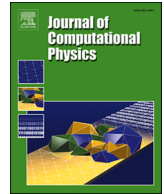




Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Neural ensemble Kalman filter: Data assimilation for compressible flows with shocks

Xu-Hui Zhou^{a,b,*}, Lorenzo Beronilla^a, Michael K. Sleeman^b, Hangchuan Hu^c,
Matthias Morzfeld^a, Andrew M. Stuart^b, Tamer A. Zaki^c

^a Scripps Institution of Oceanography, University of California San Diego, La Jolla, CA, 92093, USA

^b Computing + Mathematical Sciences, California Institute of Technology, Pasadena, CA, 91125, USA

^c Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD, 21218, USA

ARTICLE INFO

Keywords:

Data assimilation
Ensemble Kalman filter
Shocks
Discontinuities
Deep neural networks

ABSTRACT

Data assimilation (DA) for compressible flows with shocks is challenging because many classical DA methods generate spurious oscillations and nonphysical features near uncertain shocks. We focus here on the ensemble Kalman filter (EnKF). We show that the poor performance of the EnKF may be attributed to the bimodal forecast distribution that can arise in the vicinity of an uncertain shock location; this violates the assumptions underpinning the EnKF, which assume a forecast which is close to Gaussian. To address this issue we introduce the new *neural EnKF*. The basic idea is to systematically embed neural function approximations within ensemble DA by mapping the forecast ensemble of shocked flows to the parameter space (weights and biases) of a deep neural network (NN) and to subsequently perform DA in that space. The nonlinear mapping encodes sharp and smooth flow features in an ensemble of NN parameters. Neural EnKF updates are therefore well-behaved only if the NN parameters vary smoothly within the neural representation of the forecast ensemble. We show that such a smooth variation of network parameters can be enforced via physics-informed transfer learning, and demonstrate that in so-doing the neural EnKF avoids the spurious oscillations and nonphysical features that plague the EnKF. The applicability of the neural EnKF is demonstrated through a series of systematic numerical experiments with the inviscid Burgers' equation, the Sod shock tube, and a two-dimensional blast wave.

1. Introduction

Compressible flows featuring shocks and other discontinuities arise in a wide range of natural and engineered systems. In rotating detonation engines, a continuously propagating detonation wave circles the annular combustor, producing strong shocks essential for pressure-gain combustion [1]; in high-speed flight, shock waves form at the leading edges of flight vehicles and at sharp compression corners, strongly influencing the resulting pressure and thermal loads [2,3]; and in shockwave and laser lithotripsy, a focused acoustic or laser-induced pulse initiates cavitation bubble collapse near kidney stones, producing localized shock emissions that contribute to stone fragmentation [4,5]. These examples, among many other applications [6–8], highlight the significant role of shocks in modern science and engineering and underscore the importance of accurately simulating compressible flows with shocks. When the flow conditions (e.g., initial and boundary conditions) are well characterized, numerical solvers may produce high-fidelity predictions.

* Corresponding author.

E-mail address: xuz067@ucsd.edu (X.-H. Zhou).

<https://doi.org/10.1016/j.jcp.2026.115136>

Received 27 February 2026; Received in revised form 10 June 2026; Accepted 10 June 2026

Available online 17 June 2026

0021-9991/© 2026 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

In practice, however, significant uncertainties often arise from incomplete or imperfectly specified conditions, motivating the use of data-driven methods in conjunction with numerical simulations to reduce uncertainty and improve predictive capability.

Data assimilation (DA) provides a systematic Bayesian framework for reducing these uncertainties [9–11]. In many DA formulations, the system state is sequentially updated as new observations become available: with each observation, a prior estimate predicted by a numerical model is merged with observation data to form a posterior state that more faithfully represents both the governing physics and the observations. DA was initially developed within the field of numerical weather prediction [12], where atmospheric observations are used to keep atmospheric models on track, and it has since become widely adopted throughout the geosciences [13]. Building on this success, DA has seen growing application in computational fluid dynamics and related engineering problems to improve modeling fidelity and enhance predictive accuracy under uncertainty (e.g., [14–19]). However, most existing studies have focused on incompressible flows with smooth features. Effective DA for compressible *shocked* flows is an emerging sub-field [20–23].

Existing work on DA for compressible shocked flows can be broadly categorized into variational (smoothing) and sequential (filtering) methods. Within the variational paradigm, West et al. [24] applied a 4D (3D space and time) variational framework (4D-Var) to the Sod shock tube problem, estimating initial conditions that yield a solution trajectory best matching the observations over a given time interval. In contrast to identifying an optimal initial condition, sequential filtering approaches update the flow state dynamically as observations become available. Houba et al. [25] presented a systematic investigation of sequential filtering approaches for compressible flows with shocks, evaluating extended Kalman filter (EKF), ensemble Kalman filter (EnKF), and particle filter (PF) on the Sod shock tube problem. Their results showed that, due to uncertainty in shock locations, both the EKF and EnKF perform poorly in shock-dominated regimes, with the EnKF exhibiting pronounced oscillations and violations of thermodynamic realizability. In contrast, PF provided the most robust performance, provided that a sufficiently rich and appropriately structured initial ensemble is available. In subsequent work, Edoh et al. [26] introduced a positivity-preserving logarithmic variable transformation to enforce the thermodynamic realizability in the EnKF update; however, spurious oscillations near shocks were not eliminated. Edoh et al. [27] further demonstrated that these oscillatory artifacts are significantly reduced when the initial discontinuity locations are assumed to be accurately known without uncertainty.

To further improve the EnKF for shocked flows, Hansen et al. [28] applied a normal-score EnKF, originally proposed by Zhou et al. [29], to the Sod shock tube problem. The method applies a rank-based transformation to map the distribution at each spatial location toward a Gaussian distribution before the EnKF update. While this improved the performance compared to the EnKF, spurious oscillations near shocks remained because Gaussian marginal distributions do not necessarily imply a Gaussian joint distribution. They further investigated DA in the context of detonation wave dynamics and proposed a two-step local ensemble transform Kalman filter (LETKF) framework that aligns ensemble members based on the detonation front prior to the analysis step [30]. In parallel, Li et al. [31,32] proposed a structurally informed ETKF method that incorporates weighting matrices derived from state gradients into the prior covariance, thereby explicitly encoding local discontinuity information in the analysis step. This method was demonstrated for several nonlinear hyperbolic problems, including the dam-break problem, the shallow water equations, and the 2D Burgers' equation, under relatively dense observation settings. However, its performance for compressible shocked flows under sparse observations has not yet been investigated. Beyond ensemble Kalman filtering approaches, Subrahmanya and Sandu [33] introduced a feature-preserving DA method based on the ensemble transform particle filter (ETPF) to address the fundamental issue of feature smearing during DA. The method formulates the analysis update as an optimal transport problem to determine the ensemble transform weights. These weights are subsequently decomposed into a sequence of pairwise convex combinations of particles, while dynamic time warping is applied to align particles pairwise in order to preserve sharp flow features in the posterior states. The method showed promising results on test cases including shock tube problems and a 2D blast wave case. Nevertheless, it relies on the presence of consistent features across particles, and can suffer from particle collapse unless inflated observation errors are used. In a different line of PF-based approaches, Srivastava et al. [34] proposed a feature-informed DA method that directly assimilates solution features, such as shock locations, to infer a small set of governing parameters instead of performing full-state estimation. However, a small set of parameters is often insufficient to capture the dynamics of practical shock-laden flows.

In this work, we analyze the challenges associated with applying the stochastic EnKF to compressible flows with shocks and propose a simple yet effective approach. We show that the degradation of the EnKF near shocks can be traced to a fundamental mechanism: uncertainty in the shock locations induces strongly bimodal forecast distributions in physical space, violating the assumptions underlying the EnKF and leading to spurious oscillations and nonphysical updates. This mechanism is illustrated using a hyperbolic-tangent surrogate shock as a diagnostic example. Motivated by this insight, we propose performing EnKF updates in an alternative representation and develop the *neural EnKF*, which carries out the analysis step in the neural network parameter space (hereafter referred to as the *neural space*). In this approach, each ensemble member is represented by a deep neural network, with a shared architecture across all members, whose weights (and biases) implicitly encode the underlying flow features. The networks are trained progressively along a nearest-neighbor chain constructed from physical-space similarity to maintain well-behaved ensemble statistics in the neural space. Observations in physical space, such as pressure measurements, are assimilated by updating the network weights, yielding an analysis ensemble of networks whose reconstructed flow fields better reflect the data while preserving sharp flow features. We assess the DA performance of the neural EnKF through a series of systematic numerical experiments, including the inviscid Burgers' equation, the Sod shock tube problem, and a 2D blast wave case. These test cases demonstrate the neural EnKF's ability to perform robust DA in shock-laden flows.

The remainder of this paper is organized as follows. Section 2 examines the behavior of the EnKF in the presence of uncertain shocks, using a hyperbolic-tangent example to illustrate how sharp transitions challenge the EnKF update and to clarify the underlying failure mechanism. Section 3 presents the neural EnKF methodology, including the neural-network representation of ensemble members, the EnKF update in neural space, and a nearest-neighbor chain training strategy designed to maintain well-behaved

ensemble statistics in the neural space. Section 4 evaluates the proposed method through a series of systematic numerical experiments in shock-laden flows. Finally, Section 5 summarizes the main findings and discusses directions for future research.

2. Motivation

We begin by briefly reviewing the stochastic EnKF [35], which is a natural choice for the problems considered here, because it is derivative free and because it scales to high-dimensional systems; these two features have led to its widespread adoption in many practical applications including weather prediction, climate science, physical oceanography and reservoir engineering. Despite these advantages, the EnKF exhibits notable difficulties when applied to compressible flows with shocks. To study this behavior, we use a hyperbolic-tangent surrogate shock to examine how sharp transitions challenge the EnKF update and lead to spurious oscillations.

2.1. Ensemble Kalman filter (EnKF)

The EnKF updates a forecast ensemble, predicted by a numerical model, using observations. At each DA cycle, forward simulations generate a forecast ensemble $\{\mathbf{z}_i^f\}_{i=1}^{n_e}$, where $\mathbf{z}_i^f \in \mathbb{R}^{n_z}$ denotes the forecast state of the i th ensemble member and n_e is the ensemble size. The ensemble is characterized by its mean and normalized ensemble perturbations:

$$\bar{\mathbf{z}}^f = \frac{1}{n_e} \sum_{i=1}^{n_e} \mathbf{z}_i^f, \quad \mathbf{Z} = \frac{1}{\sqrt{n_e - 1}} \begin{bmatrix} \mathbf{z}_1^f - \bar{\mathbf{z}}^f, \dots, \mathbf{z}_{n_e}^f - \bar{\mathbf{z}}^f \end{bmatrix} \in \mathbb{R}^{n_z \times n_e}. \quad (1)$$

Predicted observations are defined as $\mathbf{y}_i = \mathcal{H}(\mathbf{z}_i^f)$, where \mathcal{H} is the observation operator that maps the state space to the observation space. The ensemble of predicted observations is characterized by its mean and normalized ensemble perturbations:

$$\bar{\mathbf{y}} = \frac{1}{n_e} \sum_{i=1}^{n_e} \mathbf{y}_i, \quad \mathbf{Y} = \frac{1}{\sqrt{n_e - 1}} \begin{bmatrix} \mathbf{y}_1 - \bar{\mathbf{y}}, \dots, \mathbf{y}_{n_e} - \bar{\mathbf{y}} \end{bmatrix} \in \mathbb{R}^{n_y \times n_e}. \quad (2)$$

The EnKF combines the forecast ensemble, the predicted observations, and the observations \mathbf{d} to obtain the analysis ensemble members:

$$\mathbf{z}_i^a = \mathbf{z}_i^f + \mathbf{Z}\mathbf{Y}^\top (\mathbf{Y}\mathbf{Y}^\top + \mathbf{R})^{-1} (\mathbf{d} + \boldsymbol{\eta}_i - \mathbf{y}_i), \quad \boldsymbol{\eta}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \quad (3)$$

where $\mathbf{R} \in \mathbb{R}^{n_y \times n_y}$ is the observation error covariance (a symmetric positive semidefinite matrix), and $\boldsymbol{\eta}_i$ represents the independent observation perturbations. We focus here on the stochastic EnKF as a representative EnKF formulation, although the ideas developed in this work can be naturally extended to other variants.

The EnKF update shows that each analysis member is obtained as a linear combination of the forecast ensemble. As a result, the analysis ensemble remains within the subspace spanned by the forecast ensemble, a feature commonly referred to as the *subspace property* of the EnKF [11,36]. This subspace property can become restrictive when reconstructing functions with sharp gradients using (approximately) discontinuous basis functions. We illustrate this effect through a simple example in the following subsection.

2.2. Ensemble Kalman filtering in the presence of a ‘‘shock’’

We investigate the behavior of the EnKF in the presence of a sharp, shock-like transition using a diagnostic example based on a hyperbolic-tangent profile,

$$h(x) = \frac{h_L + h_R}{2} - \frac{h_L - h_R}{2} \tanh(\kappa(x - x_s)), \quad (4)$$

where h_L and h_R denote the left and right plateau values, $\kappa > 0$ controls the steepness of the transition, and x_s specifies the nominal shock location. This smooth yet rapidly varying profile provides a controlled approximation of a 1D shock.

The true state \mathbf{z}^* is defined by the parameter set $(h_L, h_R, \kappa, x_s) = (1.0, 0.2, 25, 0)$ and evaluated on a uniform grid over $x \in [-1, 1]$ with spacing $\Delta x = 0.01$. To represent uncertainty, a 50-member forecast ensemble is generated by independently sampling the parameters (h_L, h_R, κ, x_s) from (truncated) Gaussian distributions:

$$h_L \sim \mathcal{N}(1.02, 0.1^2), \quad h_R \sim \mathcal{N}(0.22, 0.03^2), \quad \kappa \sim \mathcal{N}_{(0,\infty)}(25, 4.0^2), \quad x_s \sim \mathcal{N}_{(-1,1)}(0, 0.1^2). \quad (5)$$

Here, $\mathcal{N}_{(a,b)}(\mu, \sigma^2)$ denotes a Gaussian distribution with mean μ and variance σ^2 truncated to the interval (a, b) , with the truncation enforced through rejection sampling. Each sampled parameter set defines a forecast ensemble member, forming the forecast ensemble shown in Fig. 1(a).

