Estimating density, velocity, and pressure fields in supersonic flows using physics-informed BOS

Joseph P. Molnar[1] · Lakshmi Venkatakrishnan[2] · Bryan E. Schmidt[3] · Timothy A. Sipkens[4] · Samuel J. Grauer[1]

Abstract

We report a new workflow for background-oriented schlieren (BOS), termed "physics-informed BOS," to extract density, velocity, and pressure fields from a pair of reference and distorted images. Our method uses a physics-informed neural network (PINN) to produce flow fields that simultaneously satisfy the measurement data and governing equations. For the high-speed, approximately inviscid flows of interest in this work, we specify a physics loss based on the Euler and irrotationality equations. BOS is a quantitative fluid visualization technique that is commonly used to characterize compressible flow. Images of a background pattern, positioned behind the measurement volume, are processed with computer vision and tomography algorithms to determine the density field. Crucially, BOS features a series of ill-posed inverse problems that require supplemental information (i.e., in addition to the images) to accurately reconstruct the flow. Current methods for BOS rely upon interpolation of the images or a penalty term to promote a globally- or piecewise-smooth solution. However, these algorithms are invariably incompatible with the flow physics, leading to errors in the density field. Physics-informed BOS directly reconstructs all the flow fields using a PINN that includes the BOS measurement model and governing equations. This procedure improves the accuracy of density estimates and also yields velocity and pressure data, which were not previously available. We demonstrate our approach by reconstructing synthetic data that corresponds to analytical and numerical phantoms as well as a single pair of experimental measurements. Our physics-informed reconstructions are significantly more accurate than conventional BOS estimates. Furthermore, to the best of our knowledge, this work represents the first use of a PINN to reconstruct a supersonic flow from experimental data of any kind.

1 Introduction

Supersonic and hypersonic flows feature complex phenomena such as shock waves, shock wave–boundary layer interactions, and eddy shocklets, which must be considered in the design of next-generation aircraft and re-entry vehicles, projectiles, and combustion processes (Dolvin 2008). Computational fluid dynamics (CFD) simulations play a vital role in engineering, but many vehicles are being designed to operate outside the parameter space in which engineering experience or CFD provide reliable means of analysis. Therefore, in order to support the design process, experimental measurements are needed to characterize and understand high-speed flow phenomena as well as to develop and validate numerical models.

Background-oriented schlieren (BOS) is a non-intrusive, quantitative flow visualization tool that can be applied to high-speed systems (Raffel 2015). BOS has been widely used to characterize shock-laden flows (Venkatakrishnan and Meier 2004; Sommersel et al. 2008; Yamagishi et al. 2021; Gomez et al. 2022), visualize combustion processes (Grauer et al. 2018; Liu et al. 2022), and estimate velocity fields (Tokgoz et al. 2012), amongst other applications (Raffel 2015). The technique provides line-of-sight (LoS) integrated information about the flow via the apparent motion of a background pattern. Images of the pattern are distorted by refraction through the fluid, which is caused by density gradients along lines-of-sight from a background plate to the camera. Differences
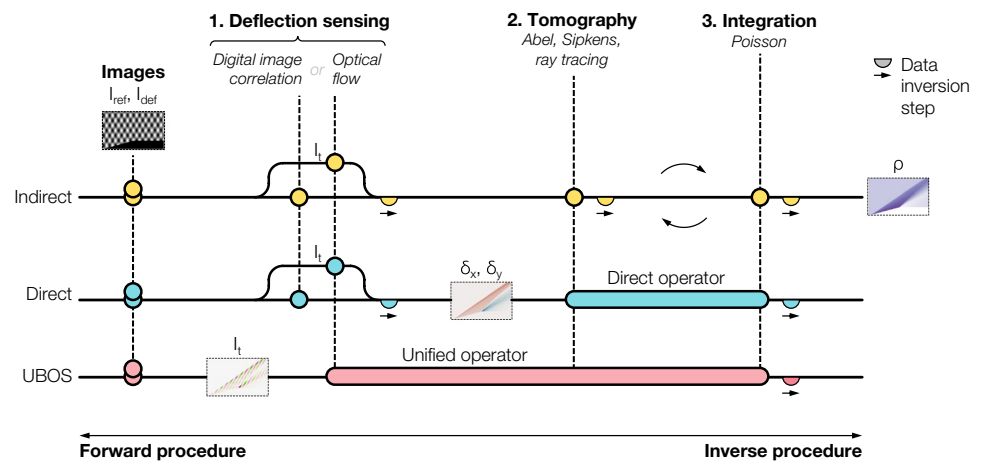
✉ Samuel J. Grauer
   sgrauer@psu.edu

[1] Department of Mechanical Engineering, Pennsylvania State University, University Park, PA, USA

[2] CSIR-National Aerospace Laboratories, Bangalore, India

[3] Department of Mechanical and Aerospace Engineering, Case Western Reserve University, Cleveland, OH, USA

[4] Metrology Research Centre, National Research Council Canada, Ottawa, Canada

Fig. 1 Graphical overview of quantitative BOS, in which image data is converted to a density field through deflection sensing, tomography, and a Poisson solver. The order of tomographic reconstruction and Poisson integration may be switched in indirect BOS or combined in direct BOS. In unified BOS (UBOS in the figure), all three procedures are performed by inverting a single operator
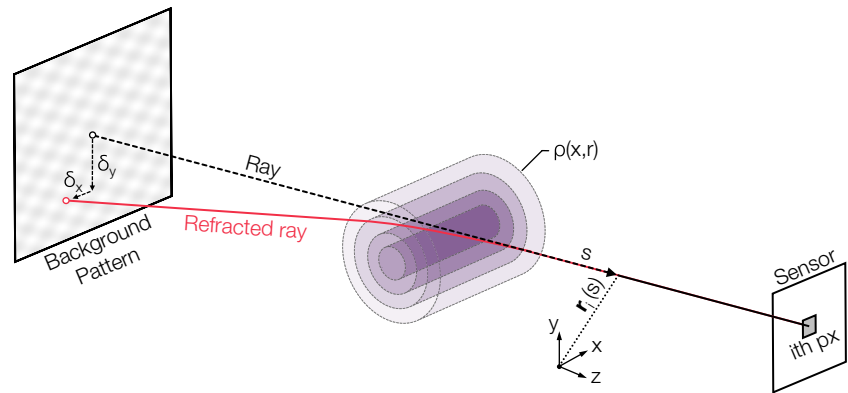

between a reference image, recorded before introducing the flow, and a distorted image from the experiment can thus be processed with a computer vision algorithm to render a "synthetic schlieren" image (Dalziel et al. 2000), which may reveal key fluid structures. Further, BOS data can be tomographically reconstructed to obtain a quantitative estimate of the density field. Unfortunately, BOS features a series of ill-posed inverse problems that admit an infinite set of solutions. Supplemental (or "prior") information is therefore needed to generate a unique, physical solution. Adding prior information to solve an inverse problem is termed "regularization," and the aim of this work is to establish a physics-based approach to regularization in BOS.

There are three inverse problems in BOS, depicted schematically in Fig. 1, which are typically arranged in the following sequence. First, the image pair is converted to a set of deflection vectors through a procedure called deflection sensing. Here, a "deflection" is the 2D displacement of a point from the reference image to the same point in the distorted image. Deflections are usually resolved at the centroid of each pixel or interrogation window, although the deflection field is continuous, in principle. Second, individual components of the deflections are tomographically reconstructed, which yields gradients of the refractive index field. BOS is inherently sensitive to the refractive index field, but the density field is directly accessible for a fluid of constant composition via the Gladstone–Dale relation. Third, these reconstructions are incorporated into a Poisson equation that must be solved to recover the refractive index or density field, per se. Steps two and three can be reversed (Rajendran et al. 2020) or combined (Nicolas et al. 2016), and it is possible to conduct all three steps at once via "unified BOS" (Grauer and Steinberg 2020). Each step that is performed in isolation requires regularization, and it is advisable to combine steps, where possible, to reduce the reliance on prior information.

Deflection sensing is typically conducted using a cross-correlation (Venkatakrishnan and Meier 2004; Castner 2012; Geerts and Yu 2017), dot tracking (Rajendran et al. 2019), or optical flow (OF) (Atcheson et al. 2009; Heineck et al. 2021; Schmidt and Woike 2021) algorithm. Cross-correlation methods identify the displacement of a multi-pixel window from the reference image to the distorted image. The use of multi-pixel windows, typically 8×8 px or larger, reduces the resolution of the deflection field (Raffel et al. 1998; Atcheson et al. 2009; Schmidt and Woike 2021), which effectively amounts to a smoothing operation. Dot tracking algorithms attempt to determine the displacement of individual features on a pseudo-PIV background, which may be enhanced with a small-window, single-dot correlation step (Rajendran et al. 2019), but the resolution of this approach is limited by the density of dots, which is necessarily lower than the resolution of the sensor. Meanwhile, OF algorithms produce displacement fields with the same resolution as the original images and have been shown to outperform correlation and tracking algorithms in both resolution and accuracy (Schmidt and Woike 2021). However, the OF problem features one equation and two unknowns per pixel, so an optimization criterion is needed to close the problem. Closure may be provided by assuming a locally-uniform solution [Lucas–Kanade OF (Lucas and Kanade 1981)], global smoothness [Horn–Schunck OF (Horn and Schunck 1981)], or a sparse representation in wavelet space (Schmidt and Woike 2021). Lucas–Kanade OF is substantially similar to cross-correlation and does not yield much better estimates (Atcheson et al. 2009). The latter two techniques can generate accurate deflections, but it is exceedingly difficult to develop a physics-based constraint for OF in BOS due to the LoS-integrated nature of the deflection field (Schmidt and Woike 2021).

