature (London)* **340**, 126 (1989).
[9] R. Narayan, B. Paczynski, and T. Piran, *Astrophys. J.* **395**, L83 (1992).
[10] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), *Phys. Rev. Lett.* **119**, 161101 (2017).
[11] B. P. Abbott *et al.* (LIGO Scientific, Virgo, Fermi GBM, INTEGRAL, IceCube, AstroSat Cadmium Zinc Telluride Imager Team, IPN, Insight-Hxmt, ANTARES, Swift, AGILE Team, 1M2H Team, Dark Energy Camera GW-EM, DES, DLT40, GRAWITA, Fermi-LAT, ATCA,

ASKAP, Las Cumbres Observatory Group, OzGrav, DWF (Deeper Wider Faster Program), AST3, CAASTRO, VINROUGE, MASTER, J-GEM, GROWTH, JAGWAR, CaltechNRAO, TTU-NRAO, NuSTAR, Pan-STARRS, MAXI Team, TZAC Consortium, KU, Nordic Optical Telescope, ePESSTO, GROND, Texas Tech University, SALT Group, TOROS, BOOTES, MWA, CALET, IKI-GW Follow-up, H.E.S.S., LOFAR, LWA, HAWC, Pierre Auger, ALMA, Euro VLBI Team, Pi of Sky, Chandra Team at McGill University, DFN, ATLAS Telescopes, High Time Resolution Universe Survey, RIMAS, RATIR, SKA South Africa/MeerKAT Collaborations), *Astrophys. J. Lett.* **848**, L12 (2017).
[12] B. P. Abbott *et al.* (LIGO Scientific, Virgo, Fermi-GBM, INTEGRAL Collaborations), *Astrophys. J. Lett.* **848**, L13 (2017).
[13] A. Goldstein *et al.*, *Astrophys. J.* **848**, L14 (2017).
[14] V. Savchenko *et al.*, *Astrophys. J.* **848**, L15 (2017).
[15] C. Palenzuela, R. Aguilera-Miret, F. Carrasco, R. Ciolfi, J. V. Kalinani, W. Kastaun, B. Miñano, and D. Viganò, *Phys. Rev. D* **106**, 023013 (2022).

- [16] K. Hayashi, K. Kiuchi, K. Kyutoku, Y. Sekiguchi, and M. Shibata, *Phys. Rev. Lett.* **134**, 211407 (2025).
- [17] J. Bamber, A. Tsokaros, M. Ruiz, and S. L. Shapiro, *Phys. Rev. D* **111**, 044038 (2025).
- [18] C. Musolino, L. Rezzolla, and E. R. Most, *Astrophys. J. Lett.* **984**, L61 (2025).
- [19] E. R. Most, J. Peterson, L. Scurto, H. Pais, and V. Dexheimer, *Astrophys. J. Lett.* **989**, L29 (2025).
- [20] E. M. Gutiérrez, W. Cook, D. Radice, S. Bernuzzi, J. Fields, P. Hammond, B. Daszuta, H. Bandyopadhyay, and M. Jacobi, *arXiv:2506.18995*.
- [21] W. Cook, E. M. Gutiérrez, S. Bernuzzi, D. Radice, B. Daszuta, J. Fields, P. Hammond, H. Bandyopadhyay, and M. Jacobi, *arXiv:2508.19342*.
- [22] I. Rainho, J. Bamber, D. Guerra, M. Miravet-Tenés, M. Ruiz, A. Tsokaros, and S. L. Shapiro, *arXiv:2510.17511*.
- [23] K. Kiuchi, A. Reboul-Salze, Y. Sekiguchi, and M. Shibata, *arXiv:2606.11299*.
- [24] A. Wen, J. V. Kalinani, M. Chabanov, M. Campanelli, R. Cioffi, and Y. Zlochower, *arXiv:2605.30548*.
- [25] A. Tsokaros, M. Ruiz, S. L. Shapiro, and K. Uryū, *Phys. Rev. Lett.* **128**, 061101 (2022).
- [26] A. Sur, W. Cook, D. Radice, B. Haskell, and S. Bernuzzi, *Mon. Not. R. Astron. Soc.* **511**, 3983 (2022).
- [27] W. Cook, R. K. Joshi, S. Bernuzzi, B. Haskell, and J. Fields, *Mon. Not. R. Astron. Soc.* **547**, stag136 (2026).
- [28] F. V. Piñas, A. K. L. Yip, P. C.-K. Cheong, and M. Ruiz, *Astrophys. J.* **1001**, 49 (2026).
- [29] A. Capobianco, W. Cook, S. Bernuzzi, B. Haskell, and J. Fields, *arXiv:2605.22921*.