Synthetic observations are constructed by sampling the true state \mathbf{z}^* at 10 uniformly spaced locations with spacing $\Delta x_{\text{obs}} = 0.2$, located at $x = \pm 0.1, \pm 0.3, \dots, \pm 0.9$, and adding independent Gaussian noise,

$$\mathbf{d} = \mathbf{H}\mathbf{z}^* + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, 0.03^2 \mathbf{I}), \quad (6)$$

where \mathbf{H} denotes a linear observation operator that extracts the state values at the observation locations. Consistent with this observation model, the predicted observations for each forecast ensemble member \mathbf{z}_i^f are given by

$$\mathbf{y}_i = \mathbf{H}\mathbf{z}_i^f. \quad (7)$$

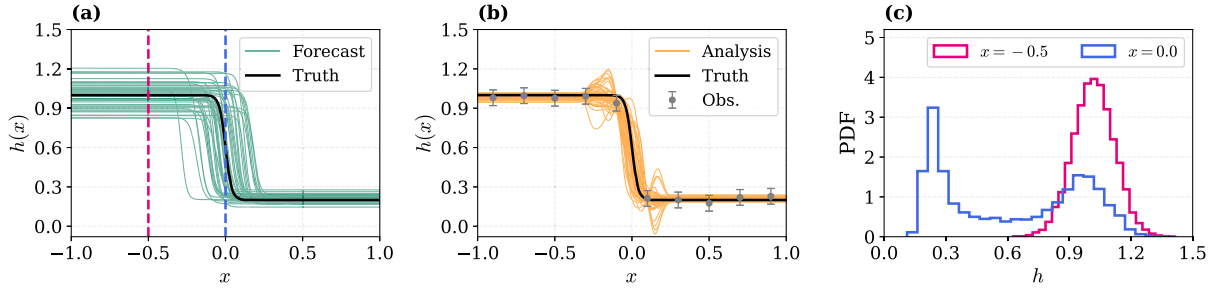


Fig. 1. Behavior of the EnKF for a shock-like transition, illustrated using a hyperbolic-tangent example: (a) forecast ensemble with 50 members; (b) analysis ensemble obtained using the EnKF, showing spurious oscillations near the shock; and (c) marginal probability density functions (PDFs) of the forecast ensemble at two representative locations indicated in (a), showing an approximately Gaussian distribution away from the shock ($x = -0.5$) and a bimodal distribution near the shock location ($x = 0$). For clarity, the PDFs in (c) are estimated using 10,000 forecast ensemble members. Here, the error bars represent ± 2 standard deviations of the observational noise; the same convention is adopted in all subsequent figures containing error bars.

Given the forecast ensemble $\{z_i^f\}_{i=1}^{50}$, the predicted observations $\{y_i\}_{i=1}^{50}$, and the synthetic data \mathbf{d} , the EnKF update is performed according to Eq. (3).

The resulting analysis ensemble exhibits pronounced oscillations in the vicinity of the sharp transition, as illustrated in Fig. 1(b). This behavior reflects a fundamental limitation of the EnKF: it performs well when the forecast state distribution is Gaussian or nearly Gaussian, but becomes ineffective in strongly non-Gaussian scenarios [37]. In this example, uncertainty in the shock location induces pronounced non-Gaussianity. Variations in the shock position cause ensemble members to place the steep transition at different spatial points, such that some members have already transitioned to the right plateau while others remain on the left at a given spatial point. This effect is further illustrated in Fig. 1(c), which shows the marginal distributions of the forecast ensemble at two representative locations: one in a smooth region far from the shock ($x = -0.5$) and one near the shock ($x = 0$). At $x = -0.5$, the distribution is approximately Gaussian. In contrast, at $x = 0$, the distribution is clearly bimodal, with one mode corresponding to ensemble members whose shock lies to the left of the point and another corresponding to those whose shock lies to the right. This bimodality explains why the EnKF produces spurious oscillations when applied near discontinuities. Such spurious oscillations are undesirable in compressible shock-laden flows, as they severely distort the physical structure of the solution.

3. Methodology

We introduce a new variant of an EnKF, the neural EnKF, to address problems caused by bimodal forecast distributions common in shock-laden flows (see above). The basic idea is to embed neural function approximations in the EnKF by mapping the forecast ensemble to the parameter space (weights and biases) of a deep neural network (NN) and to subsequently perform DA in that space. The nonlinear mapping encodes sharp and smooth flow features in a transformed ensemble of NN parameters. A key challenge here is associated with the NN parameterization of the forecast ensemble. This challenge motivates a transfer learning strategy that enforces smooth variation in the neural space through a nearest-neighbor chain.

3.1. Neural EnKF

The neural EnKF consists of three main steps, as illustrated in Fig. 2: (i) neural-network parameterization of the forecast ensemble members, (ii) the EnKF update in the neural space, and (iii) reconstruction of the analysis ensemble in physical space.

Specifically, at each DA cycle, forward simulations generate a forecast ensemble $\{z_i^f\}_{i=1}^{n_e}$ on the spatial grid \mathbf{x} . Each forecast state z_i^f is then mapped into a neural representation by training a deep neural network $F_{NN}(\theta_i^f)$, where θ_i^f denotes the network parameters. In this work, each ensemble member is represented by a single fully connected feedforward network, which predicts all flow variables simultaneously from the spatial coordinates. In the final layer, softplus activations are applied to the thermodynamic outputs (e.g., density and pressure) to ensure positivity and improve realizability of the reconstructed physical states. Each network is trained in a standard supervised learning manner to represent the corresponding forecast state by minimizing a mean-squared reconstruction loss. This procedure yields a forecast ensemble of neural networks with identical architectures but distinct network parameters θ_i^f .

With the neural representation established, the EnKF update is performed in the neural space, where the forecast ensemble is characterized by the mean and normalized perturbations of the network parameters:

$$\bar{\theta}^f = \frac{1}{n_e} \sum_{i=1}^{n_e} \theta_i^f, \quad \Theta = \frac{1}{\sqrt{n_e - 1}} \left[\theta_1^f - \bar{\theta}^f, \dots, \theta_{n_e}^f - \bar{\theta}^f \right] \in \mathbb{R}^{n_\theta \times n_e}. \quad (8)$$

The corresponding predicted observations are computed as $y_i = H(z_i^f)$. Since the neural networks in this work are trained to approximate the physical states with fitting errors that are negligible relative to the prescribed observation error, applying the observation operator to the physical states or to the neural network reconstructions yields essentially identical results. The resulting ensemble of

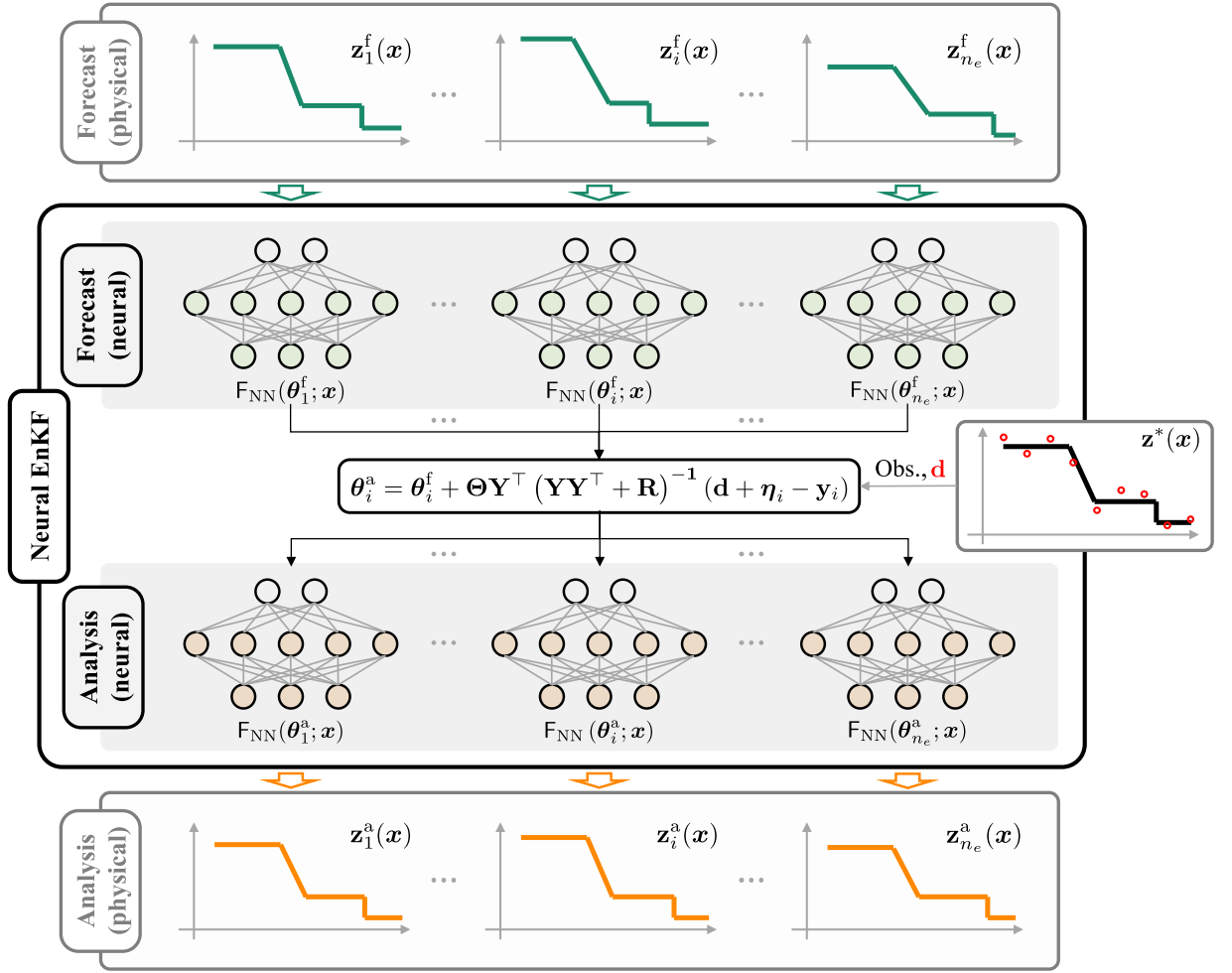


Fig. 2. Schematic of the neural EnKF framework. At each DA cycle, forward simulations generate a forecast ensemble $\{z_i^f(\mathbf{x})\}_{i=1}^{n_e}$ in physical space. Each forecast ensemble member is then parameterized by a deep neural network $F_{\text{NN}}(\theta_i^f; \mathbf{x})$, which maps the spatial coordinates to the corresponding physical state (e.g., density, velocity, and pressure in compressible flows). The network parameters θ_i^f are then updated to θ_i^a through assimilation of the physical-space observations \mathbf{d} (e.g., pressure measurements). Finally, the updated networks reconstruct the analysis ensemble $\{z_i^a(\mathbf{x})\}_{i=1}^{n_e}$ back in physical space.

predicted observations is characterized by its mean and normalized ensemble perturbations as in Eq. (2). The EnKF update is then applied to the network parameters,

$$\theta_i^a = \theta_i^f + \Theta \mathbf{Y}^T (\mathbf{Y} \mathbf{Y}^T + \mathbf{R})^{-1} (\mathbf{d} + \eta_i - y_i), \quad \eta_i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \quad (9)$$

yielding an analysis ensemble of network parameters $\{\theta_i^a\}_{i=1}^{n_e}$. The corresponding analysis states in physical space are then obtained through reconstruction,

$$z_i^a(\mathbf{x}) = F_{\text{NN}}(\theta_i^a; \mathbf{x}). \quad (10)$$

One can view the mapping from forecasted physical states to neural-network parameters as a *nonlinear* coordinate transformation applied prior to the EnKF analysis step. The nonlinearity of this transformation is important, because the affine invariance of the EnKF implies that filter performance is unaffected by linear or affine transformations.

We note that the neural EnKF differs from several existing latent-space DA approaches in important ways. Approaches such as those in [38,39] primarily use NNs to construct reduced-order latent representations and surrogate dynamics for efficient forecast evolution within DA. In contrast, this work is directly motivated by the analysis step itself. Rather than replacing the forward solver or evolving latent dynamics, NNs are used to learn representations of compressible flows with shocks in which ensemble-based Bayesian updates become more reliable. In this sense, our work shares similarities with recent efforts aimed at learning latent representations more compatible with Kalman-based filtering [40]. Our work also differs from recent studies aiming to improve the analysis step, where NNs are utilized either to enhance specific terms in the EnKF update formula [41] or to directly learn the analysis mapping [42], while still performing the analysis step in the original state space.

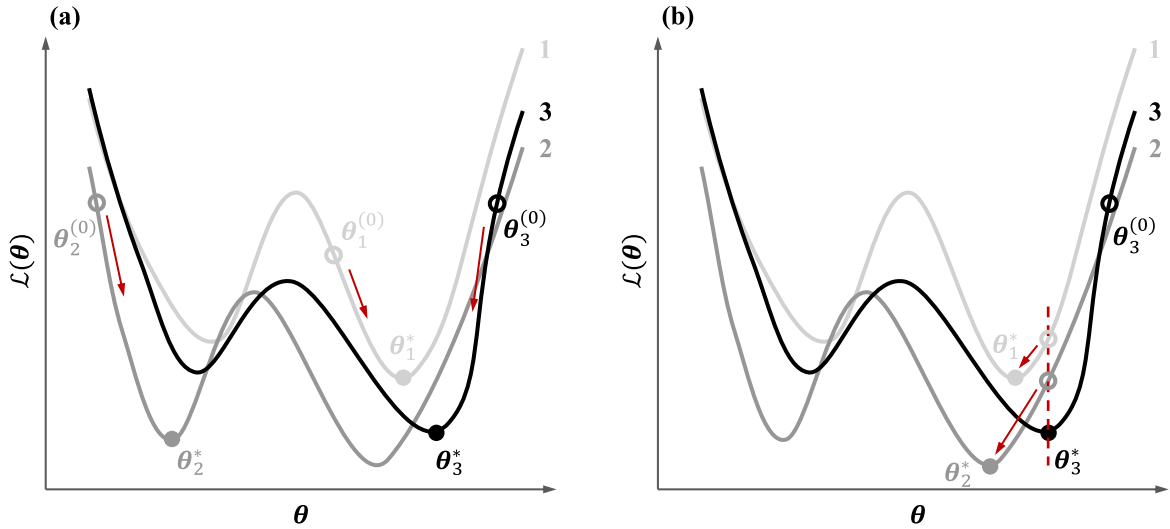


Fig. 3. Schematic illustration of neural-network training across ensemble members on a non-convex loss landscape. (a) Independent training: each ensemble member is initialized from a different random starting point $\theta_i^{(0)}$ and optimized independently, leading to convergence toward distinct local minima θ_i^* and a poorly aligned ensemble in neural space (e.g., ensemble members 1 and 3 converge to nearby minima, whereas member 2 converges to a separated minimum). (b) Transfer learning: ensemble members are initialized from the trained parameters of a nearby ensemble member (e.g., θ_3^*), guiding successive optimization processes toward neighboring regions of the loss landscape and promoting alignment of ensemble parameters in neural space.

3.2. Navigating non-convex loss landscapes with transfer learning

While the neural EnKF provides a neural-space formulation for updating ensemble members, constructing the forecast ensemble in this space requires training a separate neural network for each member. This introduces an additional challenge: neural networks used to represent flow fields are typically overparameterized, resulting in highly non-convex optimization landscapes with many equivalent or near-equivalent minima. As a result, independently trained networks can converge to disparate locations in neural space despite representing nearly identical physical states.

This behavior is illustrated schematically in Fig. 3(a) using three ensemble members as an example, where each curve represents the loss landscape associated with one ensemble member. Under independent training, each network is initialized from a distinct random parameter vector $\theta_i^{(0)}$ and optimized separately, following its own trajectory toward a local minimum θ_i^* in the neural space. Owing to the highly non-convex loss landscape, these minima may be widely separated, even when the corresponding physical states are nearly identical. As a result, the ensemble of trained networks exhibits poorly aligned parameters, which distorts the ensemble covariance structure in neural space and undermines the effectiveness of subsequent neural EnKF update.