Similar to OF, tomographic reconstruction inherently requires regularization to obtain a unique, physical solution and to counteract errors in the deflection estimates (Daun

Fig. 2 Schematic of a single-camera BOS setup for axisymmetric flows. Rays are refracted by density gradients in the flow, distorting images of the background pattern. Apparent deflections, $\boldsymbol{\delta} = [\delta_x, \delta_y]^T$, are determined with a computer vision algorithm and reconstructed to estimate ρ

et al. 2016). Many examples of BOS feature an axisymmetric flow (Raffel 2015) such that the density field can be recovered from a single perspective via a modified Abel inversion (Kogelschatz and Schneider 1972; Agrawal et al. 1999). However, discrete analytical Abel inversion is unstable due to noise amplification that is inherent to numerical differentiation and a singularity at the line of symmetry (Agrawal et al. 1999). Better performance can be realized by coupling the forward model with an explicit penalty term and solving the resulting system with an optimization technique, often referred to as classical regularization (Daun et al. 2006; Howard et al. 2016). Far and away the most common penalties for tomography are the second-order Tikhonov (Vauhkonen et al. 1998) and total variation (TV) (Kolehmainen et al. 1998) terms, which promote global and piecewise smoothness, respectively. However, both penalties are exclusively minimized by a uniform field, which is incompatible with the density fields of interest in high-speed experimental fluid mechanics. This discrepancy introduces a trade-off between minimizing the measurement residuals and the penalty term. Consequently, both Tikhonov and TV regularization include a parameter that weights the penalty and must be carefully tuned for each experiment. While the use of a penalty term can stabilize tomographic reconstructions and improve their accuracy, errors associated with these penalty-based schemes are pervasive.[1] Regularized solutions tend to be overly-smooth, missing the fine detail present in a flow.

We propose a direct, physics-informed BOS workflow to avoid the regularization errors associated with deflection sensing and tomographic reconstruction and to recover the latent velocity and pressure fields. We utilize a physics-informed neural network (PINN) (Raissi et al. 2019) to represent the flow, which requires data and physics loss terms. Our data loss is based on a unified BOS operator (Grauer and

Steinberg 2020), which directly relates a density field to raw image distortion data, and our physics loss comprises the compressible Euler and irrotationality equations. Previous work by the group of Karniadakis used a PINN to post-process 3D BOS tomography reconstructions of natural convection above an espresso cup (Cai et al. 2021). However, we found that directly embedding the measurement model into a PINN's data loss produces superior reconstructions (Molnar and Grauer 2022), leading to the present formulation.

This paper describes BOS, OF, tomography, and our physics-informed workflow. We apply the technique to reconstruct synthetic data from analytical and numerical phantoms as well as experimental images. Not only does physics-informed BOS yield better estimates of the density field than conventional techniques, it also generates estimates of the velocity and pressure fields. Furthermore, by processing noise-laden experimental LoS measurements with a realistic forward model instead of simulated pointwise data (Mao et al. 2020; Jagtap et al. 2022), this work represents an advance in the application of PINNs to high-speed flows.

2 BOS

Figure 2 depicts a common setup for axisymmetric BOS. A single camera is focused through the fluid to be measured onto a background plate that contains a pattern. Density gradients in the flow give rise to refractive index (speed of light) gradients, which cause wavefronts of light to bend (refract), thereby distorting images of the pattern. These distortions are characterized in terms of 2D displacement vectors, i.e., deflections, which are resolved at each pixel in the image. Deflections are estimated using a computer vision algorithm, and the deflection data may be reconstructed by inverting a BOS measurement model and/or solving a Poisson equation to estimate the unknown refractive index and density fields.

Throughout this work, we consider axisymmetric and planar flows that are aligned with a single background plate,

---

[1] As an example, Tikhonov regularization is characterized by "streaky" artifacts, such as those in Fig. 5 of Wei et al. (2021).

as shown in Fig. 2. We assign the x-axis to the streamwise direction; the x- and y-axes mark the horizontal and vertical directions in the plane of the background plate; and the z-axis is normal to this plane. The camera is pointed directly towards the background and rotated such that the horizontal and vertical image coordinates coincide with the x- and y-axes, respectively, and the background pattern is assumed to be in focus. Continuous and discrete deflection models for BOS are introduced below, followed by an overview of deflection sensing with OF, the unified model used in this work, and axisymmetric reconstruction methods.

2.1 Light propagation through variable index media

Wavefronts of light bend when the speed of light changes throughout a medium, which manifests as visible distortions of the background pattern in BOS. The speed of light in a medium is characterized by its refractive index, n. For gases, this property exhibits a linear dependence upon the material density, ρ, and composition, as described by the Gladstone–Dale equation,

$$n = 1 + G\rho \Rightarrow \nabla n = G\nabla\rho. \tag{1}$$

Here, G is the Gladstone–Dale coefficient (Gardiner et al. 1981), which varies with chemical composition and exhibits a slight wavelength dependence. For measurements of air at visible wavelengths, $G \approx 2.26 \times 10^{-4}$ m^3/kg.

Light propagation is fundamentally governed by Maxwell's equations (Born and Wolf 2013). For a locally homogeneous region free of current and charge, these equations simplify to a wave equation. Assuming that variations in the refractive index field occur over much longer length scales than the wavelength of light, this wave equation simplifies to an eikonal equation, which describes the phase of light waves as a function of n, alone.[2] In this limit, known as geometric optics, the propagation of light can be approximated by "rays" that travel normal to phase fronts of the wave. This phenomena is described by the so-called ray equation (Stam and Languénou 1996),

$$\frac{d}{ds}\left(n\frac{d\mathbf{x}}{ds}\right) = \nabla n, \tag{2}$$

where $\mathbf{x}$ is the position of a massless particle traversing a ray of light and s is a scalar progress variable, which indicates a distance along the ray as illustrated in Fig. 2.

The ray equation can be separated into two ordinary differential equations and integrated along a ray to calculate the deflection of that ray in the background plane, $\boldsymbol{\delta} = [\delta_x, \delta_y]^T$

(Atcheson et al. 2008). This integration follows a curved path, in principle, but the curve is slight within a BOS measurement volume (Goldhahn and Seume 2007). Therefore, the paraxial assumption is invoked and the integral is carried out along the (straight) reference trajectory. For the camera setup shown in Fig. 2, the α-direction deflection, for $\alpha \in \{x, y\}$, in pixel units is given by a path integral along the reference ray,

$$\delta_\alpha = \frac{d\,\psi}{n_0} \int_{\text{ray}} \nabla_\alpha n[\mathbf{r}(s)]\text{d}s = \underbrace{\frac{d\,\psi\,G}{n_0}}_{C_{\text{sys}}} \int_{\text{ray}} \nabla_\alpha \rho[\mathbf{r}(s)]\text{d}s. \tag{3}$$

In this expression, n_0 is the ambient refractive index, d is the distance from the center of the probe volume to the background, ψ is the pixel pitch, and C_{sys} is an overall system constant. The indicator function, $\mathbf{r} : \mathbb{R}^1 \to \mathbb{R}^3$, maps the distance along a ray, s, to the corresponding 3D location, as illustrated above in Fig. 2. Equation (3) presumes that deflections are small and the path length through the domain is short compared to d. Equivalent expressions are derived in Atcheson et al. (2008), Nicolas et al. (2016), Grauer and Steinberg (2020).

2.2 Discrete deflection model

Algebraic tomography requires a discrete approximation to the forward measurement model, which is inverted by a reconstruction algorithm. To start, the field of interest is discretized using the basis $\Phi = \{\varphi_j\}_{j=1}^N$, in which φ_j is the jth basis function out of N such functions. Taking this approach, the density field is approximated as follows:

$$\rho(\mathbf{x}) \approx \sum_j^N \rho_j\,\varphi_j(\mathbf{x}), \tag{4}$$

where ρ_j is a coefficient that scales φ_j and the field is represented by the $N \times 1$ vector $\boldsymbol{\rho} = \{\rho_j\}_{j=1}^N$. Next, the discrete density field is substituted into Eq. (3). Consider the ith ray,

$$\delta_{\alpha,i} \approx C_{\text{sys}} \int_{\text{ray}} \nabla_\alpha \left\{ \sum_{j=1}^N \rho_j\,\varphi_j[\mathbf{r}_i(s)] \right\} \text{d}s \tag{5a}$$

$$= C_{\text{sys}} \sum_{j=1}^N \rho_j \underbrace{\int_{\text{ray}} \nabla_\alpha \varphi_j[\mathbf{r}_i(s)]\text{d}s}_{D_{\alpha,i,j}}, \tag{5b}$$

where $\mathbf{r}_i$ indicates a path along the ith ray. Since the integrals over φ_j do not depend on ρ_j, they can be precomputed to form a deflectometry matrix, $\mathbf{D}_\alpha$, with elements, $D_{\alpha,i,j}$, given by the integration in Eq. (5b). This matrix relates the discrete

---

[2] See Born and Wolf (2013) for the derivation of an eikonal equation from Maxwell's equations.
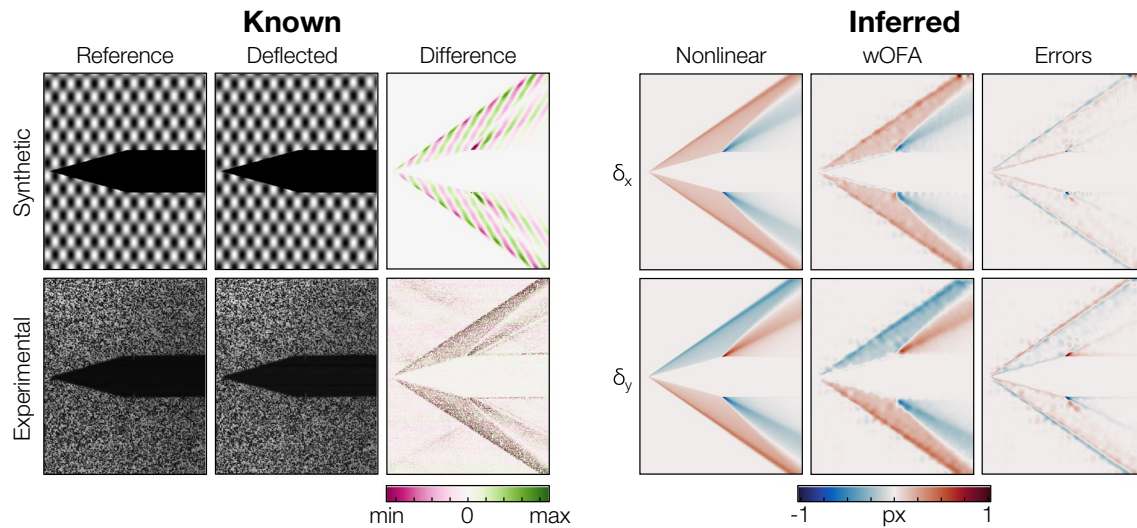
Fig. 3 Synthetic and experimental images of a Mach 2 cone cylinder scenario; image differences are processed with an OF algorithm to estimate the deflection field (left). Exact deflections from the synthetic scenario, determined via nonlinear ray tracing, are compared to estimates from an OF algorithm (right)

density field, ρ, to an $M \times 1$ vector of α-direction deflections, $\boldsymbol{\delta}_\alpha = \{\delta_{\alpha,i}\}_{i=1}^M$, such that $\mathbf{D}_\alpha \in \mathbb{R}^{M \times N}$ and

$$\boldsymbol{\delta}_\alpha = C_{\text{sys}}\, \mathbf{D}_\alpha \boldsymbol{\rho}. \tag{6}$$

For axisymmetric flows, we employ a Sipkens deflectometry operator (Sipkens et al. 2021), which is specified in 1. For planar fields, we invoke the paraxial assumption and directly approximate Eq. (3) by linear ray tracing.