- [30] E. M. Gutiérrez, D. Radice, J. Fields, and J. M. Stone, *arXiv:2601.20953*.
- [31] J. Smagorinsky, *Mon. Weather Rev.* **91**, 99 (1963).
- [32] M. Germano, U. Piomelli, P. Moin, and W. H. Cabot, in *Studying Turbulence Using Numerical Simulation Databases. 3: Proceedings of the 1990 Summer Program*, edited by D. Spinks (Center for Turbulence Research, Stanford University, 1990), pp. 5–17.
- [33] W. K. Yeo, A generalized high pass/low pass averaging procedure for deriving and solving turbulent flow equations, Ph.D. thesis, Ohio State University, 1987, <http://rave.ohiolink.edu/etdc/view?acc-num=osu1210098135>.
- [34] R. A. Clark, J. H. Ferziger, and W. C. Reynolds, *J. Fluid Mech.* **91**, 1 (1979).
- [35] M. L. Theobald, P. A. Fox, and S. Sofia, *Phys. Plasmas* **1**, 3016 (1994).
- [36] W.-C. Müller and D. Carati, *Phys. Plasmas* **9**, 824 (2002).
- [37] P. Grete, Large Eddy simulations of compressible magnetohydrodynamic turbulence, Ph.D. thesis, Georg August University of Göttingen, Germany, 2016.
- [38] P. Grete, D. G. Vlaykov, W. Schmidt, and D. R. G. Schleicher, *Phys. Plasmas* **23**, 062317 (2016).
- [39] D. G. Vlaykov, P. Grete, W. Schmidt, and D. R. G. Schleicher, *Phys. Plasmas* **23**, 062316 (2016).
- [40] M. Miesch, W. Matthaeus, A. Brandenburg, A. Petrosyan, A. Pouquet, C. Cambon, F. Jenko, D. Uzdensky, J. Stone, S. Tobias, J. Toomre, and M. Velli, *Space Sci. Rev.* **194**, 97 (2015).
- [41] W. Schmidt, *Liv. Rev. Comput. Astrophys.* **1**, 2 (2015).
- [42] D. Radice, *Astrophys. J.* **838**, L2 (2017).
- [43] D. Viganò, R. Aguilera-Miret, and C. Palenzuela, *Phys. Fluids* **31**, 105102 (2019).
- [44] F. Carrasco, D. Viganò, and C. Palenzuela, *Phys. Rev. D* **101**, 063003 (2020).
- [45] D. Viganò, R. Aguilera-Miret, F. Carrasco, B. Miñano, and C. Palenzuela, *Phys. Rev. D* **101**, 123019 (2020).
- [46] M. Shibata, K. Kiuchi, and Y. I. Sekiguchi, *Phys. Rev. D* **95**, 083005 (2017).
- [47] A. Pandya and F. Pretorius, *Phys. Rev. D* **104**, 023015 (2021).
- [48] Harry L. H. Shum, F. Abalos, Y. Bea, M. Bezares, P. Figueras, and C. Palenzuela, *Phys. Rev. D* **113**, 084029 (2026).
- [49] E. R. Most, *Phys. Rev. D* **108**, 123012 (2023).
- [50] J. Ling, A. Kurzwski, and J. Templeton, *J. Fluid Mech.* **807**, 155 (2016).
- [51] A. Beck, D. Flad, and C.-D. Munz, *J. Comput. Phys.* **398**, 108910 (2019).
- [52] Z. Wang, K. Luo, D. Li, J. Tan, and J. Fan, *Phys. Fluids* **30**, 125101 (2018).
- [53] S. G. Rosofsky and E. A. Huerta, *Phys. Rev. D* **101**, 084024 (2020).
- [54] E. A. Huerta, *arXiv:2605.19057*.
- [55] D. George and E. Huerta, *Phys. Lett. B* **778**, 64 (2018).
- [56] D. George and E. Huerta, in *NIPS Summer School 2017* (2017), *arXiv:1711.07966*.
- [57] Alvin J. K. Chua, C. R. Galley, and M. Vallisneri, *Phys. Rev. Lett.* **122**, 211101 (2019).
- [58] S. G. Rosofsky, H. A. Majed, and E. A. Huerta, *Mach. Learn. Sci. Tech.* **4**, 025022 (2023).
- [59] S. G. Rosofsky and E. A. Huerta, *Mach. Learn. Sci. Tech.* **4**, 035002 (2023).
- [60] S. Kacmaz, E. A. Huerta, and R. Haas, *Mach. Learn. Sci. Tech.* **6**, 035057 (2025).