To address this issue, we introduce a transfer learning strategy, as illustrated in Fig. 3(b). In this approach, the training of each ensemble member is initialized using the trained parameters of a nearby ensemble member (e.g., θ_3^*), thereby encouraging successive optimization processes to converge toward neighboring local minima. While this behavior cannot be guaranteed in a non-convex setting, the approach promotes parameter alignment in neural space by reducing divergence arising from random initialization. As a result, the ensemble of trained networks exhibits a more coherent organization in neural space, such that variations among ensemble members more faithfully reflect differences in the underlying physical states rather than artifacts of independent optimization. This improved alignment preserves a meaningful ensemble covariance structure, which is essential for stable and effective neural EnKF updates.

3.3. Nearest-neighbor chain training

We formalize the concept of transfer learning through a nearest-neighbor chain training strategy, in which the ensemble of neural networks is trained sequentially along a chain constructed based on physical-space similarities among ensemble members.

The construction of the nearest-neighbor chain is illustrated schematically in Fig. 4. Starting from the initial ensemble configuration (panel a), the construction begins by selecting the most “central” ensemble member (panel b), defined as the one with the smallest average distance to all other members under the chosen metric. The chain is then built iteratively by appending, at each step, the ensemble member that is closest to the currently selected set (panels c-h). Repeating this procedure yields an ordered chain that links all ensemble members and defines the sequence in which they are trained. It is important to note that the training initialization for a newly added ensemble member is not inherited from its immediate predecessor in the chain, but from its nearest neighbor within the set of previously selected members. For example, in panel (e), the newly added ensemble member 4 is initialized from member 1, its

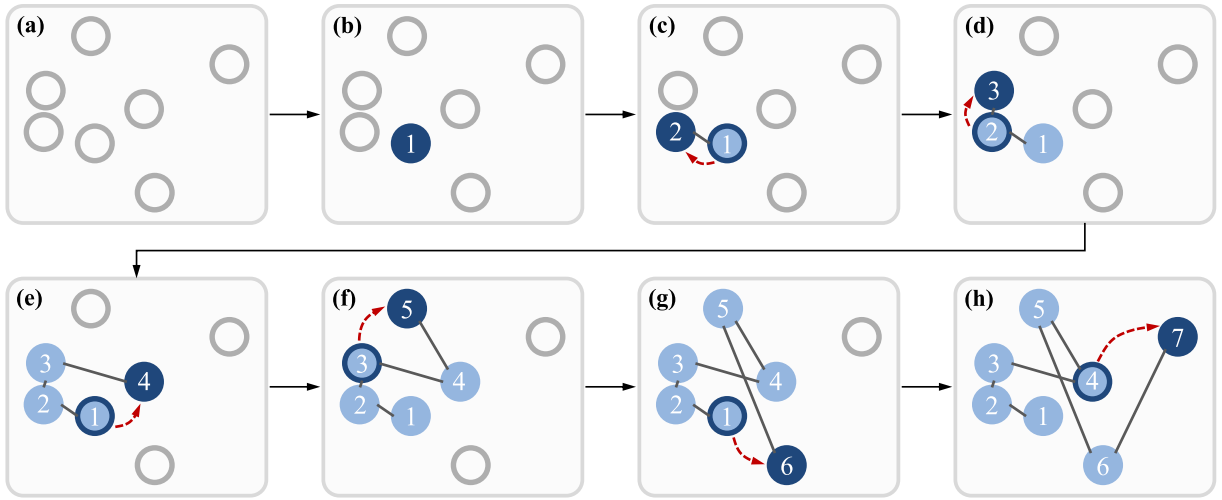


Fig. 4. Schematic illustration of the nearest-neighbor chain construction: (a) ensemble members in the chosen metric space; (b) initialization of the chain by selecting the most “central” ensemble member, defined as the member with the smallest average distance to all others; and (c)–(h) iterative construction of the chain, where at each step the ensemble member closest to the currently selected set is appended. Unselected members are shown as grey circles, previously selected members in light blue, and the member added at the current step is highlighted in dark blue. Solid/black lines indicate the resulting nearest-neighbor chain, with numbers denoting the order of insertion. Dashed/red arrows indicate the parent–child relationship used for training initialization, pointing from the nearest previously selected member (shown in light blue with a dark outline) to the newly added member. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

nearest neighbor within the selected set 1, 2, 3, rather than from member 3, which immediately precedes it in the chain. By enforcing this local nearest-neighbor dependency, the construction ensures that each network is initialized from the most relevant predecessor in physical space.

The nearest-neighbor chain construction described above is formalized in Algorithm 1. Given the forecast ensemble in physical space $\{z_i^f\}_{i=1}^{n_e}$ and a prescribed distance metric $D(\cdot, \cdot)$, the algorithm produces two ordered sequences: a training chain $\text{ORDER} = [i_1, \dots, i_{n_e}]$, which specifies the order in which ensemble members are parameterized through neural-network training, and a parent chain $\text{PARENT} = [q_1, \dots, q_{n_e}]$, which identifies, for each member, the most similar previously selected ensemble member from which training is initialized.

Algorithm 1 Nearest-neighbor chain construction.

- 1: **Input:** Forecast ensemble $\{z_i^f\}_{i=1}^{n_e}$; distance metric $D(\cdot, \cdot)$
 - 2: **Output:** Training chain $\text{ORDER} = [i_1, \dots, i_{n_e}]$; Parent chain $\text{PARENT} = [q_1, \dots, q_{n_e}]$ with $q_1 = -1$

 - Step 1: Identify the most “central” ensemble member**
 - 3: Compute pairwise distances $D_{jk} = D(z_j^f, z_k^f)$ for $1 \leq j, k \leq n_e$
 - 4: $i_1 \leftarrow \arg \min_j \sum_{k=1}^{n_e} D_{jk}$
 - 5: Initialize $\mathcal{A} \leftarrow \{i_1\}$; $\text{ORDER} \leftarrow [i_1]$; $q_1 \leftarrow -1$ ▷ first network, no parent

 - Step 2: Construct the nearest-neighbor chain**
 - 6: **for** $m = 2$ to n_e **do**
 - 7: **for** each $r \in \{1, \dots, n_e\} \setminus \mathcal{A}$ **do**
 - 8: $\hat{a}(r) \leftarrow \arg \min_{a \in \mathcal{A}} D_{ra}$ ▷ nearest selected neighbor of r
 - 9: $d(r) \leftarrow D_{r, \hat{a}(r)}$
 - 10: **end for**
 - 11: $i_m \leftarrow \arg \min_r d(r)$ ▷ next member added to chain
 - 12: $q_m \leftarrow \hat{a}(i_m)$ ▷ parent of i_m
 - 13: Append i_m to ORDER and q_m to PARENT
 - 14: $\mathcal{A} \leftarrow \mathcal{A} \cup \{i_m\}$ ▷ update selected set
 - 15: **end for**
-

The construction proceeds in two steps. In the first step, the most “central” ensemble member—defined as the member with the smallest total (or equivalently, average) distance to all other ensemble members—is selected to initialize the chain. Distances are

measured using the normalized Euclidean metric

$$D(\mathbf{z}_j^f, \mathbf{z}_k^f) = \sqrt{\frac{1}{n_z} \sum_{\ell=1}^{n_z} [\mathbf{z}_j^f - \mathbf{z}_k^f]_{\ell}^2}, \quad (11)$$

where n_z denotes the dimension of the state \mathbf{z} and $[\mathbf{a}]_{\ell}$ denotes the ℓ th component of a vector \mathbf{a} . This member is then added to the selected subset \mathcal{A} and assigned no parent ($q_1 = -1$).

In the second step, the remaining ensemble members are appended to the chain iteratively based on their similarity to the selected subset. At each iteration, all unselected members are compared to the currently selected set, and the member that is closest to this set under the prescribed distance metric is added next. The nearest previously selected member is recorded as its parent and serves as the source of initialization for training. The selected subset \mathcal{A} is then updated, and this procedure is repeated until all ensemble members are included in the chain.

Once the training chain ORDER and the corresponding parent chain PARENT are established, the neural networks are trained sequentially following ORDER. The first member, corresponding to the most ‘‘central’’ member, has no parent and is trained independently from scratch. Each subsequent member is initialized from its parent as specified by PARENT and trained with a reduced learning rate to limit parameter drift in the neural space. Training is terminated once a prescribed fitting error threshold is reached, typically on the order of 10^{-6} . This transfer learning strategy promotes smooth variation of network parameters across the ensemble and yields well-behaved ensemble statistics that support stable EnKF updates.

We note that the nearest-neighbor chain can naturally be generalized to a tree-like structure, in which multiple ensemble members share the same parent ensemble member. In this setting, once a parent ensemble member has been trained, its child ensemble members can be trained in parallel, thereby further improving training efficiency. We leave a systematic optimization of the transfer-learning process for future work and focus here on how the neural EnKF can overcome difficulties in DA for shock-laden flows.

3.4. Revisiting the motivation example

We revisit the hyperbolic-tangent surrogate shock example from Section 2 to assess the effectiveness of the neural EnKF with nearest-neighbor chain training. We compare the neural EnKF updates under two training strategies: independent training with random initialization and the proposed nearest-neighbor chain training, as illustrated in Fig. 5. The comparison demonstrates that nearest-neighbor chain training enables a stable, structure-preserving neural EnKF update, whereas independent training with random initialization does not.

As shown in panels (a1) and (a2), both training strategies provide accurate neural-network representations for the forecast ensemble members. Under independent training with random initialization, however, the subsequent neural EnKF update produces a severely distorted analysis ensemble (b1). These distortions are not confined to the shock region but extend across the entire domain, including smooth regions where no sharp features are present. This behavior reflects a failure of the neural-space update and, in this example, results in analysis states that are even worse than those obtained using the EnKF (see Fig. 1(b)). The origin of this failure is evident in the neural-network parameter variations shown in panel (c1). Under independent training, the parameter values across ensemble members exhibit highly irregular and non-smooth variations, indicating poor alignment in neural space, which in turn leads to ineffective neural EnKF updates. In contrast, panel (c2) exhibits smooth parameter variations across ensemble members under nearest-neighbor chain training. Most neural-network parameters take nearly identical values across the ensemble, with only small and gradually varying deviations along the chain, indicating a highly concentrated and well-aligned parameter distribution in neural space. This alignment leads to effective neural EnKF updates, yielding a structure-preserving analysis ensemble shown in panel (b2) without spurious oscillations.

4. Numerical experiments with neural EnKF

The neural EnKF is evaluated on three test cases of increasing complexity and consistently demonstrates robust DA performance, effectively preserving sharp flow structures while avoiding spurious oscillations. The test suite consists of three representative problems: (1) the inviscid Burgers’ equation; (2) the Sod shock tube; and (3) a 2D blast wave.

4.1. Numerical experiments on the inviscid Burgers’ equation

We consider the inviscid Burgers’ equation

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0, \quad (12)$$

on the domain $x \in [0, 1]$ with periodic boundary conditions, where u denotes the scalar velocity field. The reference initial velocity profile is piecewise linear and contains multiple sharp features, including a small trapezoidal segment on the left (Fig. 6(a)).

The domain is discretized using 400 uniformly spaced grid points. The equation is solved using the PyClaw library [43] and advanced to a final time $T = 0.5$ with a fixed time step of $\Delta t = 2 \times 10^{-4}$.

To model uncertainty in the initial condition, we construct an ensemble of $n_e = 40$ members. While the reference initial profile contains the left trapezoidal feature, several ensemble members omit this structure, resulting in pronounced structural heterogeneity across the ensemble (Fig. 6(a)), where the members missing the left trapezoidal segment are highlighted in red). Consequently, the

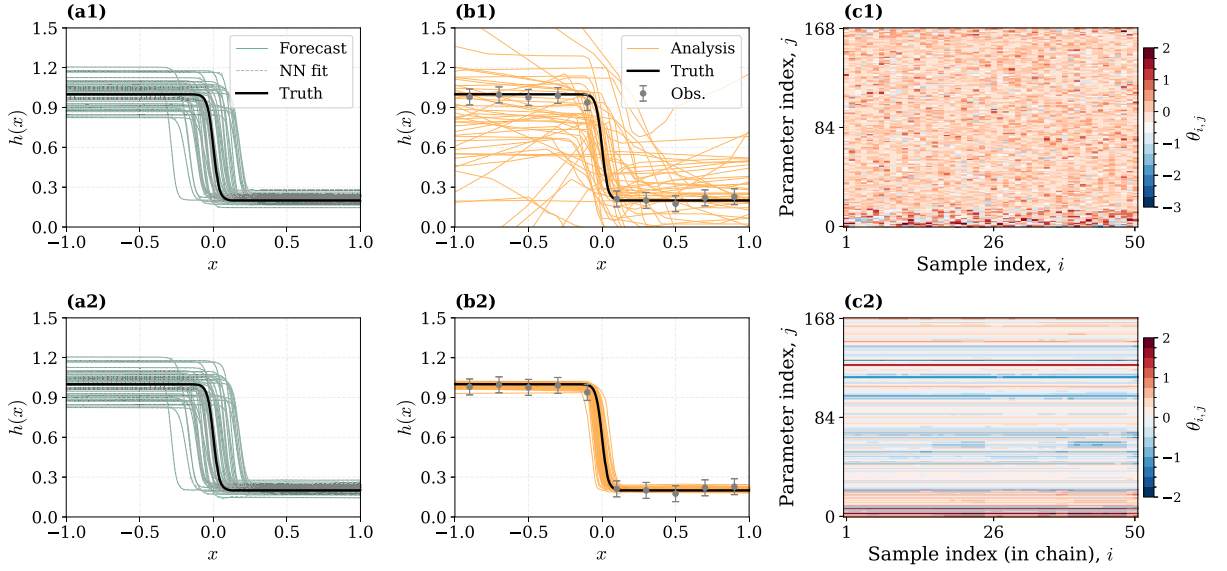


Fig. 5. Comparison of neural EnKF updates under two training strategies for the hyperbolic-tangent surrogate shock example. Panels (a1–c1) correspond to *independent training with random initialization*, while panels (a2–c2) correspond to the proposed *nearest-neighbor chain training*. (a1) Forecast ensemble together with the corresponding neural-network fits for each ensemble member, indicating negligible fitting errors. (b1) Analysis ensemble obtained using independent training, exhibiting irregular and spatially incoherent updates across ensemble members. (c1) Variation of neural-network parameters across ensemble members under independent training, showing highly irregular variability and poor alignment of parameters in neural space. (a2–b2) Forecast and analysis ensembles obtained using nearest-neighbor chain training, in which the analysis ensemble preserves the shock-like structure and better reflects the observation data. (c2) Smooth neural-network parameter variations under nearest-neighbor chain training, showing well-aligned parameter distributions in neural space. In panels (c1) and (c2), the horizontal axis indexes the ensemble members (50 samples), while the vertical axis indexes the flattened neural-network parameters (169 trainable parameters).

ensemble does not share a consistent representation of the underlying solution features. This configuration provides a stringent test for DA methods, as it violates the structural similarity assumptions typically required by feature alignment-based approaches (e.g., [33]).