2.3 OF

In BOS, refraction results in the *apparent* 2D motion of a background pattern between a reference image, I_{ref}, and a distorted (or "deflected") image, I_{def}. This corresponds to a classic problem in computer vision known as optical flow, in which the image pair is used to infer a set of deflection vectors (Szeliski 2010). Figure 3 shows synthetic and experimental examples of I_{ref} and I_{def} for a cone cylinder shock scenario as well as the image differences, $\Delta I = I_{\text{def}} - I_{\text{ref}}$. Note that these differences are nearly imperceptible when the images are placed side by side. Nevertheless, the cone shock structure is clearly visible in the plot of ΔI. The image pair is used to infer δ_x and δ_y at each pixel, which collectively form the measurement vectors for Eq. (6), i.e., $\boldsymbol{\delta}_x$ and $\boldsymbol{\delta}_y$.

The transformation of I_{ref} into I_{def} is modeled in terms of a 2D field of deflection vectors, which warp the original intensity distribution (Davies 2004). There are three key assumptions that underlie OF for BOS (Atcheson et al. 2009):

1. Changes in the scene are strictly due to refraction, as opposed to motion or dynamic lighting (shadows, reflections, etc.).
2. The intensity of features is conserved, meaning there is neither emission from the flow nor extinction of light from the background. Stated mathematically,

$$I(x, y, t) = I\big(x + \delta_x, y + \delta_y, t + \delta_t\big), \tag{7}$$

   where δ_t is the time interval between frames.
3. The magnitude of deflections is small.

Given these assumptions, the OF equation can be approximated by a first-order Taylor series expansion of Eq. (7),

$$I(x, y, t) \approx I(x, y, t) + \frac{\partial I}{\partial x}\delta_x + \frac{\partial I}{\partial y}\delta_y + \frac{\partial I}{\partial t}\delta_t + \dots \tag{8}$$

Methods based on this expression, called gradient-based OF, are invalid for large deflections, e.g., in particle-based OF velocimetry, in which case Eq. (7) must be solved with a variation algorithm (Schmidt and Sutton 2019, 2020). However, BOS deflections are generally small enough to satisfy the Taylor series expansion (Goldhahn and Seume 2007). Time plays an arbitrary role in the OF equations in BOS. Therefore, δ_t may be set to unity such that Eq. (8) reduces to

$$I_x\, \delta_x + I_y\, \delta_y = -I_t, \tag{9}$$

where I_x and I_y are finite difference approximations to the partial derivatives in Eq. (8), evaluated via the reference image, and I_t is set to ΔI. Equation (9) applies to each pixel and contains two unknowns, δ_x and δ_y, for each equation (with one equation per pixel).
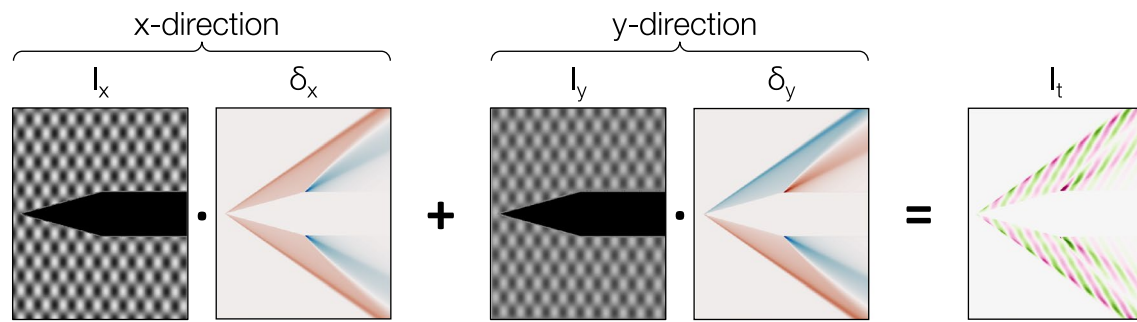
Fig. 4 Visualization of unified BOS: *x*- and *y*-direction image gradients are multiplied by deflections to estimate the image difference, i.e., Eq. (9) is calculated at each pixel. Color scales are the same as in Fig. 3

The Horn–Schunck OF closure is ubiquitous since it is both easy to implement and produces acceptable solutions for a wide array of applications (Horn and Schunck 1981). However, the method amounts to *first-order* Tikhonov regularization, which is known to produce overly-smooth results in the context of flow field measurement (Corpetti et al. 2006; Yuan et al. 2005; Kadri-Harouna et al. 2013). Recently, a variational method called wavelet-based OF analysis (wOFA) was shown to yield more accurate deflection fields for BOS than Horn–Schunck OF (Schmidt and Woike 2021). In wOFA, the deflection field is represented using a wavelet basis and the (hypothetical) distorted image, $I(x + \delta_x, y + \delta_y, t + \delta_t)$, is directly evaluated via bicubic spline interpolation; see (Schmidt and Sutton 2020) for a complete description of the method. This approach allows for the evaluation of residuals from Eq. (7) without computing the finite differences of I, I_x and I_y, as must be done in gradient-based OF. Further, the wavelet transform exploits regularities in the deflection field to facilitate a compressed representation of $\boldsymbol{\delta}_x$ and $\boldsymbol{\delta}_y$, reducing the number of unknowns.

We use the wOFA procedure described in Schmidt and Woike (2021) for deflection sensing in our conventional BOS workflow. Sample deflections from wOFA applied to the noise-free synthetic cone shock image pair are shown in Fig. 3. This figure also shows the *exact* deflections that were determined by nonlinear ray tracing. Despite the high-accuracy of the wOFA approach, "wavelet fingerprints" can be seen throughout the shocked region. These high-frequency discrepancies between the ideal (true) deflections and wOFA estimates act as measurement errors in the reconstruction algorithm.

2.4 Unified BOS

Until recently, deflection sensing and reconstruction were performed sequentially in BOS. As a result, non-physical regularization was needed in the deflection sensing step because $\boldsymbol{\delta}_x$ and $\boldsymbol{\delta}_y$ are not directly related to the density field

(rather, they are LoS-integrated quantities). However, Grauer and Steinberg (2020) observed that coupled gradient OF equations, i.e., one instance of Eq. (9) for each pixel, can formulated as a matrix system. The discrete BOS model from Sect. 2.2 can then be substituted into the matrix OF equation to relate the density field to raw image difference measurements. In other words, the density field can be reconstructed from a vector of image difference data in a single step. All regularization in this "unified" procedure is applied to the density field per se, and the errors associated with deflection sensing are avoided.

Figure 4 is a visualization of unified BOS in matrix form. Equation (9) is applied to each pixel, independently. This amounts to a diagonal matrix operation applied to $\boldsymbol{\delta}_x$ and $\boldsymbol{\delta}_y$, which may be expressed as a function of $\boldsymbol{\rho}$. To start, diagonal matrices are formed from the horizontal and vertical intensity gradients, $\mathbf{X}_{i,i} = I_{x,i}$ and $\mathbf{Y}_{i,i} = I_{y,i}$ for $i = 1, 2, \ldots, M$ for a system of M pixels. In this work, we employ second-order central differences to compute I_x and I_y. Next, we construct a data vector from the image pair, $\mathbf{b} = \{-I_{t,i}\}_{i=1}^{M}$. Lastly, the *x*- and *y*-direction instances of Eq. (6) are substituted in for $\boldsymbol{\delta}_x$ and $\boldsymbol{\delta}_y$, respectively, to relate $\mathbf{b}$ to $\boldsymbol{\rho}$,

$$\mathbf{X}\boldsymbol{\delta}_x + \mathbf{Y}\boldsymbol{\delta}_y = \underbrace{C_{\text{sys}}\left(\mathbf{X}\mathbf{D}_x + \mathbf{Y}\mathbf{D}_y\right)}_{\mathbf{A}} \boldsymbol{\rho} = \mathbf{b}. \tag{10}$$

Here, $\mathbf{A}$ is the $M \times N$ unified BOS operator. Equation (10) can be solved with the same reconstruction techniques developed to solve Eq. (6).

2.5 Axisymmetric reconstruction

There are many methods for reconstructing a 2D or 3D flow field from path-integrated measurements, as recently reviewed in Grauer et al. (2023). This paper considers the application of BOS to axisymmetric and planar flows. The geometric simplicity of these configurations facilitates an efficient representation of the fields and a simplified measurement model. A brief overview of Abel inversion for

axisymmetric BOS and a regularization technique for unified BOS are presented below.

2.5.1 Abel inversion for BOS

Path integrals through a radially symmetric object are described by the Abel transform, which has an explicit, analytical inverse. The forward transform *loosely* corresponds to the measurement model for many tomography modalities, such as absorption or emission tomography. Consequently, the inverse Abel transform can be adapted to reconstruct absorbance or emission data when the target object is axisymmetric. However, the BOS measurement model features a path integral over *gradients* of the density field, as opposed to integrals over ρ per se. This complication necessitates a tailored reconstruction strategy.

Recall the forward model for radial (y-axis) deflections in Eq. (3),

$$\delta_y = C_{\text{sys}} \int_{\text{ray}} \nabla_y \rho(s) \mathrm{d}s = C_{\text{sys}} \nabla_y \underbrace{\int_{\text{ray}} \rho(s) \mathrm{d}s}_{\bar{\rho}} = C_{\text{sys}} \nabla_y \bar{\rho}, \tag{11}$$

where $\bar{\rho}$ is the so-called projected density field, which corresponds to the forward Abel transform of ρ when the flow is axisymmetric (Raffel 2015). The gradient and integral operations in Eq. (11) are reversible,[3] leading to two distinct methods for reconstruction, which are outlined in Fig. 1.