- [61] T. Dieselhorst, W. Cook, S. Bernuzzi, and D. Radice, *Symmetry* **13**, 2157 (2021).
- [62] S. Kacmaz, R. Haas, and E. A. Huerta, *Symmetry* **17**, 1409 (2025).
- [63] R. Mudimadugula, F. Schianchi, A. Neuweiler, T. Wouters, H. Gieg, and T. Dietrich, *Eur. Phys. J. A* **61**, 193 (2025).
- [64] N. Bhojwani, C. Wang, H.-Y. Wang, C. Sun, E. R. Most, and A. Anandkumar, in *38th conference on Neural Information Processing Systems* (2025), *arXiv:2512.01576*.
- [65] J. M. Stone, K. Tomida, C. J. White, and K. G. Felker, *Astrophys. J. Suppl. Ser.* **249**, 4 (2020).
- [66] A. Leonard, *Adv. Geophys.* **18**, 237 (1975).
- [67] S. B. Pope, *Turbulent Flows* (Cambridge University Press, Cambridge, England, 2000).
- [68] P. Sagaut, *Large Eddy Simulation for Incompressible Flows* (Springer, Berlin, Heidelberg, 2006).
- [69] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, *arXiv:1912.01703*.
- [70] G. Guennebaud, B. Jacob *et al.*, EIGEN, <https://libeigen.gitlab.io> (2010).
- [71] B. Daszuta, F. Zappa, W. Cook, D. Radice, S. Bernuzzi, and V. Morozova, *Astrophys. J. Suppl. Ser.* **257**, 25 (2021).

- [72] W. Cook, B. Daszuta, J. Fields, P. Hammond, S. Albanesi, F. Zappa, S. Bernuzzi, and D. Radice, *Astrophys. J. Suppl. Ser.* **277**, 3 (2025).
- [73] B. Daszuta and W. Cook, in *New Frontiers in GRMHD Simulations* (Springer, Singapore, 2025).
- [74] B. Daszuta, W. Cook, P. Hammond, J. Fields, E. M. Gutiérrez, S. Bernuzzi, and D. Radice, *Phys. Rev. D* **112**, 103006 (2025).
- [75] B. Daszuta, S. Bernuzzi, M. Jacobi, E. M. Gutiérrez, P. Hammond, W. Cook, and D. Radice, [arXiv:2602.18290](https://arxiv.org/abs/2602.18290).
- [76] C. R. Evans and J. F. Hawley, *Astrophys. J.* **332**, 659 (1988).
- [77] T. A. Gardiner and J. M. Stone, *J. Comput. Phys.* **227**, 4123 (2008).
- [78] R. Aguilera-Miret, D. Viganò, F. Carrasco, B. Miñano, and C. Palenzuela, *Phys. Rev. D* **102**, 103006 (2020).
- [79] A. Kolmogorov, *Akad. Nauk SSSR Dokl.* **30**, 301 (1941).
- [80] A. P. Kazantsev, *Sov. J. Exp. Theor. Phys.* **26**, 1031 (1968).
- [81] M. Miravet-Tenés, P. Cerdá-Durán, M. Obergaulinger, and J. A. Font, *Mon. Not. R. Astron. Soc.* **527**, 1081 (2023).
- [82] K. Kiuchi, P. Cerdá-Durán, K. Kyutoku, Y. Sekiguchi, and M. Shibata, *Phys. Rev. D* **92**, 124034 (2015).
- [83] W. Kastaun, J. V. Kalinani, and R. Cioffi, *Phys. Rev. D* **103**, 023018 (2021).
- [84] B. Daszuta, S. Bernuzzi, J. Fontbuté, R. Zhai, A. T.-L. Lam, J. Fields, and D. Radice, [arXiv:2605.30430](https://arxiv.org/abs/2605.30430).
- [85] R. Aguilera-Miret, D. Viganò, and C. Palenzuela, *Astrophys. J. Lett.* **926**, L31 (2022).
- [86] M. Miravet-Tenés, P. Cerdá-Durán, M. Obergaulinger, and J. A. Font, *Mon. Not. R. Astron. Soc.* **517**, 3505 (2022).
- [87] T. Celora, N. Andersson, I. Hawke, and G. L. Comer, *Phys. Rev. D* **104**, 084090 (2021).
- [88] T. Celora, M. J. Hatton, I. Hawke, and N. Andersson, *Phys. Rev. D* **110**, 123040 (2024).
- [89] T. Celora, N. Andersson, I. Hawke, G. L. Comer, and M. J. Hatton, *Phys. Rev. D* **110**, 123039 (2024).
- [90] <https://www.gauss-centre.eu>.