The large initial uncertainty is progressively reduced through sequential DA using sparse, noisy observations. Neural EnKF updates are applied at regular intervals of $\Delta t_{\text{obs}} = 0.01$ starting at $t = 0.01$, resulting in 50 DA cycles. At each DA step, each forecast ensemble member $\mathbf{z}_i^f = \mathbf{u}_i^f$ is represented by a neural network with parameters θ_i^f , with the network architecture detailed in Appendix A. Synthetic observations \mathbf{d} are generated by sampling the true velocity field \mathbf{u}^* at 12 uniformly distributed interior locations over the domain, i.e., $x_k = 1/24 + k/12$ for $k = 0, \dots, 11$. The resulting values are perturbed by additive Gaussian noise,

$$\mathbf{d} = \mathbf{H}\mathbf{u}^* + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \quad (13)$$

where \mathbf{R} is a diagonal covariance matrix with entries $R_{kk} = (\max(0.05 [\mathbf{H}\mathbf{u}^*]_k, 5 \times 10^{-3}))^2$. Consistent with the observation model, the predicted observations for each forecast ensemble member are

$$\mathbf{y}_i = \mathbf{H}\mathbf{u}_i^f. \quad (14)$$

Given the ensemble of network parameters $\{\theta_i^f\}_{i=1}^{40}$, the corresponding predicted observations $\{\mathbf{y}_i\}_{i=1}^{40}$, and the synthetic observations \mathbf{d} , the neural EnKF update is performed according to Eq. (9). The updated network parameters are then mapped back to physical space to obtain the analysis ensemble, as described in Eq. (10).

The neural EnKF exhibits stable and effective DA performance in this challenging setting, with rapid convergence toward the true solution, as shown in Fig. 6. After the first DA cycle, the analysis ensemble shows a marked reduction in spread and improved agreement with the true profile over most of the domain, including partial reconstruction of the missing left trapezoidal feature in the members that initially lack this structure. By the 20th DA cycle, the ensemble continues to approach the true solution, with these members progressively recovering the absent feature. By the 50th DA cycle, the analysis ensemble closely matches the true velocity profile across the entire domain, indicating robust recovery despite the substantial initial structural mismatch.

The DA performance is further quantified by examining the evolution of the root-mean-square error (RMSE) and the ensemble spread over successive DA cycles. The RMSE is defined as

$$\text{RMSE} = \sqrt{\frac{1}{n_z} \sum_{\ell=1}^{n_z} [\bar{\mathbf{z}}^a - \mathbf{z}^a]_{\ell}^2}, \quad \bar{\mathbf{z}}^a = \frac{1}{n_e} \sum_{i=1}^{n_e} \mathbf{z}_i^a, \quad (15)$$

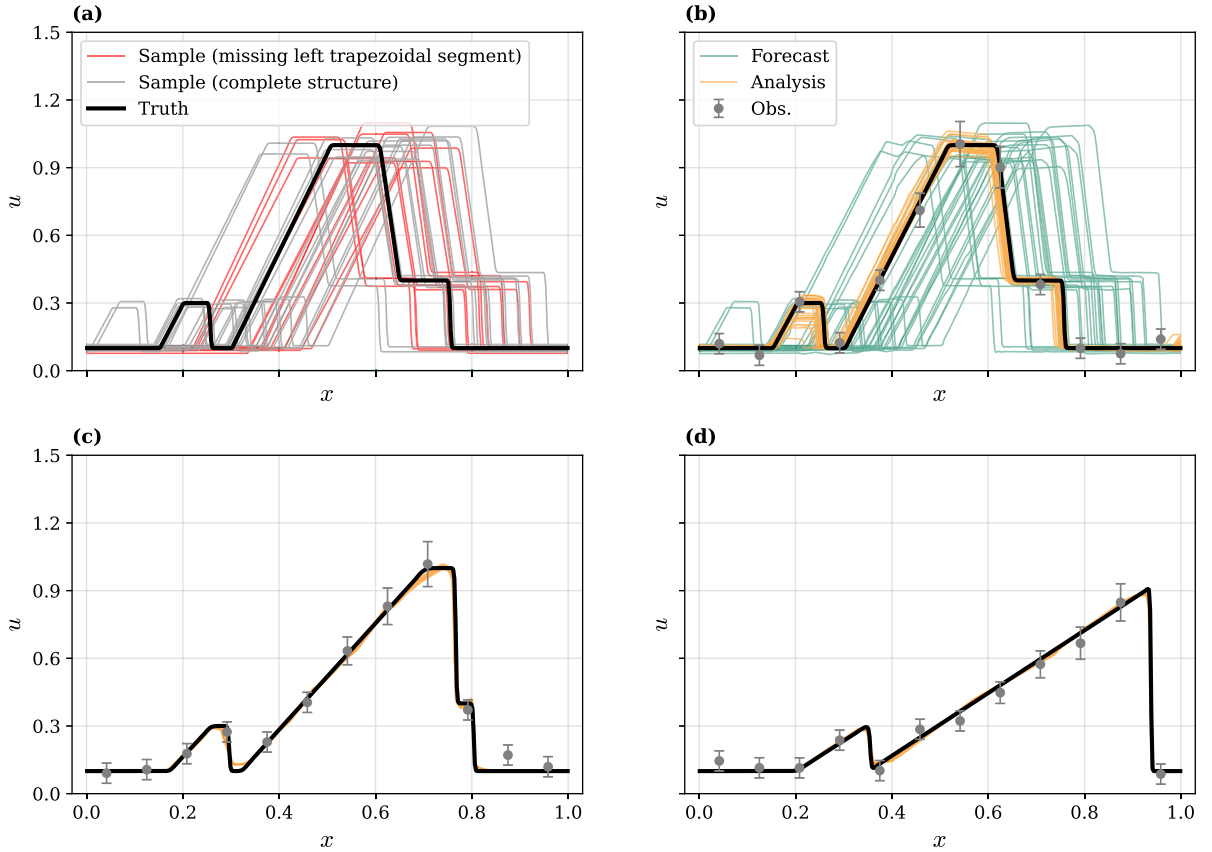


Fig. 6. DA results for the inviscid Burgers’ equation with a structurally heterogeneous initial ensemble under neural EnKF updates. (a) Initial ensemble, where several members (red) lack the left trapezoidal segment present in the true profile (black), while the remaining members (grey) retain the complete structure. (b) Forecast and analysis ensembles at the first DA cycle, illustrating rapid recovery of the overall profile. (c) and (d) Analysis ensembles at the 20th and 50th DA cycles, respectively, showing progressive structural convergence toward the true solution during intermediate and late stages of assimilation. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

where \bar{z}^a and z^* denote the analysis ensemble mean and true state at a given DA step. The ensemble spread is defined as

$$\text{Spread} = \sqrt{\frac{1}{n_e} \text{trace}(\mathbf{P}^a)}, \quad \mathbf{P}^a = \frac{1}{n_e - 1} \sum_{i=1}^{n_e} (z_i^a - \bar{z}^a)(z_i^a - \bar{z}^a)^\top. \tag{16}$$

As shown in Fig. 7, both the RMSE and the spread decrease rapidly during the initial DA cycles and subsequently stabilize at comparably low levels. Importantly, the close agreement between the RMSE and the spread at later DA stages (i.e., $\text{RMSE} \approx \text{spread}$) indicates consistency between the estimation error and the ensemble uncertainty. We note that here the analysis ensemble mean is computed by averaging the reconstructed ensemble members in physical space. Alternatively, one may first compute the ensemble mean in the neural space and then reconstruct the corresponding physical state, which may yield sharper shock features in the ensemble mean.

4.2. Numerical experiments on the Sod shock tube problem

We consider the Sod shock tube problem [44], governed by the compressible Euler equations in conservative form,

$$\frac{\partial \mathbf{W}}{\partial t} + \nabla \cdot \mathcal{F}(\mathbf{W}) = 0, \tag{17}$$

where \mathbf{W} denotes the conservative state vector and $\mathcal{F}(\mathbf{W})$ denotes the corresponding advective fluxes. These are defined as

$$\mathbf{W} = \begin{bmatrix} \rho \\ \rho \mathbf{V} \\ \rho E \end{bmatrix}, \quad \mathcal{F}(\mathbf{W}) = \begin{bmatrix} \rho \mathbf{V}^\top \\ \rho \mathbf{V} \otimes \mathbf{V} + p \mathbf{I} \\ (\rho E + p) \mathbf{V}^\top \end{bmatrix}. \tag{18}$$

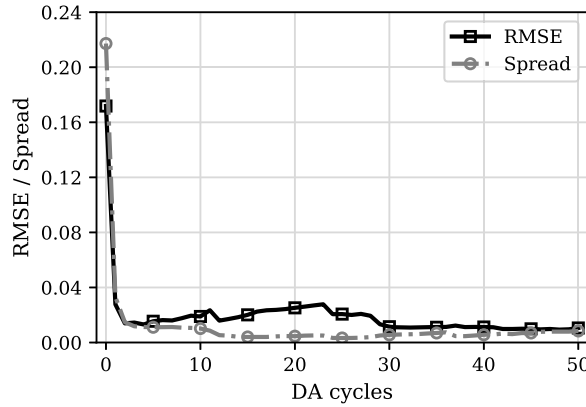


Fig. 7. Evolution of RMSE and ensemble spread over DA cycles for the inviscid Burgers’ equation with an initial ensemble exhibiting structural heterogeneity. Both metrics decrease rapidly during the initial assimilation stage and subsequently stabilize at comparable low levels. Markers are shown every five DA cycles for clarity.

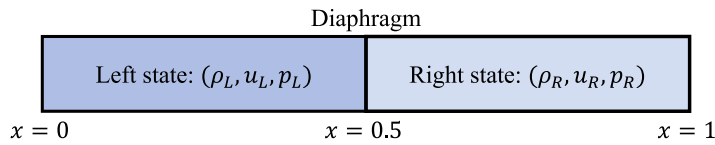


Fig. 8. Schematic of the reference shock tube configuration. The diaphragm at $x_d = 0.5$ separates the high-pressure left state (ρ_L, u_L, p_L) and the low-pressure right state (ρ_R, u_R, p_R) .

where ρ and p are the density and pressure, $\mathbf{V} = [u, v, w]^T$ is the velocity vector, \mathbf{I} is the identity matrix, and the operator \otimes denotes the outer product. The total energy per unit mass is

$$E = e + \frac{1}{2}|\mathbf{V}|^2, \tag{19}$$

where e is the internal energy per unit mass and $|\mathbf{V}|$ denotes the Euclidean norm of the velocity vector. The system is closed by the ideal-gas equation of state,

$$p = (\gamma - 1)\rho e, \tag{20}$$

with $\gamma = 1.4$. In the 1D setting considered here, the velocity reduces to $\mathbf{V} = [u, 0, 0]^T$.

The problem is considered on the domain $x \in [0, 1]$, with non-reflecting boundary conditions imposed at both ends to permit outgoing waves to exit without artificial reflection. The reference initial condition consists of a diaphragm located at $x_d = 0.5$ separating two constant states,

$$(\rho_L, u_L, p_L) = (1.0, 0, 1.0), \quad (\rho_R, u_R, p_R) = (0.125, 0, 0.1), \tag{21}$$

corresponding to high- and low-pressure regions, respectively, as illustrated in Fig. 8.

The governing equations are discretized on a uniform grid of 400 cells and solved using the open-source solver M2C (Multiphysics Modeling and Computation) [45,46]. Numerical fluxes are computed using a Roe-type approximate Riemann solver. Time integration is performed using an explicit second-order Runge–Kutta scheme with a fixed time step of $\Delta t = 5 \times 10^{-4}$, corresponding to a Courant–Friedrichs–Lewy (CFL) number of approximately 0.4. The simulation is advanced to a final time of $T = 0.2$, a standard configuration for this benchmark problem.

To represent uncertainty in the initial condition, we construct an ensemble of $n_e = 40$ perturbed states by independently sampling the left- and right-state parameters, as well as the diaphragm location, from truncated Gaussian distributions:

$$\begin{aligned} \rho_L &\sim \mathcal{N}_{(0,\infty)}(0.95, 0.05^2), & p_L &\sim \mathcal{N}_{(0,\infty)}(1.05, 0.05^2), & x_d &\sim \mathcal{N}_{(0.1,0.9)}(0.55, 0.2^2), \\ \rho_R &\sim \mathcal{N}_{(0,\infty)}(0.13, 0.01^2), & p_R &\sim \mathcal{N}_{(0,\infty)}(0.09, 0.01^2). \end{aligned} \tag{22}$$

Each sampled parameter set defines one ensemble member, as illustrated in the top row of Fig. 9. We note that the means of the sampling distributions are intentionally biased relative to the reference values, a setting that may better reflect practical DA scenarios and was not examined in some related studies (e.g., [33]).

The large initial uncertainty is progressively reduced through sequential DA using the neural EnKF with sparse, noisy pressure observations. Neural EnKF updates are applied every $\Delta t_{\text{obs}} = 0.025$ starting at $t = 0.025$, yielding eight DA steps. At each DA step, each forecast ensemble member $\mathbf{z}_i^f = [\rho_i^f, u_i^f, p_i^f]^T$ represents the full state vector, consisting of the density, velocity, and pressure components. The entire state is parameterized by a single neural network with parameters θ_i^f , which implicitly represents the coupled flow variables.

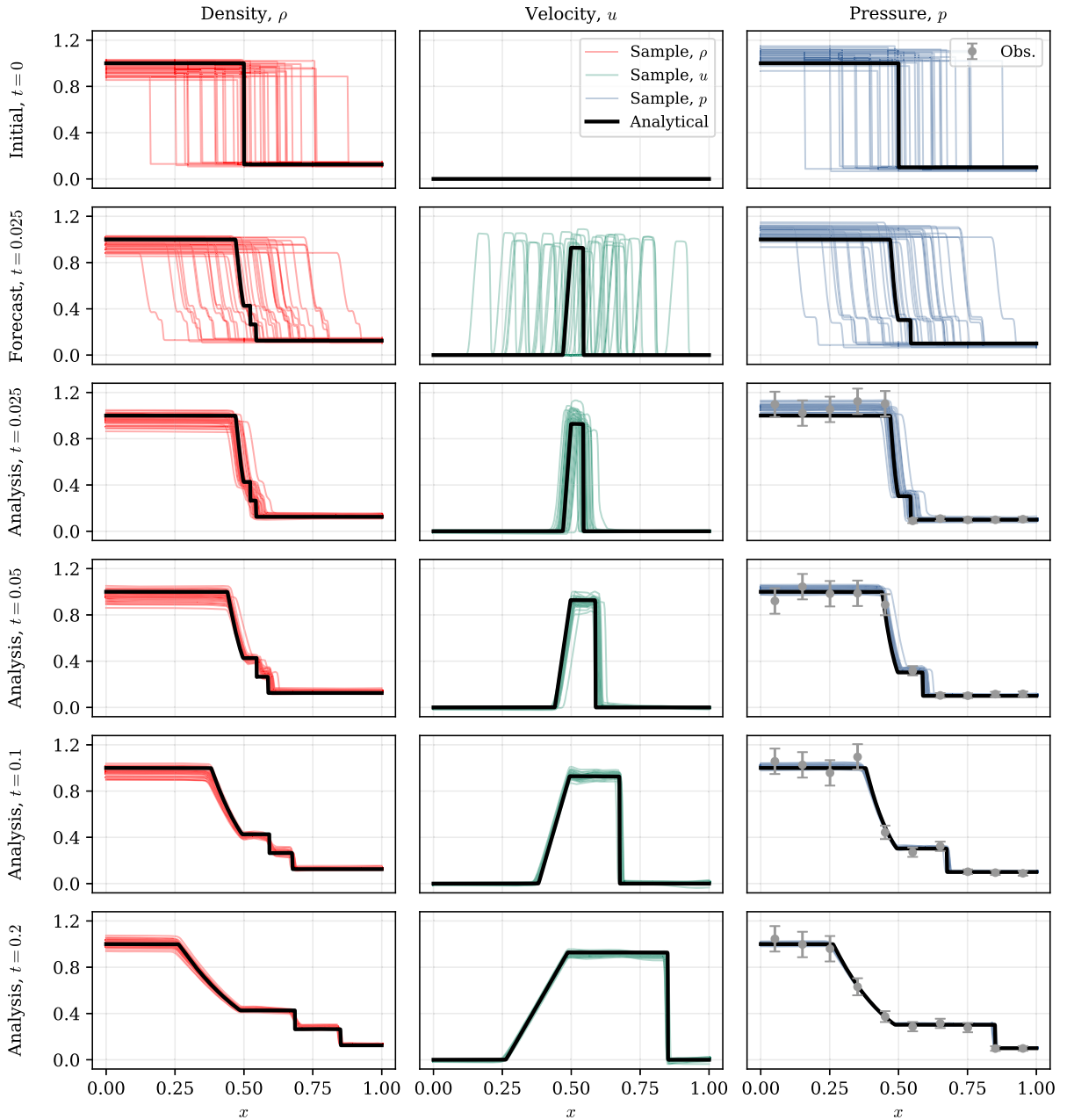


Fig. 9. Analysis ensemble for the Sod shock tube problem at representative DA steps, showing increasing agreement with the analytical solution. Columns show density ρ , velocity u , and pressure p . Rows correspond to the initial ensemble ($t = 0$), the forecast ensemble prior to the first DA step ($t = 0.025$), and the analysis ensembles at the 1st, 2nd, 4th, and 8th DA steps ($t = 0.025, 0.05, 0.1, \text{ and } 0.2$). The density field exhibits three flow features, from left to right: a rarefaction wave, a contact interface, and a shock, whereas the velocity and pressure fields contain a rarefaction wave and a shock but do not include a contact discontinuity.