1. Indirect reconstruction: Eq. (11) is used to construct a Poisson equation that may be solved for $\bar{\rho}$, which is recovered via the standard inverse Abel transform.[4]
2. Direct reconstruction: the Abel transform is modified to incorporate ∇_y and ρ is directly determined from δ_y. This resultant transform is often referred to as the "deflectometry" version (Kolhe and Agrawal 2009).

The Poisson equation in indirect Abel inversion contains gradients of δ_x and δ_y. Differentiating these variables amplifies errors from the deflection sensing procedure. Direct methods are comparatively stable, so we adopt the latter approach in this work.

The deflectometry Abel transform and its inverse are

---

[3] This result is a consequence of the Leibniz integral rule, assuming constant bounds of integration.

[4] These steps are commonly switched in 3D BOS tomography (Atcheson et al. 2008); this is also possible in axisymmetric BOS (see the work of Ota et al. (2015), Hirose et al. (2019), for instance, but doing so is rare.

$$\delta_y(y) = 2\, C_{\text{sys}} \int_y^R \nabla_y\, \rho(r) \frac{y}{\sqrt{r^2 - y^2}}\, \mathrm{d}r \quad \text{and} \tag{12a}$$

$$\rho(r) = -\frac{1}{\pi\, C_{\text{sys}}} \int_r^R \delta_y(y) \frac{1}{\sqrt{y^2 - r^2}}\, \mathrm{d}y, \tag{12b}$$

where the radial direction is aligned with the y-axis and R is the outer radius of the flow, beyond which δ_y is zero (Kolhe and Agrawal 2009). Notice that Eq. (12) requires continuous deflection data, which is not available in practice, so the expression must be discretized. The resultant direct deflectometry Abel inversion for a discrete field is

$$\boldsymbol{\rho} = C_{\text{sys}}^{-1}\, \mathbf{K}\boldsymbol{\delta}_y, \tag{13}$$

where $\mathbf{K}$ is an inverse Abel operator that is obtained by discretizing Eq. (12b). Since δ_y is resolved at discrete intervals (viz., elements of $\boldsymbol{\delta}_y$), $\mathbf{K}$ effectively interpolates this data. The two most common techniques for building $\mathbf{K}$ follow Simpson's 1/3 rule and the two-point method. Expressions for the elements of $\mathbf{K}$ derived using Simpson's 1/3 rule and two-point interpolation are provided in 1.

Unfortunately, the simplifications required to obtain $\mathbf{K}$, such as the assumption of parallel rays, are often violated by practical imaging systems. Further, Eq. (13) is a 1D transform that must be independently applied to each axial segment of a 2D flow. Neither axial gradients in the flow nor axial deflections are considered in the inversion. Therefore, inverting a high-fidelity 2D forward model like the unified BOS matrix is more robust (Daun et al. 2006; Sipkens et al. 2021).

2.5.2 Tikhonov regularization

The unified BOS model relates the (as yet unknown) density field to a measured image difference vector, $\mathbf{b}$. Therefore, this model must be inverted to estimate $\boldsymbol{\rho}$ from $\mathbf{b}$. This is a discrete ill-posed inverse problem and additional information is required for stability (Daun et al. 2016). We utilize a *second-order* Tikhonov penalty to promote smooth solutions:

$$\boldsymbol{\rho}_\omega = \arg\min_{\boldsymbol{\rho}} \big(\|\mathbf{A}\boldsymbol{\rho} - \mathbf{b}\|_2^2 + \omega_{\text{Tik}}^2 \|\mathbf{L}\boldsymbol{\rho}\|_2^2 \big). \tag{14}$$

In this expression, $\mathbf{L}$ is a discrete Laplacian operator (or "Tikhonov matrix"), defined in 1, ω_{Tik} is a regularization parameter that controls the influence of the penalty, and solutions may be obtained with a linear least squares algorithm. As ω_{Tik} goes to zero, $\boldsymbol{\rho}_\omega$ approaches the least squares solution, which is highly sensitive to noise in $\mathbf{b}$. Conversely, for very large values of ω_{Tik}, Eq. (14) is minimized by the uniform vector that minimizes the measurement residuals.

At moderate values of ω_{Tik}, however, $\boldsymbol{\rho}_\omega$ corresponds to a smooth field that approximately satisfies the measurement equations. Therefore, we optimize the regularization parameter through a phantom study, wherein $\boldsymbol{\rho}_\omega$ is computed from a synthetic image pair and compared to the exact density distribution. This process is repeated for a large range of regularization parameters, and we set ω_{Tik} to the value which maximizes the accuracy of $\boldsymbol{\rho}_\omega$.

3 Physics-informed BOS

Current BOS algorithms can produce a quantitative estimate of the density field in a target flow, but regularization is required at each stage of the workflow. Although most regularization schemes for BOS are *inspired* by physics, existing methods are ultimately incompatible with the underlying flow fields and thus give rise to errors or "reconstruction artifacts."

Data assimilation (DA) is a promising alternative to the standard suite of BOS techniques. DA algorithms seek to solve the equations governing fluid motion subject to data-based constraints (Hayase 2015). This approach avoids the pitfalls of the BOS methods described above and also provides access to the latent velocity, pressure, and energy fields. There are numerous methods that can be used to solve (or approximately solve) the governing equations while conforming to experimental measurements. Kalman filter (Cornick et al. 2009; Ali et al. 2022), state observer (Saredi et al. 2021), adjoint–variational (Mons et al. 2021; Wang et al. 2022), and hybrid simulation (Vinnichenko et al. 2022) algorithms have all been used to reconstruct flow fields with input from an experiment. For instance, local ensemble Kalman filter DA was employed to forecast temperature and velocity fields in a Rayleigh–Bénard convection cell from a set of experimental shadowgraphs (Cornick et al. 2009). A similar framework was developed at ONERA by Ali et al. (2022), who repeatedly solved the RANS equations and sequentially updated turbulence model parameters with a Kalman filter to match synthetic BOS measurements. State observer methods incorporate proportional or proportional–integral feedback, based on measurements of one or more fields, into the governing equations (Saredi et al. 2021). Variational techniques optimize a control vector, such as the initial flow state, to minimize an arbitrary data loss (Mons et al. 2021; Wang et al. 2022), and hybrid CFD simulations are conducted with one or more fields or parameters that are fixed by data. As an example of the latter technique, Vinnichenko et al. (2022) conducted a hybrid simulation of natural convection using a BOS-based estimate of the temperature field to determine the buoyancy term.

Unfortunately, these DA methods come at a high computational cost. Furthermore, all examples of BOS DA reported to-date have employed either (a) idealized synthetic BOS data (Ali et al. 2022) or (b) temperature field estimates from a conventional BOS algorithm (Cai et al. 2021; Vinnichenko et al. 2022). When the data loss, constraint, or forcing term in a DA algorithm includes *reconstructions*, as opposed to the measurement model and raw signal, the resultant fields are adversely affected by non-physical reconstruction artifacts introduced by the BOS algorithm. These artifacts can bias the DA algorithm or even prevent convergence, so it is essential to accurately mimic the image formation process in the data loss term (Molnar and Grauer 2022).

We conduct physics-informed BOS using a low-cost, flexible, easy-to-implement DA scheme in which the flow is represented with a PINN. The goal is to optimize an aggregate loss, consisting of data and physics residuals, using mature deep learning tools. The result is accurate, spatially-resolved estimates of all the flow fields.

3.1 PINN framework

Physics-informed neural networks utilize a deep, feed-forward network to map spatio-temporal inputs to flow fields (Raissi et al. 2019). Figure 5 shows the architecture of a PINN set up for physics-informed BOS. We utilize (x, r) and (x, y) as inputs for axisymmetric and planar flows, respectively. Density (ρ), velocity (u and v), and total energy (E), are the outputs. Automatic differentiation (AD) is employed to calculate exact partial derivatives of the network, and these partials are used to evaluate the governing equations throughout the measurement domain. The PINN does not generally satisfy these equations and the residuals are added up in a physics loss. Separately, a data (or measurement) loss is obtained by evaluating the unified BOS measurement model and comparing synthetic image differences, computed using the density field outputted by the PINN, to experimental image differences. The total objective loss (data + physics) is minimized via backpropagation to estimate the flow field in functional form.

In this work, we specify a physics loss using the compressible, steady, 2D Euler equations. This is an appropriate choice for the present demonstration, which features invisid flow, but viscous effects, transience, and 3D fields can be included as necessary. For now, each of the non-dimensional Euler equations is re-arranged to isolate a residual, ε:

$$\varepsilon_1 = (\rho\,u)_x + (\rho\,v)_r + \beta\left(\frac{1}{r}\rho\,v\right) \tag{15a}$$

$$\varepsilon_2 = \left(\rho\,u^2 + p\right)_x + (\rho\,v\,u)_r + \beta\left(\frac{1}{r}\rho\,u\,v\right) \tag{15b}$$
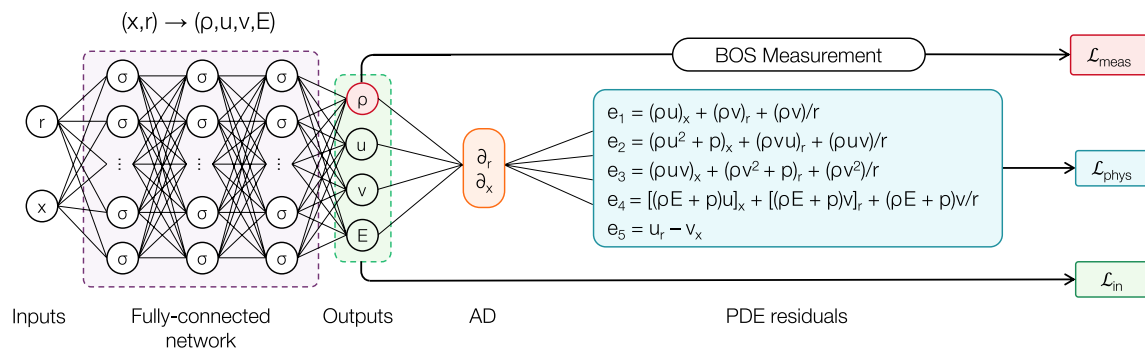
Fig. 5 Our PINNs map axial and radial (or vertical) coordinates to the fields of interest. The unified BOS measurement model is embedded in the data loss term; the Euler and irrotationality equations are employed for physics; experimental wind tunnel conditions are enforced at the inlet

$$\varepsilon_3 = (\rho u v)_x + \left(\rho v^2 + p\right)_r + \beta\left(\frac{1}{r}\rho v^2\right) \tag{15c}$$

$$\varepsilon_4 = \left[(\rho E + p)u\right]_x + \left[(\rho E + p)v\right]_r + \beta\left[\frac{1}{r}(\rho E + p)v\right]. \tag{15d}$$

Here, $(\cdot)_x$ and $(\cdot)_r$ denote partial derivatives with respect to x and r (naturally, r is replaced with y for planar cases) and β is the radial source coefficient which equals 1 for axisymmetric flow and 0 for planar flow. In order to compute residuals along the axis of symmetry, we multiply the right side of Eq. (15) by r. The field variables in Eq. (15) must be scaled by an appropriate reference to obtain dimensional values. We adopt the inflow conditions as our reference set, with a single reference for both components of velocity. A polytropic equation of state is employed to close the Euler equations,

$$p = (\gamma - 1)\rho\left[E - \frac{1}{2}\left(u^2 + v^2\right)\right], \tag{16}$$

where γ is the ratio of specific heats. Lastly, since this paper is concerned with approximately inviscid flow and approximately uniform inlet conditions, we consider an additional equation that is satisfied by irrotational velocity fields,

$$\varepsilon_5 = u_r - v_x. \tag{17}$$

Residuals from Eqs. (15) and (17) are integrated over the domain, culminating in an overall physics loss,

$$\mathcal{L}_{\text{phys}} = \frac{1}{\pi R^2 L} \int_0^L \int_0^R \left\|\left[\varepsilon_1, \ldots, \varepsilon_5\right]\right\|_2^2 2\pi r \, dr \, dx, \tag{18}$$

where L and R are the length and radius of the measurement domain. (Once again, the appropriate modifications are made for planar cases.) This expression is approximated by Monte Carlo sampling, and residuals from the irrotational equation are omitted in certain cases, as explicitly noted in the results sections.

There is an infinite set of flow fields that minimizes Eq. (18). Therefore, we connect the PINN to our experimental target with a data loss. This loss features the unified BOS model, such that the density field from the PINN is employed to compute image differences, which may then be compared to the unprocessed experimental images. In other words, our data loss is

$$\mathcal{L}_{\text{meas}} = \|\mathbf{A}\boldsymbol{\rho} - \mathbf{b}\|_2^2, \tag{19}$$

where $\mathbf{b}$ is contains experimental image differences, $\mathbf{A}$ is the unified BOS operator, and $\boldsymbol{\rho}$ is a vector of density data that is obtained from the PINN. We query the PINN at the support nodes of our basis, $\boldsymbol{\Phi}$, to construct $\boldsymbol{\rho}$. It should also be emphasized that the PINN outputs normalized values; hence, the outputted density data must be multiplied by a reference density (the inlet density in our case) to populate $\boldsymbol{\rho}$.

The final loss term is an inlet boundary condition, which is generally known to first-order for experiments such as the wind tunnel tests described in Sect. 4. Our inlet loss is

$$\mathcal{L}_{\text{in}} = \frac{1}{\pi R^2} \int_0^R \left\|\left[\rho - \rho_{\text{in}}, u - u_{\text{in}}, v - v_{\text{in}}, E - E_{\text{in}}\right]\right\|_2^2 2\pi r \, dr, \tag{20}$$

where ρ, u, v, and E are evaluated at $x = 0$ and $(\cdot)_{\text{in}}$ denotes an inlet value and the radial integral is replaced with a linear one for planar flow. The objective loss to be minimized is

$$\mathcal{L}_{\text{total}} = \omega_{\text{meas}} \mathcal{L}_{\text{meas}} + \omega_{\text{phys}} \mathcal{L}_{\text{phys}} + \omega_{\text{in}} \mathcal{L}_{\text{in}}. \tag{21}$$

This equation is uniquely minimized by the true flow fields so long as the PINN is big enough to express the flow, but the relative weight of the loss terms can help or hamper training. We optimize the weighting parameters, ω_{phys}, ω_{meas}, and ω_{in}, by conducting a simple parameter sweep with a synthetic case; optimal weights from this test, $\omega_{\text{meas}}/\omega_{\text{phys}} = 10$ and $\omega_{\text{in}}/\omega_{\text{phys}} = 100$, are used throughout the paper. Several groups have developed heuristics to programmatically assign weights to individual components of a PINN's objective

loss, e.g., Wang et al. (2022), Wang et al. (2021), Jin et al. (2021). However, the benefits of these adaptive techniques are generally marginal in the presence of realistic measurement noise, as discussed in Molnar and Grauer (2022). Another promising approach is to use a traditional algorithm for constrained optimization, like the *alternating direction method of multipliers*, which was applied to PINNs by Basir and Senocak (2022). We plan to assess this scheme in future research.

3.2 Network architecture and training

The PINNs used for this work are implemented in Tensor-Flow (Abadi et al. 2016). Networks that represent planar flows comprise five hidden layers, with 50 neurons per output variable, while the networks representing axisymmetric flows have ten hidden layers to ensure adequate expressivity. We employ swish activation functions unless otherwise noted. Weights are randomly initialized with a standard normal distribution and biases are initially set to zero.

Training is performed by minimizing Eq. (21) with the Adam optimizer (Kingma and Ba 2014) at a learning rate of 10^{-3} for the first three passes through the full image dataset and 10^{-4} thereafter. The PINNs are trained until the total loss plateaus; we define a plateau as a 5000-iteration stretch over which the 500-iteration running average of $\mathcal{L}_{\text{total}}$ varies by less than 0.5%. Here, an iteration is defined as a single update of the parameters by the Adam optimizer. Reconstructions are computed on an NVIDIA Tesla P100 graphics processing unit. Planar reconstructions initially took five hours, on average, and axisymmetric reconstructions took around nine hours; optimization of the code in Tensor-Flow 2.9.2 has reduced these times to approximately two and three hours, respectively.

3.3 PINNs for high-speed flow

Supersonic flow is compressible and subject to shock formations in most practical scenarios. The Euler equations are hyperbolic and shocks manifest as discontinuities in the corresponding solution. It is challenging to account for these effects with a numerical solver, which has led to a rich literature on bespoke CFD methods for supersonic conditions. Flow with shocks is similarly problematic for PINNs and several groups are actively adapting traditional CFD techniques to improve the ability of PINNs to represent high-speed flow. We discuss techniques for applying PINNs to hyperbolic equations in 2.

The use of PINNs to learn real compressible flow fields from experimental data is of particular interest, and several researchers have attempted to simulate this process with a *pseudo-schlieren* scenario. These tests use a loss consisting of point-wise $\nabla \rho$ "measurements" as opposed to LoS-integrated signals. Mao et al. (2020) pioneered the use of a pseudo-schlieren loss to estimate 1D shock-laden airflow (Mao also developed a forward solver for a 2D oblique shock with no schlieren-type data). Cai et al. (2022) reconstructed a synthetic 2D bow shock in the same way, quickly followed by the paper of Jagtap et al. (2022), who utilized domain decomposition (via an extended PINN) to facilitate the representation of oblique and bow shocks as well as an expansion fan. All three studies utilized a dense array of noise-free, synthetic, multi-modal measurements. For instance, Jagtap and coworkers specified 700 local density gradient pairs and 50 pressure taps in their data loss term. By way of context, well-instrumented cones support up to 16 taps (Casper et al. 2016), and a typical field-ready design has no taps at all. Since real schlieren data is LoS-integrated and camera rays may diverge in the measurement volume (Walsh et al. 2000; Sipkens et al. 2021), particularly when access windows are required for imaging, we restrict our tests to *plausible* synthetic data and real experimental images.

4 Measurement scenarios

We use physics-informed BOS to process experimental and synthetic image data for the axisymmetric cone cylinder scenario depicted in Fig. 6. Additionally, we demonstrate our technique on an analytical (planar) Prandtl–Meyer expansion fan. Both scenarios feature Mach 2 flow, leading to significant shock formations.

4.1 BOS measurements

4.1.1 Experimental setup

The BOS setup featured in this paper is located at the National Aerospace Laboratories (NAL), Bangalore, and is described in detail in Venkatakrishnan and Meier (2004). The facility houses a trisonic wind tunnel and the experiment is conducted in a square test section having a $0.3 \times 0.3 \, \text{m}^2$ profile. Square access windows, approximately 1" thick, are installed on opposite sides of the tunnel to enable imaging of the background pattern. The model is a 15° half-angle cone cylinder, positioned in the middle of the tunnel with zero inclination. This configuration was chosen because the resultant density field about the cone can be determined from classical cone shock tables (Sims 1964). Testing was conducted for Mach 2 inflow, which we assumed to be uniform throughout the cross section.

The background was placed 0.7 m from the tunnel window. Imaging was conducted with a scientific-grade camera (Kodak ES 1.0) having a 1 MP sensor and 9 μm pixels. The camera was fitted with a 50 mm lens, stopped down to *f*/8, and placed 1.4 m away from the tunnel, pointing towards