Synthetic pressure observations \mathbf{d} are constructed by sampling the analytical pressure solution p^* (evaluated on the same spatial grid as the numerical solution) at 10 uniformly spaced locations $x = 0.05, 0.15, \dots, 0.95$, and perturbing the sampled values with additive Gaussian noise:

$$\mathbf{d} = \mathbf{H}p^* + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \tag{23}$$

where \mathbf{R} is a diagonal covariance matrix with entries $R_{kk} = (0.05[\mathbf{H}p^*]_k + 0.01)^2$. We note that the analytical solution is used to generate the synthetic observations, whereas forward simulations are performed using a finite-resolution numerical solver. This deliberate model–data mismatch introduces a more realistic and challenging DA setting. Consistent with the observation model, the

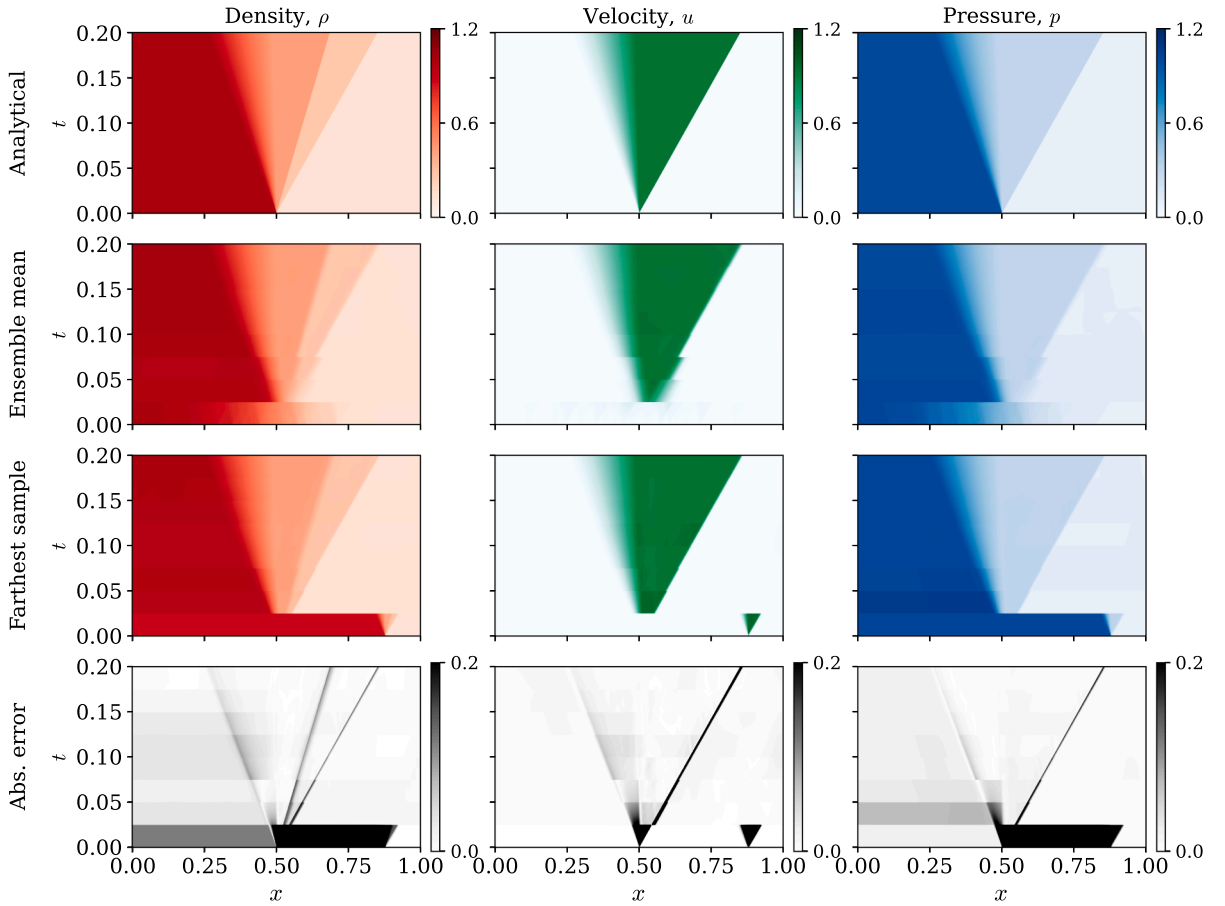


Fig. 10. Comparison of the space–time evolution under DA for the Sod shock tube problem, with neural EnKF updates applied every $\Delta t_{\text{obs}} = 0.025$. Columns show density ρ , velocity u , and pressure p . Rows correspond to the analytical solution, ensemble mean, a “farthest” ensemble member, and the absolute error between the analytical solution and the farthest ensemble member. The farthest ensemble member, defined once as the last member in the nearest-neighbor chain at the first DA step, corresponds to the member with the largest initial structural mismatch. The analytical solution and error fields are plotted on a fine grid with 3200 cells to enhance visualization of sharp discontinuities, while the ensemble mean and farthest ensemble member are shown on the numerical grid with 400 cells.

predicted observations associated with each forecast ensemble member are given by

$$\mathbf{y}_i = \mathbf{H}\mathbf{p}_i^f. \quad (24)$$

Given the ensemble of neural-network parameters $\{\theta_i^f\}_{i=1}^{40}$, the corresponding predicted observations $\{\mathbf{y}_i\}_{i=1}^{40}$, and the synthetic pressure observations \mathbf{d} , the neural EnKF update is performed according to Eq. (9). The updated network parameters are then mapped back to physical space through the reconstruction described in Eq. (10).

The neural EnKF demonstrates robust DA performance, progressively recovering the true solution over successive assimilation steps without introducing spurious oscillations. For comparison, results obtained with the EnKF are reported in Appendix B, where strong oscillations and nonphysical states (negative density and pressure) are observed. Fig. 9 illustrates the behavior of the neural EnKF through the evolution of the ensemble across successive assimilation steps. Prior to the first DA update ($t = 0.025$), the forecast ensemble exhibits substantial spread and pronounced misalignment of discontinuities, reflecting significant uncertainty in the locations of the shock, contact interface, and rarefaction wave. Following the first DA step, the analysis ensemble contracts noticeably and exhibits improved alignment of the dominant flow features. With successive DA updates, the ensemble progressively sharpens around the analytical solution, and the spread associated with misaligned discontinuities is systematically reduced. By the later assimilation times, the ensemble members are tightly clustered and the recovered solution accurately captures the rarefaction wave, contact interface, and shock without introducing spurious oscillations.

The convergence behavior, however, differs across state variables. The pressure field, which is directly observed, converges most rapidly, while the velocity field exhibits a comparable convergence behavior due to its strong dynamical coupling with pressure through the momentum equation. In contrast, the density field shows a more gradual reduction of ensemble uncertainty: although the analysis ensemble progressively recovers the contact interface and overall solution structure, the contraction of spread lags behind that of pressure and velocity. This behavior is expected, since pressure observations alone do not uniquely determine the density field

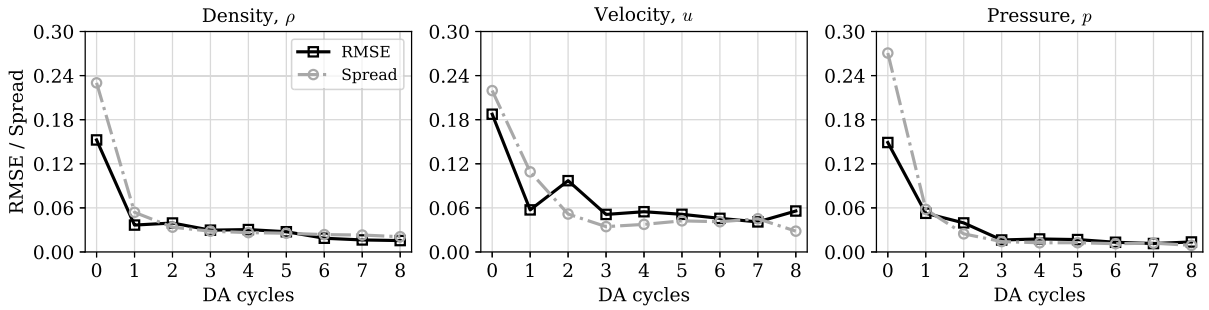


Fig. 11. RMSE and ensemble spread over successive DA cycles for the Sod shock tube problem under the neural EnKF, showing rapid reduction of both metrics across all three variables during the first few DA cycles and sustained comparably low levels thereafter. The value at cycle 0 is computed from the forecast ensemble prior to the first DA update ($t = 0.025$), while cycles 1–8 correspond to the analysis ensembles after successive DA steps.

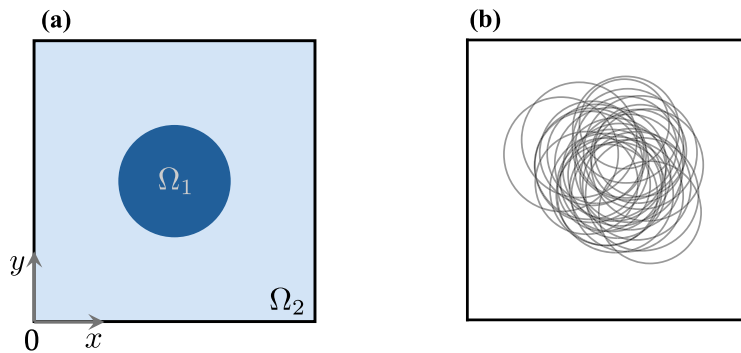


Fig. 12. Schematic illustration of the 2D blast-wave initial configuration and ensemble sampling. (a) Computational domain $\Omega = \Omega_1 \cup \Omega_2$, where Ω_1 denotes the circular high-pressure blast region and Ω_2 the surrounding low-pressure region. (b) Ensemble of sampled initial discontinuity locations obtained by Gaussian sampling of the blast center and radius. For clarity, uncertainty in the density and pressure within Ω_1 is not shown.

through the equation of state without additional thermodynamic information, such as temperature or internal energy. As a result, multiple density–energy configurations can remain consistent with the assimilated pressure observations, limiting the contraction of density uncertainty and leading to a slower reduction of spread. A more detailed analysis of this effect is provided in [Appendix C](#).

To further illustrate the dynamical behavior under DA, [Fig. 10](#) compares the space–time evolution of the analytical solution with that of the ensemble mean and a “farthest” ensemble member over the full simulation window. Here, the ensemble mean represents a statistical state estimate rather than a physical solution. The farthest ensemble member is defined once as the last member in the nearest-neighbor chain constructed at the first DA step, corresponding to the ensemble member with the largest initial structural mismatch. This ensemble member is then tracked throughout the assimilation process and evolves according to the governing equations between successive DA updates. Since neural EnKF updates are applied at regular intervals of $\Delta t_{\text{obs}} = 0.025$, both the ensemble-mean and farthest-member evolutions exhibit a piecewise-in-time structure, with each segment corresponding to forward integration between DA steps. As DA proceeds, the ensemble mean (second row) becomes progressively better aligned with the analytical solution (top row), reflecting the improvement of the ensemble-averaged state estimate. The evolution of the farthest ensemble member (third row) provides complementary insight into the behavior of individual realizations under DA. Despite its unfavorable initial condition, this trajectory rapidly adjusts at the first DA step ($t = 0.025$) and subsequently tracks the analytical solution with increasing fidelity, indicating that the neural EnKF effectively corrects pronounced structural misalignment of dominant flow features even for extreme ensemble members. These trends are further highlighted by the absolute-error fields (bottom row), which show the point-wise differences between the analytical solution and the farthest ensemble member. While noticeable discrepancies are present near discontinuities at early times, they decrease systematically over successive DA cycles. By the final DA step, the remaining differences are confined to narrow regions around the shock and contact interface and are primarily attributable to the finite spatial resolution of the numerical solver rather than deficiencies of the neural EnKF.

The DA performance is further quantified using the RMSE and ensemble spread, as shown in [Fig. 11](#). Here, the value at cycle 0 is computed from the forecast ensemble prior to the first DA update ($t = 0.025$), while cycles 1–8 correspond to the analysis ensembles after successive DA steps. For all three state variables, both metrics decrease rapidly during the initial DA cycles, reflecting effective reduction of estimation error and ensemble uncertainty through assimilation. Following the initial spin-up phase, the RMSE and

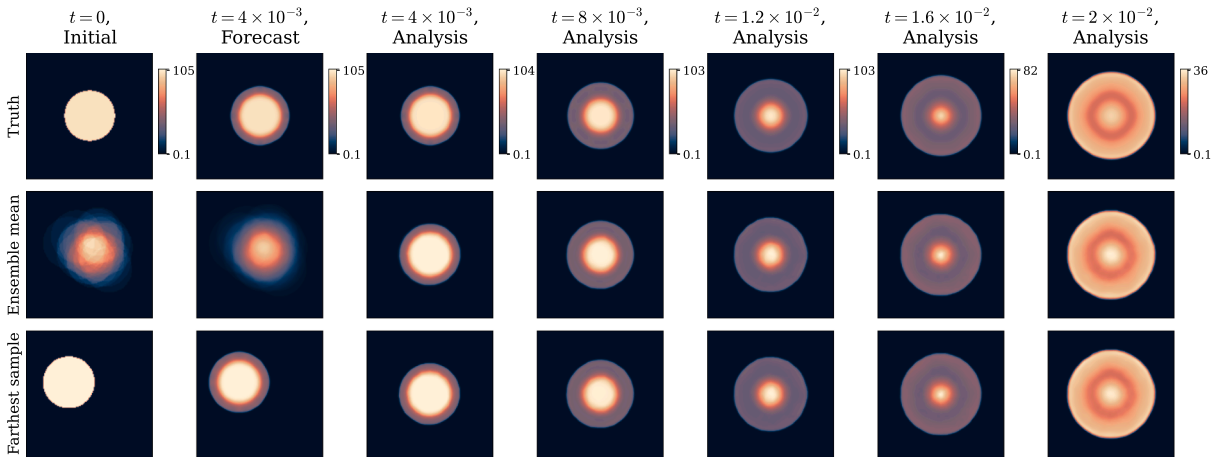


Fig. 13. Pressure fields for the 2D blast wave over the full assimilation window, demonstrating rapid correction after the first neural EnKF update and sustained agreement with the reference solution thereafter. Rows show the reference solution, the ensemble mean (statistical state estimate), and the farthest ensemble member, defined as the last member in the nearest-neighbor chain constructed at the first DA step. Columns correspond to the initial condition ($t = 0$), the forecast states prior to the first DA update ($t = 4.0 \times 10^{-3}$), and the analysis states after five successive DA steps up to the final time ($t = 2.0 \times 10^{-2}$).