Fig. 6 Schmatic of the cone cylinder experiment at NAL, Bangalore. Mach 2 airflow passes over a 15° cone cylinder. The cylinder is located in the middle of a square 0.3×0.3 m^2 test section. Photos of a backlit background pattern are recorded through 1"-thick quartz acccess windows using a 1 MP camera
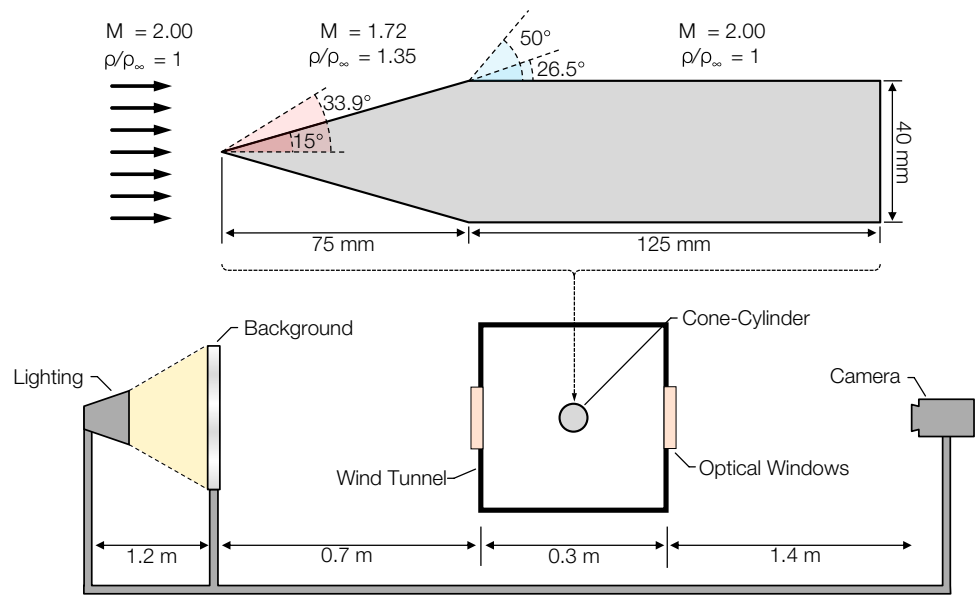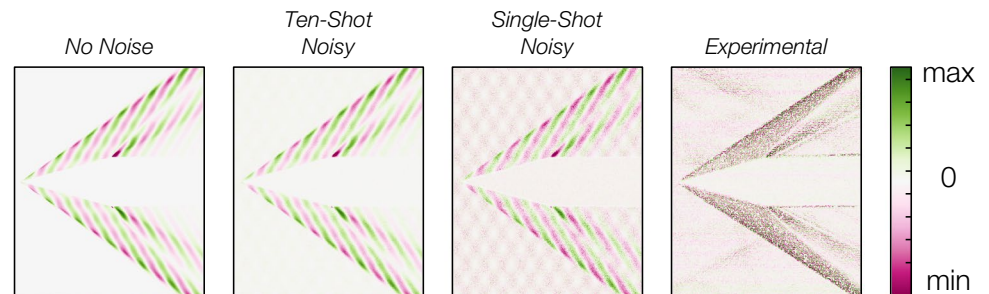
Fig. 7 Image difference data for a Mach 2 cone cylinder experiment. From left to right, the panel depicts clean, ten-shot average, and single-shot synthetic data followed by an experimental acquisition

the background plate. We consider a 540×652 px subsection of the sensor that is centered on the axis of symmetry. Experimental images of the cone are shown in Fig. 3, and the difference data can be seen on the right side of Fig. 7. Note that we used a *single pair* of experimental images. Significant artifacts are visible in resulting image differences, which exhibit noticeable striations that are emblematic of fixed pattern noise.
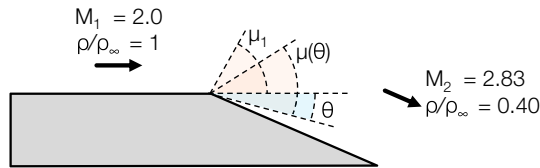
4.1.2 Generating synthetic data

In order to benchmark the accuracy of our reconstruction scheme, we test the method on synthetic images that correspond to ground truth ("phantom") density fields, introduced in the next section. To start, a pinhole camera model, based on the parameters presented above, is used to determine the principal ray for each point on the sensor; we supersample the sensor at four times the native resolution, i.e., using four rays per pixel. Linear ray tracing is employed for regions of constant density: outside the wind tunnel, inside the tunnel for the reference image, and through the windows (including

Snell's Law-type refraction at each interface). Further, we sample the aperture using the method of Cook et al. (1984) to account for the blurring and finite depth-of-field produced by a real aperture. Our measurement operator is thus effectively a "cone beam" sensitivity matrix that can mimic BOS measurements recorded with distinct *f*-numbers.

While reference images are generated by linear ray tracing throughout the domain, distorted images require nonlinear ray tracing inside the wind tunnel and linear ray tracing outside. Nonlinear ray tracing involves numerically solving the ray equation for a nonuniform refractive index field. We use the fourth-order Runge–Kutta scheme developed by Sharma et al. (1982) to solve Eq. (2), described in 3. Lastly, we corrupt some of the synthetic images with noise. Three cases are considered: clean (noise-free) image pairs, single-shot noisy image pairs, and ten-shot average image pairs. Individual noisy images are generated by applying a Gaussian blur filter with a standard deviation of 1.5 px and simulating shot-noise. The latter effect is modeled as a Poisson random variable, assuming a maximum signal of 5000 counts (Foi et al. 2008). Clean, ten-shot noisy, and single-shot difference data can be seen in the first three panels of Fig. 7. In

Table 1 Parameters for a Prandtl–Meyer Expansion Fan Phantom

Region	M	ρ, kg/m^3	u, m/s	v, m/s	p, kPa
Inflow	2	0.55	517.6	0.0	26.3
Outflow	2.83	0.22	572.5	−208.4	7.2

Fig. 8 Supersonic flow over a convex ramp leads to an expansion fan. The fan is characterized throughout in terms of a turning angle, θ, which is related to the Mach angle, μ

separate tests, we added 1–10% centered Gaussian errors to each *noisy* shot to mimic low-to-high levels of thermal noise.

Notice that the experimental images in Fig. 3 contain a dot pattern background whereas the synthetic background contains a superposition of axis-aligned sinusoidal waves. Historically dot patterns, inspired by PIV, were used to facilitate cross-correlation deflection sensing. However, since OF takes advantage of intensity gradients in the background, any blank space between the dots is of no use. Therefore, we stick to the sine wave pattern, which has been shown to improve performance (Grauer and Steinberg 2020; Schmidt and Woike 2021).

4.2 Phantom flow fields

4.2.1 Axisymmetric cone cylinder shock

We simulate the Mach 2 cone cylinder experiment reported in Venkatakrishnan and Meier (2004) to estimate the density, velocity, and energy fields corresponding to our experimental data. Simulations are performed using the compressible Euler solver in SU2 7.3.0 (Economon et al. 2016). The axisymmetric computational domain has a radius of 0.15 m and length of 0.25 m, excluding the cone cylinder geometry. Inlet conditions are specified to match experimental conditions for a settling pressure and temperature of 2.1 kg/cm^2 and 300 K, respectively, and $M = 2$, where M is the Mach number. The ratio of specific heats for air is set to the standard value of 1.4. Discretization errors were confirmed to be minimal through a grid convergence study, which is reported in 4, along with additional details about the simulation.

4.2.2 Planar expansion fan

A secondary, analytical phantom is considered, namely: the canonical Prandtl–Meyer expansion fan. This phenomenon occurs when supersonic flow encounters a convex ramp, as illustrated in Fig. 8; the flow accelerates over the ramp and sustains a continuous isentropic expansion across an infinite sequence of "Mach waves" (Anderson 1990). We generate an expansion fan for $M_1 = 2$ inflow that encounters a 20° corner. Key parameters of the flow are provided in Table 1, and details about the field calculations are given in 4. When simulating BOS measurements of the fan, we assume a slightly narrower test section of width 3.75 cm to limit the magnitude of deflections, such that the paraxial assumption holds true. A single pair of supersampled images are used in the inversion. Additionally, we consider a few cases with "pressure tap" information using the analytical value at the ramp.

5 Results and discussion

5.1 Axisymmetric cone cylinder shock

First, we compare density fields obtained by conventional BOS algorithms to our physics-informed estimates. Section 2.5 introduces three methods for axisymmetric BOS: Simpson's 1/3 Abel inversion, two-point Abel inversion, and unified BOS; the latter is conducted with a Sipkens kernel and Tikhonov regularization. Both Abel inversions require deflection data, which we obtain using a state-of-the-art wOFA algorithm (Schmidt and Woike 2021). Conversely, the unified BOS and physics-informed algorithms operate directly on the unprocessed image difference data.

Figure 9 depicts our Mach 2 cone cylinder density phantom as well as reconstructions computed with synthetic data (single-shot, noisy) and experimental data. The CFD simulation was designed to replicate the experimental conditions, so the phantom is expected to very nearly approximate true flow fields from the experiment. All BOS techniques considered in this paper produce a qualitatively accurate estimate of the density field. Uniform inflow compresses as it is deflected over the cone cylinder and expands as it flows past the shoulder. However, there is a clear difference in quality between these techniques. The explicit inverse Abel transforms underpredict compression, overpredict expansion, and exhibit a distinct error at the top-right corner that is associated with the Abel transform (namely, with the erroneous assumption of parallel rays). Results from these algorithms are substantially similar. Next, unified BOS with a Sipkens kernel produces a superior prediction in terms of magnitude,
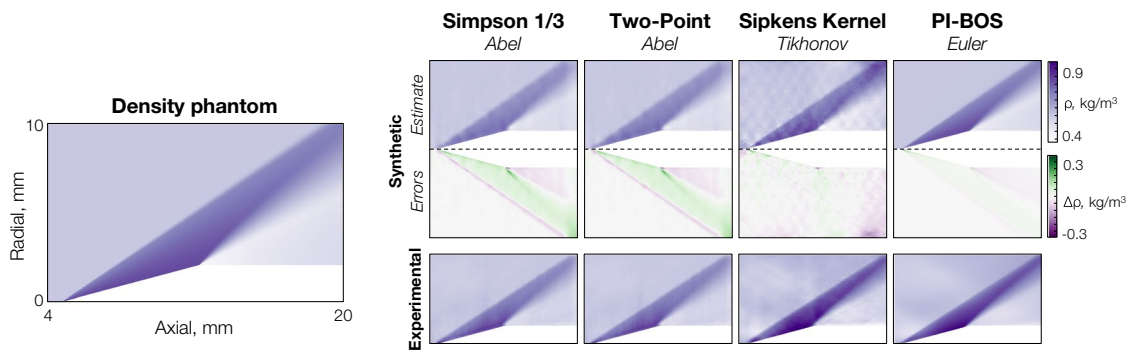
Fig. 9 Phantom density field and reconstructions from noisy (single-shot) synthetic and experimental data. Errors are plotted for the synthetic case. Physics-informed BOS significantly increases the accuracy of reconstructions

although artifacts from the sine wave background are clearly visible in the density field. These artifacts can be suppressed by increasing ω_{Tik}, but doing so damps the magnitude of the field and thereby amplifies reconstruction errors. Lastly, the physics-informed reconstruction almost perfectly recovers the density field, exhibiting crisp features, reminiscent of the ground truth density field, including a well-defined expansion fan.

To quantify relative performance, we calculate normalized root-mean-square errors (NRMSEs) for each reconstruction in percentage form. Both Simpson's 1/3 rule and two-point Abel inversion generate 8.34% error and the Sipkens kernel with Tikhonov smoothing produces 6.12% error for this phantom. By comparison, our physics-informed density field estimate has an error of only 3.75%. While it is not possible to comprehensively baseline the accuracy of our experimental reconstruction, we can compare the resultant density field to predicted values from an analytical cone shock table (Sims 1964). This comparison is shown in Fig. 10, alongside data from Venkatakrishnan and Meier (2004). Those authors used a 5.1 MP camera (Sony DSC F-707) to record BOS images of the same cone shock structure, and they reconstructed the density field via filtered back projection.[5] The plot corresponds to a density cut taken 2 mm downstream of the cone apex. Our reconstruction neatly matches the analytical result, exhibiting a closer correspondence than the results from (Venkatakrishnan and Meier 2004). Note that we omitted the 4° region immediately above the cone, which was severely affected by blur and calibration errors. Altogether, these results suggest that our physics-informed technique yields a qualitatively and quantitatively accurate estimate of the true field. Furthermore, to the best of our knowledge, this represents the first use of a PINN to reconstruct experimental measurements of a supersonic flow.

Not only does physics-informed BOS increase the accuracy of density field estimates, it also provides access to
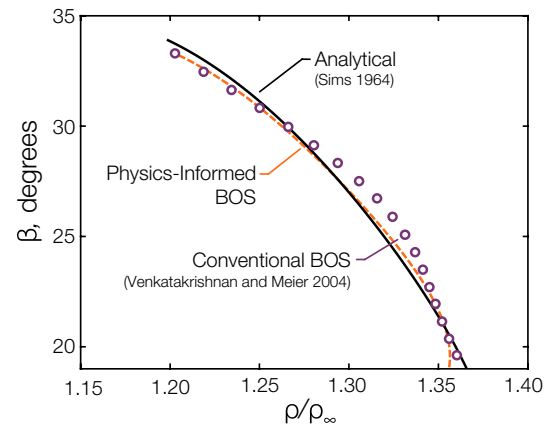

Fig. 10 Density ratio behind the forward shock, 2 mm downstream of the cone apex, as a function of angular elevation with respect to the axis of symmetry. Results are compared to previous experimental (Venkatakrishnan and Meier 2004) and analytical cone table (Sims 1964) data

the latent fields. Fundamentally, the PINN outputs density, velocity, and total energy fields, which may be used to compute pressure or temperature, as desired. Velocity and pressure are directly relevant to the analysis of aerodynamic performance, so we present density, velocity, and pressure fields obtained from the cone cylinder measurements. These fields are estimated from synthetic and experimental data using physics-informed BOS. NRMSEs for all estimates computed with synthetic data are provided in Table 2; the corresponding panel of phantoms and reconstructions can be found in Fig. 11.

All fields are accurately reconstructed from synthetic measurements, with a maximum error of 3.75%, 1.13%, and 3.09% in the density, velocity, and pressure estimates,

---

[5] Images from the 5.1 MP data set, recorded in 2003, are no longer available for processing.

Table 2 Percentage NRMSEs for Cone Shock Reconstructions from Best-Case Conventional and Physics-Informed BOS

Data	Density		u-velocity	v-velocity	Pressure
	Conv.	PI-BOS			
Clean	4.81	1.13	0.36	0.53	1.39
Clean (no ε_5)		1.22	1.39	0.74	1.56
Averaged	4.98	3.16	0.47	0.70	1.86
Avg. (no ε_5)		3.24	2.82	1.30	2.07
Single-shot	6.12	3.75	0.61	1.13	3.09
SS (no ε_5)		4.16	2.09	1.95	3.42

respectively, when the full loss is employed. This high degree of accuracy corresponds to a clear visual resemblance in all cases. Indeed, differences between the phantoms and reconstructions of synthetic data are nearly imperceptible, regardless of the level of noise. In subsequent tests with *additional* Gaussian errors, having a standard deviation up to 10% of the intensity range, reconstruction errors for the ten-shot data remained below 5%. However, the errors did increase with added noise (as expected), and visible artifacts are present in the experimental reconstruction due to significant noise in the image data and erratic background pattern gradients. Of particular note, all non-zero differences upstream of the compression shock in Fig. 7 are erroneous, indicating the intensity and spatial extent of the noise. Nevertheless, our experimental results are consonant with our
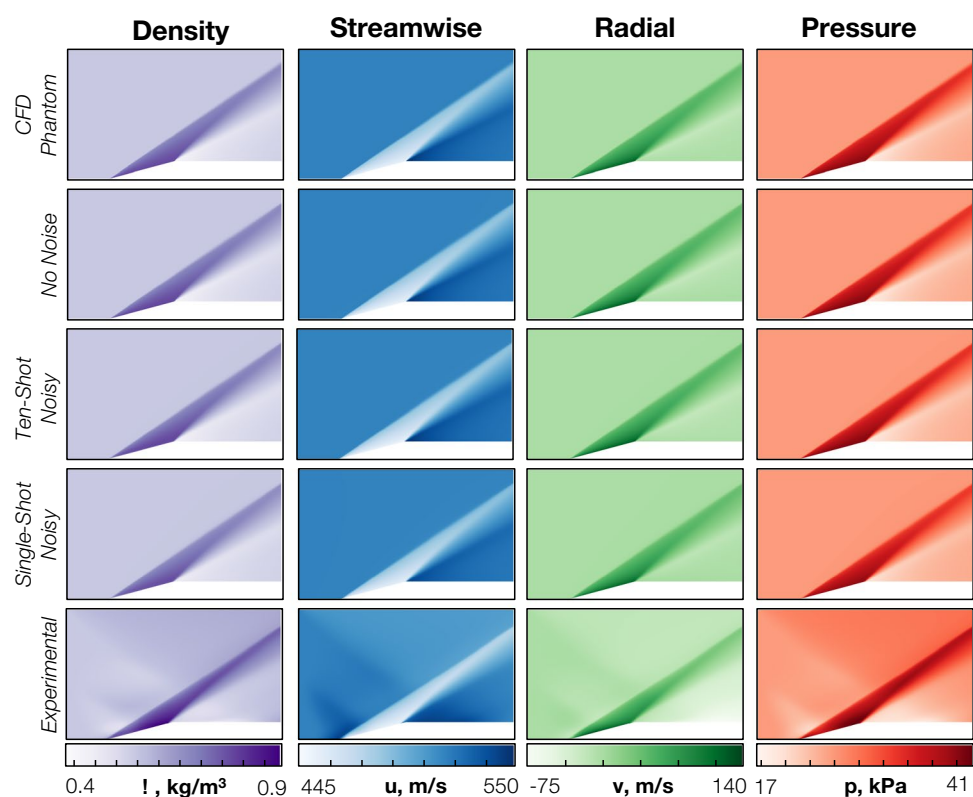
CFD results, which supports the use of physics-informed BOS. In addition, a Bayesian framework could be leveraged to counteract biased and correlated noise (Molnar and Grauer 2022). We also be note that the PINN contains fewer parameters than the CFD simulation, having 363,200 and 1,189,492 parameters, respectively, although the CFD solver has a runtime of approximately 10 min (as opposed to the PINN's several-hour training time).

Table 2 includes two sets of errors for each image pair: one based on the whole physics loss (i.e., with the Euler and irrotationality equations, ε_1–ε_4 and ε_5) and one without ε_5. The resultant reconstructions are shown in Fig. 12. The utility of adding an irrotational flow residual to $\mathcal{L}_{phys}$ is marginal for the synthetic cases. However, the benefit of including ε_5 increases with noise, per Table 2. Consequently, there is a significant qualitative improvement in the experimental results when ε_5 is used, which manifests as a clear reduction of artifacts, with an acute improvement in the magnitude of the velocity field past the expansion fan. This result suggests that including additional terms, where appropriate, such as an entropy pair residual, could further improve the stability and accuracy of physics-informed BOS.

5.2 Planar expansion fan

Second, we present results for the Prandtl–Meyer expansion fan. A very similar set of fields was used by Jagtap et al. (2022), allowing for a direct comparison between our PINN