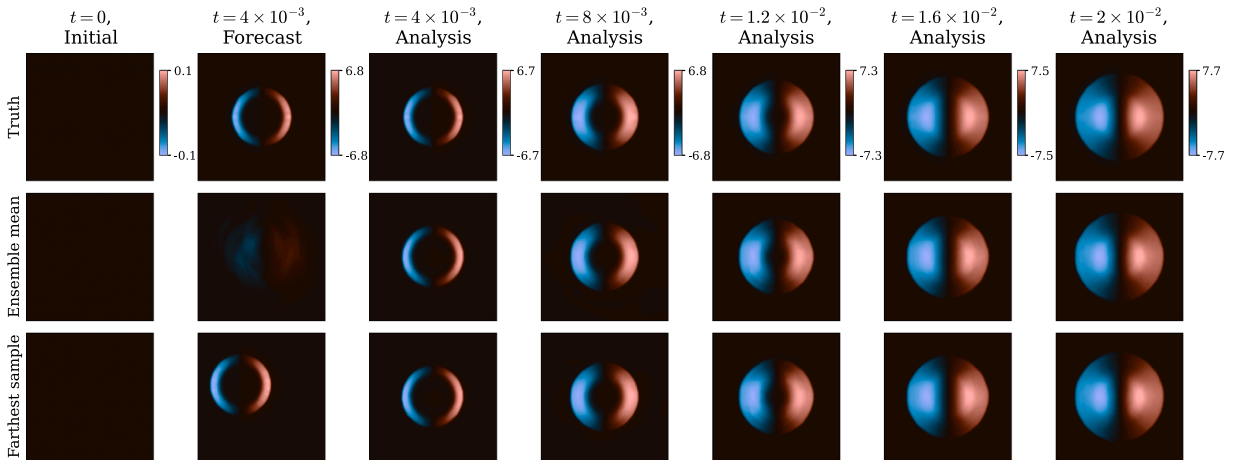


Fig. 14. Velocity fields (x -component) for the 2D blast wave over the full assimilation window, demonstrating effective correction after the first neural EnKF update and sustained agreement with the reference solution thereafter. Rows show the reference solution, the ensemble mean (statistical state estimate), and the farthest ensemble member. Columns correspond to the initial static condition ($t = 0$), the forecast states prior to the first DA update ($t = 4.0 \times 10^{-3}$), and the analysis states after five successive DA steps up to the final time ($t = 2.0 \times 10^{-2}$).

ensemble spread remain at comparably low levels across subsequent DA cycles, indicating that the state estimate remains reasonably accurate and that ensemble uncertainty is effectively controlled without collapse.

To further evaluate the robustness of the neural EnKF under various DA configurations, we perform a parametric study examining the effects of observation noise level, spatial density of observations, and observation frequency. Detailed experimental setups and results are provided in [Appendix D](#).

4.3. Numerical experiments on the blast wave problem

We consider a 2D blast wave problem governed by the same compressible Euler equations as in the shock tube case, i.e., [Eqs. \(17\)–\(20\)](#). In this 2D setting, the velocity field takes the form $\mathbf{V} = [u, v, 0]^T$.

The problem is considered on the domain $\Omega = [0, 1] \times [0, 1]$, with non-reflecting boundary conditions imposed on all boundaries to permit outgoing waves to exit without artificial reflection. The initial condition consists of a circular discontinuity separating two constant states, as illustrated in [Fig. 12\(a\)](#). Specifically, a high-pressure circular region Ω_1 is embedded in a low-pressure exterior region $\Omega_2 = \Omega \setminus \Omega_1$. The reference initial condition is prescribed as

$$(\rho_{\Omega_1}, u_{\Omega_1}, v_{\Omega_1}, p_{\Omega_1}) = (1.0, 0.0, 0.0, 100), \quad (\rho_{\Omega_2}, u_{\Omega_2}, v_{\Omega_2}, p_{\Omega_2}) = (1.0, 0.0, 0.0, 0.1), \quad (25)$$

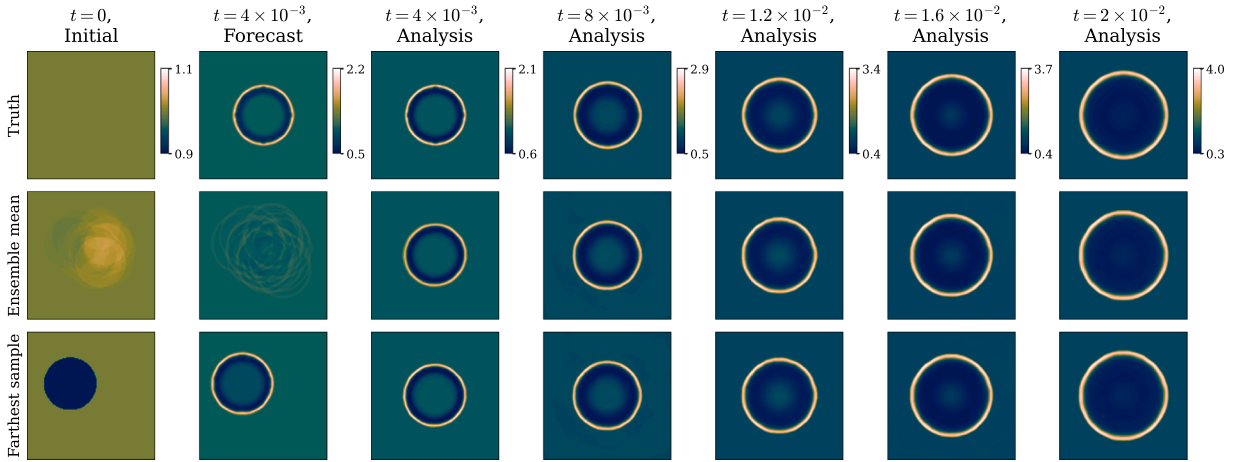


Fig. 15. Density fields for the 2D blast wave over the full assimilation window, demonstrating effective correction after the first two neural EnKF updates and sustained agreement with the reference solution thereafter. Rows show the reference solution, the ensemble mean (statistical state estimate), and the farthest ensemble member. Columns correspond to the initial condition ($t = 0$), the forecast state prior to the first DA update ($t = 4.0 \times 10^{-3}$), and the analysis states after five successive DA steps up to the final time ($t = 2.0 \times 10^{-2}$).

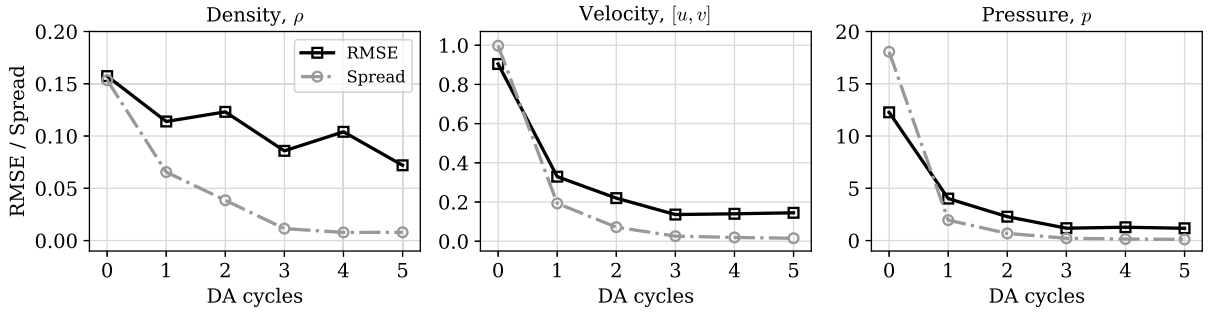


Fig. 16. Evolution of RMSE and ensemble spread over successive DA cycles for the 2D blast wave. Pressure and velocity exhibit a rapid reduction in both metrics after the first DA cycle, followed by sustained low levels. Density shows a comparable trend in ensemble spread, while its RMSE decreases more gradually and remains relatively low throughout the assimilation window. Cycle 0 corresponds to the forecast ensemble prior to the first DA update ($t = 4 \times 10^{-3}$), and cycles 1–5 denote analysis ensembles after successive DA steps.

where Ω_1 is a circular region centered at $(x_c, y_c) = (0.5, 0.5)$ with radius $r = 0.2$. This configuration generates a strong radially propagating blast wave.

The Euler equations are discretized on a uniform Cartesian grid of 128×128 cells. Time integration is performed using an explicit second-order Runge–Kutta scheme with a fixed time step of $\Delta t = 10^{-4}$, corresponding to a CFL number of approximately 0.2. The simulation is advanced to a final time of $T = 2.0 \times 10^{-2}$.

To represent uncertainty in the initial condition, we construct an ensemble of $n_e = 30$ perturbed states by independently sampling the parameters defining the blast wave configuration from truncated Gaussian distributions. Specifically, the blast center coordinates, radius, density, and pressure are sampled as:

$$\begin{aligned} x_c &\sim \mathcal{N}_{(0.3,0.7)}(0.53, 0.1^2), & y_c &\sim \mathcal{N}_{(0.3,0.7)}(0.53, 0.1^2), & r &\sim \mathcal{N}_{(0.16,0.24)}(0.19, 0.02^2), \\ \rho_{\Omega_1} &\sim \mathcal{N}_{(0,\infty)}(1.05, 0.1^2), & p_{\Omega_1} &\sim \mathcal{N}_{(0,\infty)}(102.0, 3.0^2). \end{aligned} \tag{26}$$

Each sampled parameter set defines one ensemble member, resulting in an ensemble of initial blast configurations. The induced variability in the initial discontinuity location is illustrated in Fig. 12(b). As in the Sod shock tube case, the means of the sampling distributions are intentionally biased relative to the corresponding reference values.

The resulting uncertainty is progressively reduced through sequential DA using the neural EnKF with sparse, noisy pressure observations. Neural EnKF updates are applied every $\Delta t_{\text{obs}} = 4.0 \times 10^{-3}$ starting at $t = 4.0 \times 10^{-3}$, yielding five DA steps. At each DA step, each forecast ensemble member $\mathbf{z}_i^f = [\rho_i^f, u_i^f, v_i^f, p_i^f]$ represents the full state vector, comprising the density, velocity components in the x - and y -directions, and pressure. The entire state is parameterized by a single neural network with parameters θ_i^f . Synthetic pressure observations \mathbf{d} are constructed by sampling the reference pressure field p^* on a uniform 7×7 grid of observation locations,

with coordinates $(x, y) \in \{0.05, 0.2, \dots, 0.95\} \times \{0.05, 0.2, \dots, 0.95\}$, and adding independent Gaussian noise:

$$\mathbf{d} = \mathbf{H}\mathbf{p}^* + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}). \quad (27)$$

where \mathbf{R} is a diagonal covariance matrix whose entries are given by $R_{kk} = (0.03 [\mathbf{H}\mathbf{p}^*]_k + 10^{-3})^2$. Consistent with the observation model, the predicted observations for each forecast ensemble member are

$$\mathbf{y}_i = \mathbf{H}\mathbf{p}_i^f. \quad (28)$$

Given the ensemble of neural-network parameters $\{\boldsymbol{\theta}_i^f\}_{i=1}^{30}$, the corresponding predicted observations $\{\mathbf{y}_i\}_{i=1}^{30}$, and the synthetic observation \mathbf{d} , the neural EnKF update is performed according to Eq. (9), after which the physical-space analysis ensemble is reconstructed following Eq. (10).

The neural EnKF also demonstrates robust DA performance in the 2D blast wave case, as illustrated in Figs. 13–15, confirming its applicability to higher-dimensional shock-laden flows. Specifically, Fig. 13 shows the pressure fields over the full assimilation window, where the ensemble mean and the farthest ensemble member after each neural EnKF update are compared against the reference solution. Here, the farthest ensemble member is defined in the same manner as in the Sod shock tube problem. Prior to the first DA update ($t = 4 \times 10^{-3}$), the forecast state exhibits noticeable uncertainty in the blast location and radius, leading to a smeared pressure discontinuity in the ensemble mean and a pronounced structural deviation in the farthest member. As a realization located at the edge of the ensemble distribution, this member provides a stringent test of the neural EnKF update. Following the first update, its pressure field undergoes substantial structural realignment, with the blast location and radius rapidly adjusted toward the reference solution, and the ensemble mean exhibits a similarly sharp correction. As DA proceeds, subsequent updates further refine both the ensemble mean and the farthest member while preserving the discontinuity structure. Through the final time, the pressure fields remain closely aligned with the reference evolution, without introducing spurious oscillations.

Similarly, the velocity (x-component) and density fields, shown in Figs. 14 and 15, demonstrate effective structural correction after the first neural EnKF update. Although the ensemble mean remains slightly more diffused than the reference solution immediately after this initial update, the primary uncertainty in the discontinuity structure is substantially reduced. By the third DA step, both the ensemble mean and the farthest ensemble member become nearly indistinguishable from the reference solution. Subsequent updates provide only minor refinements while preserving the recovered discontinuity structure.

To further quantify the DA performance, Fig. 16 presents the evolution of RMSE and ensemble spread over successive DA cycles. Consistent with the field-level comparisons above, the pressure field exhibits a sharp reduction in both RMSE and ensemble spread after the first DA cycle, followed by sustained low-error and low-uncertainty levels with a continued gradual decrease. The velocity field displays a comparable trend. The density field follows a similar overall pattern but with a more gradual reduction in the estimation error. Owing to the substantially smaller initial density contrast between the interior and exterior of the blast compared with the pressure jump, the density RMSE is already low prior to the first DA update and remains consistently low throughout the assimilation window. The relatively weak sensitivity of density to pressure observations may also contribute to this behavior.

5. Conclusion

This study considers DA for compressible flows with shocks and other types of discontinuities. We show that the EnKF fails in this problem due to a multimodal forecast distribution induced by uncertain shock locations. Such multimodality fundamentally violates the Gaussian assumptions underlying the EnKF update, leading to spurious oscillations and nonphysical artifacts.

To address this limitation, we introduce the neural EnKF framework, which reformulates the assimilation problem by performing updates in neural network parameter space rather than in physical state space. While representing individual physical states with neural networks is conceptually straightforward, the central difficulty lies in constructing an ensemble in neural space whose statistical structure is suitable for stable EnKF updates. We resolve this issue through a nearest-neighbor chain training/transfer learning strategy. By ordering the training sequence according to physical-space similarity, the method promotes a consistent relationship between variations in neural-network parameters and variations in the underlying physical states. This alignment yields a structured neural-space ensemble that mitigates the adverse effects of multimodality observed in physical space and facilitates stable EnKF updates. Numerical experiments across multiple shock-dominated test cases demonstrate that the neural EnKF achieves robust, structure-preserving DA performance while substantially reducing spurious oscillations.

Several important directions warrant further investigation. First, although the chain-based training strategy is empirically effective, a rigorous theoretical characterization of the induced ensemble geometry in the neural space remains to be developed. Second, extending the framework to large-scale 3D shock-laden flows will require further advances in computational and training efficiency, improved ensemble management, and scalable DA strategies such as localization [47–49]. Third, the method relies on identifying a nonlinear transformation to a function representation that captures the smoothly varying features of the ensemble members and performing the analysis step in this representation; we have achieved this here with neural networks, but other approaches are possible.

CRedit authorship contribution statement

Xu-Hui Zhou: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization; **Lorenzo Beronilla:** Writing – review & editing, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization; **Michael K. Sleeman:** Writing – review & editing; **Hangchuan Hu:** Writing –

review & editing; **Matthias Morzfeld**: Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization; **Andrew M. Stuart**: Writing – review & editing, Supervision, Project administration, Funding acquisition, Conceptualization; **Tamer A. Zaki**: Writing – review & editing, Supervision, Project administration, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the AFOSR/AFRL Center of Excellence in Assimilation of Flow Features in Compressible Reacting Flows (Grant No. FA9550-25-1-0011). Matthias Morzfeld is supported by the U.S. ONR Grant N000142512298. The authors gratefully acknowledge Prof. Kevin Wang, Dr. Xuning Zhao, and Dr. Wentao Ma for their assistance with the M2C solver. The authors also thank Rahul Jain, Ruben Fernandez, and Prof. Venkat Raman for valuable discussions on the application context related to rotating detonation engines. This work used computational resources provided by the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program through allocation PHY250246, and was carried out on Anvil at Purdue University.

Data availability

Data will be made available on request.

Appendix A. Neural network architectures and training details

A.1. Neural network architectures

Table A.1 summarizes the neural network architectures used in the numerical experiments. In all cases, each ensemble member is represented by a fully connected network that maps spatial coordinates to the corresponding flow variables. ReLU activations are employed in the hidden layers, while the final layer uses no activation for velocity components and a softplus activation for density and pressure to enforce positivity.

The selected architectures were chosen to provide sufficient expressive capacity to accurately represent the flow states encountered throughout the DA process. In practice, this capability can be assessed prior to the DA experiments using representative snapshots sampled from the forecast trajectories of a small number of ensemble members. In the 1D experiments (inviscid Burgers' equation and the shock tube case), the networks are over-parameterized relative to the physical state dimension, a deliberate choice to ensure accurate representation of sharp structures while testing stability in high-dimensional weight space. In contrast, for the 2D blast-wave problem, the number of trainable parameters is already smaller than the corresponding physical state dimension. In practical 3D applications, where the discretized state dimension grows rapidly with spatial resolution, the parameter dimension need not scale proportionally, and we can therefore expect even greater gains in parameter efficiency.

Table A.1
Neural network architectures used in the numerical experiments.

	Burgers' equation	Sod shock tube	Blast wave
Input variables	x	x	x, y
Output variables	u	ρ, u, p	ρ, u, v, p
# input neurons	1	1	2
# hidden layers	4	4	6
# neurons per hidden layer	50	64	100
# output neurons	1	3	4
Hidden-layer activation	ReLU	ReLU	ReLU
Last-layer activation	–	Softplus (ρ, p only)	Softplus (ρ, p only)
# trainable parameters	7,801	12,803	51,204
Physical state dimension	400 (= 1×400)	1,200 (= 3×400)	65,536 (= 4×128^2)

We also experimented with two alternative architecture choices. First, we tested deeper and wider variants of the currently selected architectures and observed similar DA performance for the Sod shock tube and blast wave cases. Second, we explored representing different flow variables using separate neural networks for each ensemble member. For example, in the Sod shock tube case, separate networks were used to represent density, velocity, and pressure individually. However, this led to less stable DA updates because the analysis could not reliably preserve shared structures across variables, such as maintaining consistent shock locations. Consequently, the final framework employs a single neural network to jointly represent all flow variables for each ensemble member.

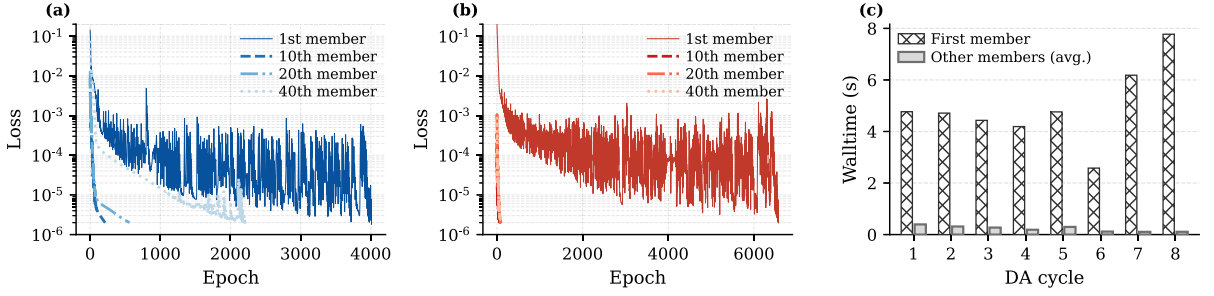


Fig. A.1. Training convergence histories and wall-clock times for the Sod shock tube case: (a,b) training loss histories for selected ensemble members during the first and final DA cycles, respectively; and (c) training wall-clock times for each DA cycle, comparing the first ensemble member in the nearest-neighbor chain with the average over the remaining 39 ensemble members and demonstrating the significant reduction in training cost enabled by warm-start initialization in the chain-based training strategy. Here, the first ensemble member in the chain is trained using a learning rate of 10^{-2} , while the remaining ensemble members use a learning rate of 10^{-3} . All reported wall-clock times correspond to single-core CPU execution.

A.2. Training details

For the forecast ensemble member \mathbf{z}_i^f , the corresponding NN parameters θ_i^f are obtained by minimizing the reconstruction loss

$$\mathcal{L}(\theta_i^f) = \frac{1}{n_x} \sum_{j=1}^{n_x} \left\| F_{\text{NN}}(\theta_i^f; \mathbf{x}_j) - \mathbf{z}_i^f(\mathbf{x}_j) \right\|_2^2, \quad (\text{A.1})$$

where n_x denotes the number of spatial grid points in the computational domain and $\|\cdot\|_2$ denotes the Euclidean norm. The optimization is performed using the Adam optimizer [50], and training is terminated once the reconstruction loss falls below a prescribed threshold, typically of order 10^{-6} .

When applying this training procedure to the full forecast ensemble, we use the nearest-neighbor chain training strategy (see main text) to promote coherence between neural representations across ensemble members. This strategy also significantly accelerates NN training convergence through warm-start initialization, as illustrated in Fig. A.1. Specifically, for the Sod shock tube problem, the convergence histories in panels (a) and (b) show that ensemble members later in the chain converge substantially faster than the first ensemble member trained from random initialization. This acceleration becomes even more pronounced in panel (b) during the final DA cycle, where later ensemble members converge within substantially fewer training epochs. This behavior is expected because the ensemble spread typically decreases as DA progresses, leading to greater similarity between neighboring ensemble members in the chain. Panel (c) further demonstrates the resulting reduction in NN training wall-clock times across DA cycles. Similar training behavior is also observed for the other test cases considered in this work.

Appendix B. EnKF for the shock tube case

Fig. B.1 demonstrates that the analysis ensemble produced by the EnKF exhibits pronounced oscillations across the domain in the shock tube problem. These oscillations severely distort the solution structure, leading to nonphysical states with negative density and pressure and ultimately causing the forward solver to fail in the subsequent DA cycle.

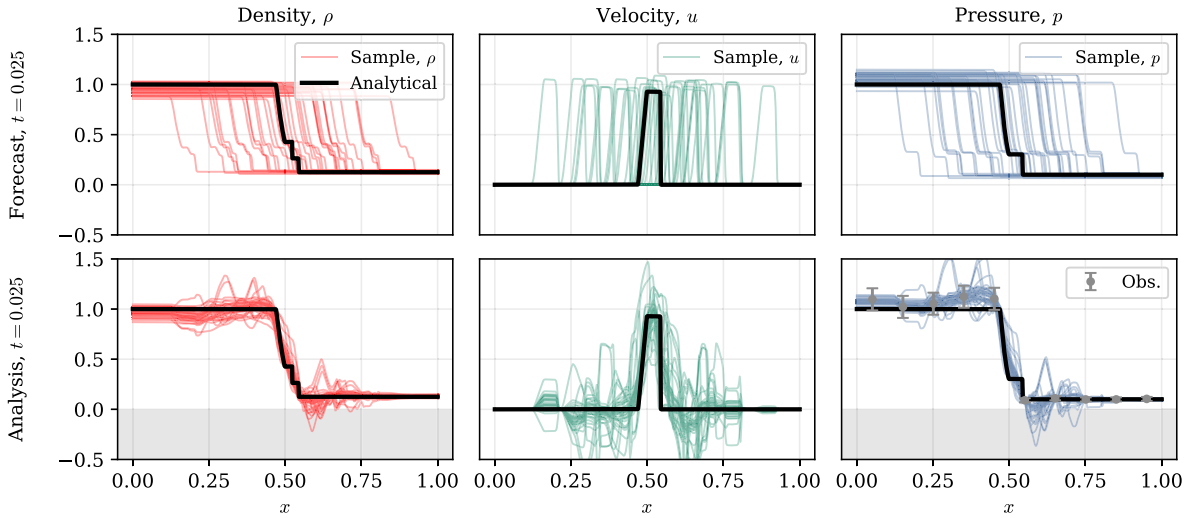


Fig. B.1. EnKF performance for the shock tube problem at the first DA step ($t = 0.025$), showing strong oscillations and the emergence of nonphysical states. The shaded regions indicate the nonphysical range ($\rho < 0$ and $p < 0$).

Appendix C. Pressure-density sensitivity in the shock tube case

To clarify why the density field recovers more slowly under pressure-only observations in the shock tube problem, we examine the forward propagation of uncertainty in the initial density. Here, uncertainty is introduced in the left-state density ρ_L , while all other initial-condition parameters, including p_L, ρ_R, p_R , and the diaphragm location x_d , are fixed at their reference values (see Eq. (21)). This controlled setup isolates the effect of density perturbations and allows us to assess how uncertainty in ρ_L propagates through the flow variables in the absence of DA.

As shown in Fig. C.1, the ensemble exhibits noticeable variability in the density field at $t = 0$, confined to the left state where the perturbation is introduced. After evolution to $t = 0.2$, the density profiles continue to display visible spread, whereas the corresponding pressure profiles remain tightly clustered across the ensemble. To quantify this uncertainty propagation, the ensemble spread of the pressure field at $t = 0.2$ is approximately 0.01, compared with the imposed standard deviation of 0.1 for the injected density perturbation. This suggests that perturbations in ρ_L are only weakly transmitted to the pressure field over the time interval considered.

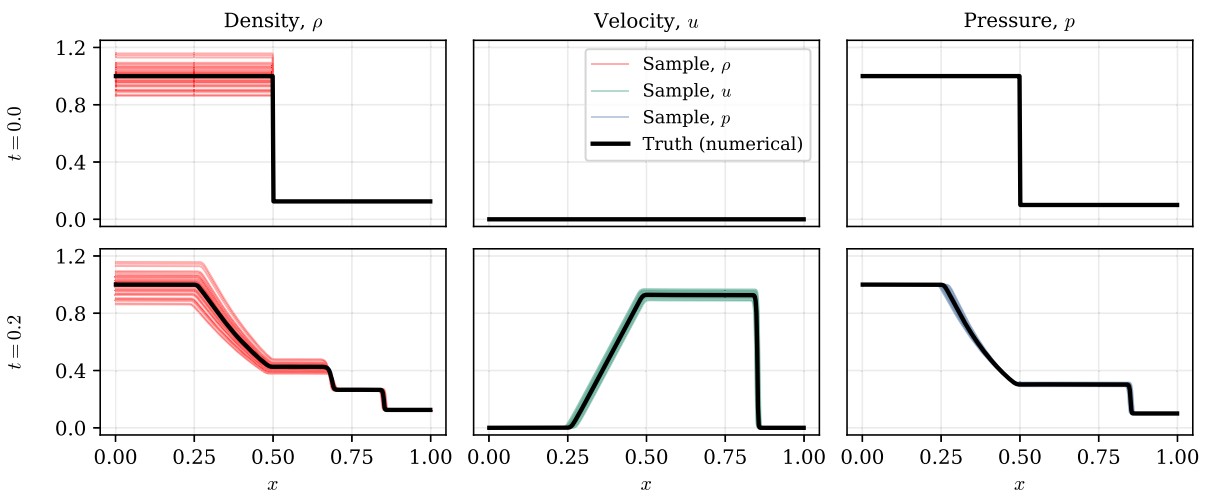


Fig. C.1. Forward propagation of uncertainty in the left-state density ρ_L for the shock tube problem, illustrating its weak influence on the pressure field. The ensemble is evolved from $t = 0.0$ to 0.2 without DA. Columns show density ρ , velocity u , and pressure p ; rows correspond to the initial and final times.

Appendix D. Parametric study for the shock tube problem

To assess the robustness of the neural EnKF under different DA configurations, we conduct a parametric study for the shock tube problem. Specifically, we examine the sensitivity of the DA performance to variations in observation noise level, number of observation locations, and observation interval, while keeping all other numerical and DA settings fixed.

The configurations considered are summarized in Table D.1. The baseline corresponds to the setup used in the main text. Experiments A1–A2 vary the observation noise level. Experiments B1–B2 alter the number of observation locations, equivalently changing the observation spacing Δx_{obs} . Experiments C1–C2 vary the observation interval Δt_{obs} , leading to different numbers of DA cycles over the entire simulation window ($T = 0.2$).

Table D.1

DA configurations used in the parametric study of the shock tube problem. The baseline corresponds to the setup used in the main text. Experiments A1–A2 vary the observation noise level; B1–B2 vary the number of observation locations, thereby changing the observation spacing Δx_{obs} ; and C1–C2 vary the observation interval Δt_{obs} , leading to different numbers of DA cycles. Parameters that differ from the baseline are highlighted in **bold**.

	Obs. noise	# Obs. loc.	Δx_{obs}	# DA cycles	Δt_{obs}
Baseline	5%	10	0.1	8	0.025
Exp. A1	1%	10	0.1	8	0.025
Exp. A2	10%	10	0.1	8	0.025
Exp. B1	5%	5	0.2	8	0.025
Exp. B2	5%	20	0.05	8	0.025
Exp. C1	5%	10	0.1	4	0.05
Exp. C2	5%	10	0.1	16	0.0125

The final-time analysis ensembles obtained under different DA configurations are shown in Fig. D.1. Overall, the neural EnKF exhibits robust performance across all tested settings, consistently recovering the dominant flow structures. The sharpness of discontinuities and the ensemble spread, however, vary systematically with the strength of the observational constraints.

Specifically, stronger observational constraints—reduced observation noise (Exp. A1), a larger number of observation locations (Exp. B2), or more frequent assimilation (Exp. C2)—lead to smaller ensemble spread and sharper reconstruction of discontinuities, particularly the shock. In these cases, the analysis ensembles are more tightly concentrated around the analytical solution, indicating improved estimation accuracy and reduced uncertainty. Conversely, weaker constraints—higher noise levels (Exp. A2), fewer observation locations (Exp. B1), or less frequent assimilation (Exp. C1)—result in larger ensemble spread and less sharply localized transitions near discontinuities. Among these factors, using fewer observation locations (Exp. B1) produces the most noticeable degradation, highlighting the difficulty of accurately localizing shocks when spatial information is insufficient.

Across all configurations, pressure remains the most robustly recovered variable due to direct observation. Velocity follows closely owing to its strong dynamical coupling with pressure, whereas density exhibits greater sensitivity to the DA configuration, reflecting its weaker constraint by pressure observations alone, as discussed in Appendix C.

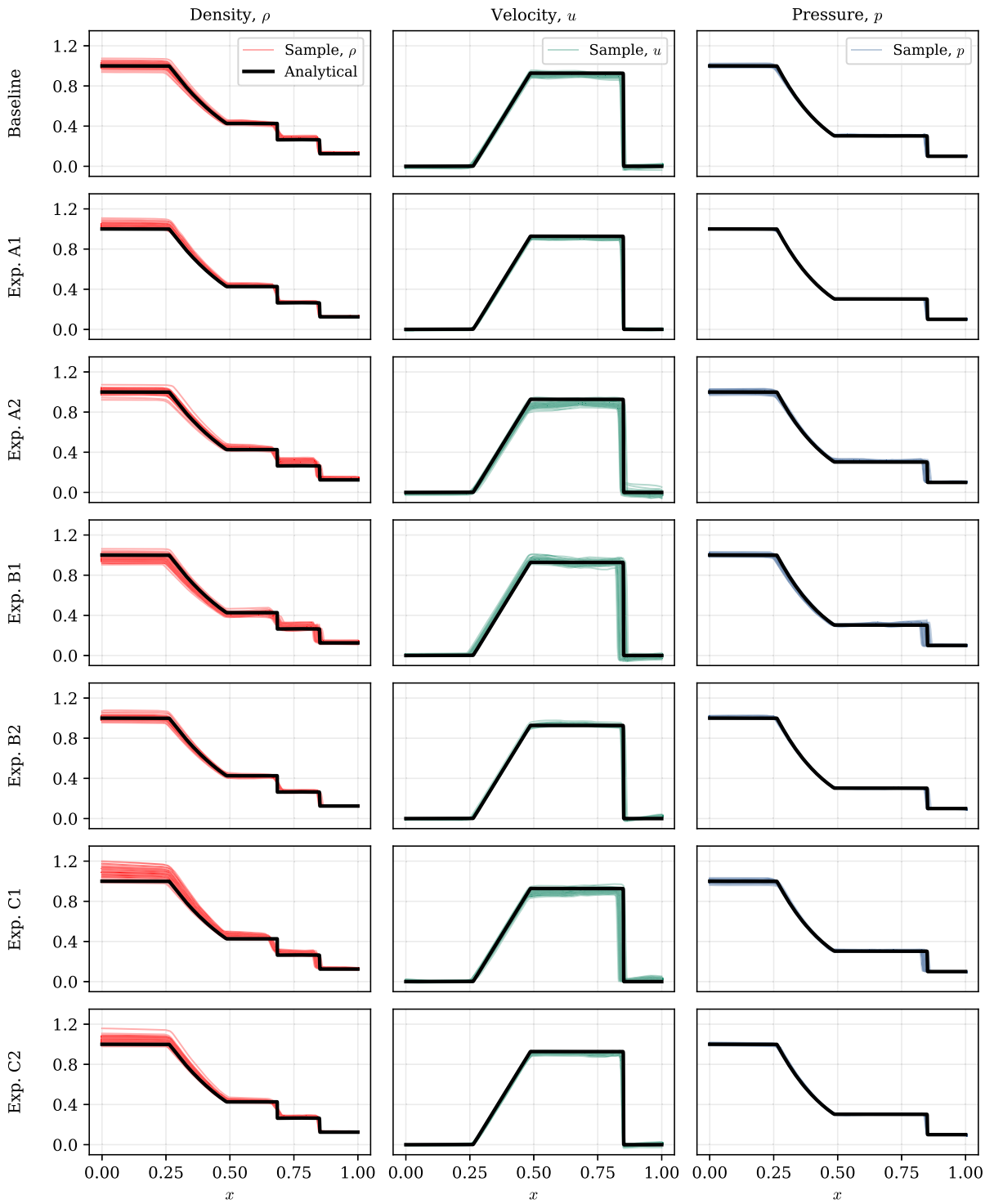


Fig. D.1. Final-time ($t = 0.2$) analysis ensembles for the shock tube problem under the DA configurations summarized in Table D.1. Columns show density ρ , velocity u , and pressure p . Rows correspond to the baseline case used in the main text; Experiments A1–A2 (low and high observation noise); Experiments B1–B2 (reduced and increased numbers of observation locations); and Experiments C1–C2 (lower and higher observation frequency).

References

- [1] V. Raman, S. Prakash, M. Gamba, Nonidealities in rotating detonation engines, *Annu. Rev. Fluid Mech.* 55 (2023) 639–674.
- [2] J.D. Anderson, *Hypersonic and High Temperature Gas Dynamics*, AIAA, 1989.
- [3] M. Schuabb, L. Duan, K.M. Casper, R.M. Wagnild, M.M. Choudhari, P. Paredes, Hypersonic boundary-layer transition over a circular cone in a Mach 8 digital wind tunnel, *J. Fluid Mech.* 1017 (2025) A33.
- [4] S. Cao, G. Wang, O. Coutier-Delgosha, K. Wang, Shock-induced bubble collapse near solid materials: effect of acoustic impedance, *J. Fluid Mech.* 907 (2021) A17.
- [5] X. Zhao, W. Ma, J. Chen, G. Xiang, P. Zhong, K. Wang, Vapour bubbles produced by long-pulsed laser: a race between advection and phase transition, *J. Fluid Mech.* 999 (2024) A103.
- [6] S. Cao, Y. Zhang, D. Liao, P. Zhong, K.G. Wang, Shock-induced damage and dynamic fracture in cylindrical bodies submerged in liquid, *Int. J. Solids Struct.* 169 (2019) 55–71.
- [7] W. Ma, X. Zhao, C. Gilbert, K. Wang, Computational analysis of bubble–structure interactions in near-field underwater explosion, *Int. J. Solids Struct.* 242 (2022) 111527.
- [8] A. Narkhede, S. Islam, X. Sun, K. Wang, Fluid–structure coupled simulation framework for lightweight explosion containment structures under large deformations, *Int. J. Impact Eng.* 199 (2025) 105238.
- [9] G. Evensen, *Data Assimilation: the Ensemble Kalman Filter*, Springer, 2009.
- [10] M. Asch, M. Bocquet, M. Nodet, *Data Assimilation: Methods, Algorithms, and Applications*, SIAM, 2016.
- [11] G. Evensen, F.C. Vossepoel, P.J. Van Leeuwen, *Data Assimilation Fundamentals: A Unified Formulation of the State and Parameter Estimation Problem*, Springer Nature, 2022.
- [12] E. Kalnay, *Atmospheric Modeling, Data Assimilation and Predictability*, Cambridge university press, 2003.
- [13] A. Carrassi, M. Bocquet, L. Bertino, G. Evensen, Data assimilation in the geosciences: an overview of methods, issues, and perspectives, *Wiley Interdiscip. Rev. Clim. Change* 9 (2018) e535.
- [14] X.-L. Zhang, H. Xiao, X. Luo, G. He, Ensemble Kalman method for learning turbulence models from indirect observation data, *J. Fluid Mech.* 949 (2022) A26.
- [15] X.-H. Zhou, H. Wang, J. McClure, C. Chen, H. Xiao, Inference of relative permeability curves in reservoir rocks with ensemble Kalman method, *Eur. Phys. J. E* 46 (2023) 44.
- [16] M.I. Zafar, X.-H. Zhou, C.J. Roy, D. Stelter, H. Xiao, Data-driven turbulence modeling approach for cold-wall hypersonic boundary layers, *J. Thermophys. Heat Transf.* 40 (2026) 379–395.
- [17] Z.-R. Liu, H.-C. Wang, Z.-L. Zhao, H. Xiao, Towards a unified data-driven turbulence model through multi-objective learning, (2025). [arXiv preprint arXiv:2509.17189](https://arxiv.org/abs/2509.17189)
- [18] M. Wang, T.A. Zaki, State estimation in turbulent channel flow from limited observations, *J. Fluid Mech.* 917 (2021) A9.
- [19] T.A. Zaki, Turbulence from an observer perspective, *Annu. Rev. Fluid Mech.* 57 (2025) 311–334.
- [20] D.A. Buchta, T.A. Zaki, Observation-infused simulations of high-speed boundary-layer transition, *J. Fluid Mech.* 916 (2021) A44.
- [21] D.A. Buchta, S.J. Laurence, T.A. Zaki, Assimilation of wall-pressure measurements in high-speed flow over a cone, *J. Fluid Mech.* 947 (2022) R2.
- [22] P. Morra, C. Meneveau, T.A. Zaki, ML for fast assimilation of wall-pressure measurements from hypersonic flow over a cone, *Sci. Rep.* 14 (2024) 12853.
- [23] Q. Wang, T.A. Zaki, Domain of dependence for wall-pressure measurements in high-speed boundary layers, *J. Fluid Mech.* 1009 (2025) A67.
- [24] E.J. West, J.F. MacArt, R. Munipalli, Variational data assimilation in shock tube flows, in: *AIAA SCITECH 2025 Forum*, 2025, p. 1166.
- [25] T. Houba, A. Edoh, R. Munipalli, M.E. Harvazinski, Sequential data assimilation in flows with shocks, in: *AIAA SCITECH 2024 Forum*, 2024, p. 0587.
- [26] A. Edoh, T. Houba, R. Munipalli, M.E. Harvazinski, Sequential ensemble Kalman filtering of compressible flows with shocks: enforcing positivity, in: *AIAA SCITECH 2025 Forum*, 2025, p. 0918.
- [27] A.K. Edoh, E.J. West, T. Houba, R. Munipalli, M.E. Harvazinski, W. Kang, Data assimilation of compressible flows with discontinuities: evaluating algorithms on Sod’s shock tube, *Int. J. Numer. Methods Fluids* 98 (2026) 644–672.
- [28] J.J. Hansen, D. Brouzet, M. Ihme, A normal-score ensemble Kalman filter for 1D shock waves, in: *AIAA SCITECH 2024 Forum*, 2024, p. 1022.
- [29] H. Zhou, J.J. Gomez-Hernandez, H.-J.H. Franssen, L. Li, An approach to handling non-gaussianity of parameters and state variables in ensemble Kalman filtering, *Adv. Water Resour.* 34 (2011) 844–864.
- [30] J.J. Hansen, D. Brouzet, M. Ihme, Data assimilation of state-indirect observations for constraining detonation wave dynamics, *Proc. Combust. Inst.* 41 (2025) 105812.
- [31] T. Li, A. Gelb, Y. Lee, A structurally informed data assimilation approach for nonlinear partial differential equations, *J. Comput. Phys.* 519 (2024) 113396.
- [32] T. Li, A. Gelb, Y. Lee, Structurally informed data assimilation in two dimensions, (2025). [arXiv preprint arXiv:2510.06369](https://arxiv.org/abs/2510.06369)
- [33] A.N. Subrahmanya, A. Sandu, Feature preserving data assimilation via feature alignment, *Comput. Methods Appl. Mech. Eng.* 447 (2025) 118345.
- [34] A. Srivastava, W. Kang, D.M. Tartakovsky, Feature-informed data assimilation, *J. Comput. Phys.* 494 (2023) 112499.
- [35] G. Burgers, P.J. van Leeuwen, G. Evensen, Analysis scheme in the ensemble Kalman filter, *Mon. Weather Rev.* 126 (1998) 1719–1724.
- [36] M.A. Iglesias, K.J. Law, A.M. Stuart, Ensemble Kalman methods for inverse problems, *Inverse Probl.* 29 (2013) 045001.
- [37] M. Morzfeld, D. Hodyss, Gaussian approximations in filters and smoothers for data assimilation, *Tellus A Dyn. Meteorol. Oceanogr.* 71 (2019) 1600344.
- [38] M. Peyron, A. Fillion, S. Gürol, V. Marchais, S. Gratton, P. Boudier, G. Goret, Latent space data assimilation by using deep learning, *Q. J. R. Meteorol. Soc.* 147 (2021) 3759–3777.
- [39] H. Mousavi, A. Jones, J. Eldredge, Sequential estimation of disturbed aerodynamic flows from sparse measurements via a reduced latent space, (2025). [arXiv preprint arXiv:2509.03795](https://arxiv.org/abs/2509.03795)
- [40] X.T. Tong, Y. Wang, L. Yan, Latent autoencoder ensemble Kalman filter for data assimilation, (2026). [arXiv preprint arXiv:2603.06752](https://arxiv.org/abs/2603.06752)
- [41] E. Bach, R. Baptista, E. Calvello, B. Chen, A. Stuart, Learning enhanced ensemble filters, *J. Comput. Phys.* (2025) 114550.
- [42] X.-H. Zhou, Z.-R. Liu, H. Xiao, BI-EqNO: Generalized approximate Bayesian inference with an equivariant neural operator framework, 2024, [arXiv preprint arXiv:2410.16420](https://arxiv.org/abs/2410.16420)
- [43] D.I. Ketcheson, K.T. Mandli, A.J. Ahmadi, A. Alghamdi, M.Q. de Luna, M. Parsani, M.G. Knepley, M. Emmett, PyClaw: accessible, extensible, scalable tools for wave propagation problems, *SIAM J. Sci. Comput.* 34 (2012) C210–C231.
- [44] G.A. Sod, A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws, *J. Comput. Phys.* 27 (1978) 1–31.
- [45] X. Zhao, W. Ma, S. Islam, A. Narkhede, K. Wang, M2C: an open-source software for multiphysics simulation of compressible multi-material flows and fluid–structure interactions, *Comput. Phys. Commun.* (2026) 110023.
- [46] K. Wang, M2C: multiphysics modeling and computation, 2026, (<https://github.com/kevinwgy/m2c>). Accessed: 2026-01-25.
- [47] E. Ott, B.R. Hunt, I. Szunyogh, A.V. Zimin, E.J. Kostelich, M. Corazza, E. Kalnay, D. Patil, J.A. Yorke, A local ensemble Kalman filter for atmospheric data assimilation, *Tellus A Dyn. Meteorol. Oceanogr.* 56 (2004) 415–428.
- [48] B.R. Hunt, E.J. Kostelich, I. Szunyogh, Efficient data assimilation for spatiotemporal chaos: a local ensemble transform Kalman filter, *Phys. D Nonlinear Phenom.* 230 (2007) 112–126.
- [49] X.T. Tong, M. Morzfeld, Localized ensemble Kalman inversion, *Inverse Probl.* 39 (2023) 064002.
- [50] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, (2014). [arXiv preprint arXiv:1412.6980](https://arxiv.org/abs/1412.6980)