Fig. 11 Reconstructions of all axisymmetric cone shock data sets. Synthetic data are reconstructed with increasing noise (clean, ten-shot average, and single-shot), followed by experimental data. All reconstructions (including the experimental case) bear a close resemblance to the CFD fields
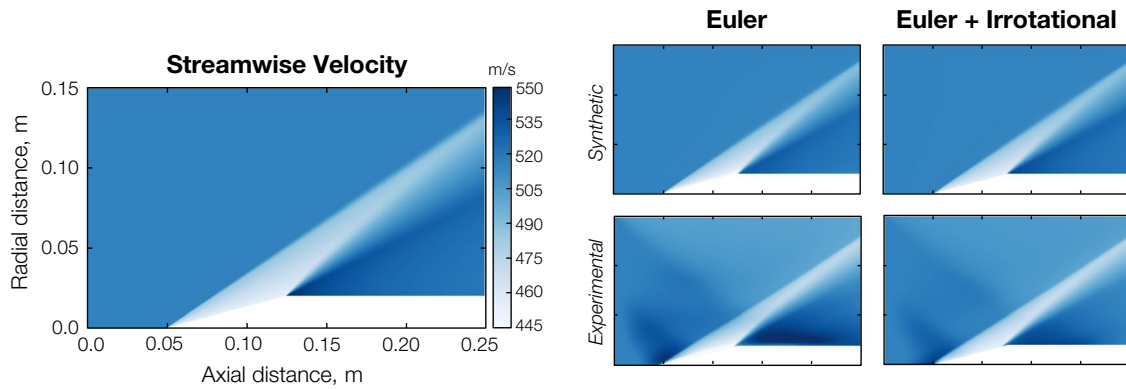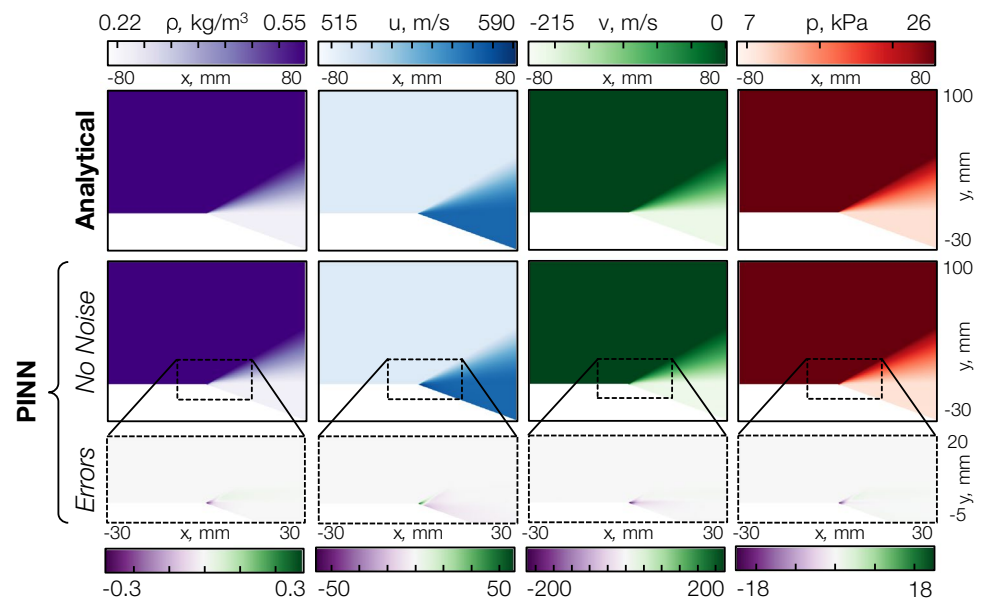
Fig. 12 Streamwise velocity (left) and single-shot synthetic and experimental reconstructions (right). Incorporating an irrotationality equation provides an improvement that is enhanced in the presence of noise, as for the experimental images
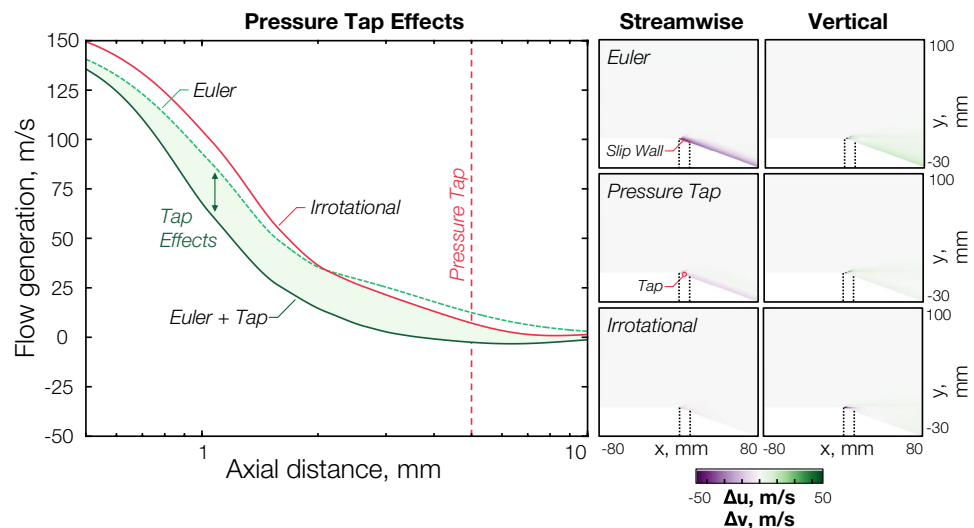
Fig. 13 Reconstructions of a planar expansion fan from clean synthetic data. Errors are concentrated about the singularity

implementations. Figure 13 depicts physics-informed BOS estimates of the fan's density, velocity, and pressure fields, all of which are recovered with high fidelity. The reconstructions exhibit minimal errors which are concentrated in the immediate vicinity of the singularity, located at the corner of the wedge. Including this region in the *data* loss term can cause the solution to blow up, resulting in nonsensical fields. We tested this using a BOS data loss as well as the local density gradient loss term employed by Jagtap et al. (2022). The instability is present for both loss formulations. To avoid this issue, we omit a minute region surrounding the corner from our data loss. Excluding these gradients is justified for BOS, in any case, because they are too strong to satisfy the paraxial assumption, even for the narrow test section of interest, i.e., 3.75 cm. By contrast, physics residuals are computed throughout the whole domain.

We also explored the stability of expansion fan reconstructions to varying measurement sources and physics loss components. Figure 14 depicts "flow generation" at the tip of the wedge and velocity error maps for different loss component scenarios. Positive flow generation indicates regions where the slip wall condition is not satisfied due to an erroneous inflow of momentum. We do not include a slip wall boundary condition in $\mathcal{L}_{total}$, so this behavior is inferred from the BOS data, Euler equations, and inlet conditions, alone. However, there is a short region at the leading edge of the ramp over which we observe flow generation errors. These errors quickly diminish with distance normal to the ramp and down the ramp. Including an additional measurement towards the leading edge of the ramp, such as a pressure tap, significantly diminishes flow generation (note the x-axis is plotted in log scale).

Fig. 14 Flow generation along the first ten millimeters of the 80 mm ramp, plotted for distinct loss combinations (left). Axial and transverse velocity errors for each loss combination (right). Dashed lines in the error maps indicate the region over which flow generation is plotted

While reconstructions of synthetic cone shock data are robust to the inclusion of ε_5 in $\mathcal{L}_{phys}$, fan reconstructions are more sensitive, possibly due to a greater multiplicity of weak solutions to the Euler equations at the zero-viscosity limit. Streamwise velocity errors decrease from 1.15% for a simple Euler loss to 0.61% when the pressure tap is included and to 0.48% for the Euler equations plus ε_5 (all other fields exhibit similar errors across these scenarios). In the first case, considering only the Euler equations, the PINN yields a significant over-prediction of streamwise velocity and under-prediction of the vertical component along the wedge. This is largely remedied by adding a single pressure tap 5 mm past the lip of the wedge, underlining the general principle that multi-modal measurements should be utilized whenever possible. Similar to Mao et al. (2020), we find that the location of the pressure tap is an important consideration. Moving the tap much further down the wedge, e.g., to 75 mm, fully eradicates the gains to accuracy that we observe in the 5 mm case. This fade-out occurs because the flow generation problem is local and the tap is most useful in a region where image data is lacking (recall that we exclude BOS data near the singularity in this case). Unfortunately, while a single tap can reduce flow generation, it is insufficient to fully mitigate velocity field errors. Full-field constraints, such as the irrotationality equation, provide much more utility, as can be seen in the error maps in Fig. 14.

In addition to these BOS-related tests, the expansion fan scenario serves as a convenient testbed for analyzing the performance of PINNs used for hyperbolic problems. For instance, we tested the dimensional and non-dimensional Euler equations in training and found that the former physics loss is far less stable.[6] This finding confirms the numerical tests of Haghighat et al. (2022), who attributed the performance differential to the disparate magnitudes of the *dimensional* Euler loss components, spanning roughly five orders of magnitude in our scenarios. An adaptive weighting scheme could potentially help to overcome this issue, but we previously observed that such schemes are unstable for noisy data (Molnar and Grauer 2022). Therefore, we recommend using the non-dimensional Euler equations, instead. Separately, we tested a suite of activation functions, including the swish, GELU, hyperbolic tangent, leaky ReLU, and sigmoid functions. To lacklustre effect, the performance of these functions was quite similar, although each choice exhibited a unique optimum of ω_{meas}, ω_{phys}, and ω_{in}, likely because some activation functions saturate (e.g., tanh and sigmoid) while others do not (e.g., swish and GELU).

6 Conclusions and outlook

We present a data assimilation technique for background-oriented schlieren called physics-informed BOS. This approach combines a comprehensive measurement model, based on unified BOS, with the governing physical equations, in this case the Euler and irrotational equations, to infer steady density, velocity, and energy fields from a single pair of images. The method utilizes a physics-informed neural network to represent the flow; PINNs are flexible, easy-to-use tools for DA, but they had not previously been used to reconstruct supersonic flow from experimental data. Indeed, this work reports the first such use of a PINN to the best of our knowledge. We report accurate, multi-parameter reconstructions without the use of pressure data, artificial viscosity

---

[6] Note that the dimensional and non-dimensional Euler equations have the same form.

regularization, or an entropy condition. Several important conclusions can be drawn from this work.

1. Physics-informed BOS with a PINN yields more accurate estimates of the density fields than conventional BOS algorithms. In all synthetic cases, differences between the phantoms and reconstructions are nearly imperceptible. Additionally, in the experimental case, there is good agreement with the CFD solution.
2. Additional physics-based residuals, such as an irrotationality equation, can be included in the physics loss to improve the stability of training, accuracy of reconstructions, and resilience to noise. In much the same way, multi-modal measurement information can guide the PINN towards the correct, physical solution during training.
3. Akin to conventional CFD techniques, singularities pose a significant challenge for PINNs and need to be addressed. In scenarios where a singularity may arise, physics constraints should be strengthened to stabilize the DA scheme.
4. Tuning the relative weight of loss components is essential to the procedure, and the optimal weighting depends upon one's choice of activation functions.

Appendix A: Bases and kernels for axisymmetric BOS tomography

Appendix A.1: Bases

Recalling Eq. (4), the radially-symmetric density field is approximated as a sum over the basis, Φ,

$$\rho(x, r) \approx \sum_j^N \rho_j \, \varphi_j(x, r), \tag{22}$$

where φ_j is the jth basis function. Typically, these functions are uniform, linear, or quadratic in BOS tomography. Following Sipkens et al. (2021), we utilize a uniform discretization scheme in the axial direction and a linear scheme in the radial direction.

The uniform axial functions are given by

$$\eta_j(x) = H(x - x_j) - H(x - x_{j+1}), \tag{23}$$

for a set of axial locations, x_j, where $H(\cdot)$ is the Heaviside function. Differentiating η with respect to x yields

$$\frac{\partial \eta_j}{\partial x} = f_\delta(x - x_j) - f_\delta(x - x_{j+1}), \tag{24}$$

where f_δ is the Dirac delta function. Our radial basis comprises piecewise linear functions that span circular annuli,

$$\phi_j(r) = \begin{cases} \frac{r - r_{j-1}}{r_j - r_{j-1}} & r_{j-1} < r \leq r_j \\ \frac{r - r_{j+1}}{r_j - r_{j+1}} & r_j < r \leq r_{j+1} \\ 0 & \text{otherwise} \end{cases}. \tag{25}$$

Note that this expression must be modified at the first and final functions to exclude the $j - 1$ and $j + 1$ terms, as appropriate, assuming a uniform inner circle for the first basis function and a field that decays to zero at the outer radius. The radial gradient of ϕ_j is

$$\frac{\partial \phi_j}{\partial r} = \begin{cases} \frac{1}{r_j - r_{j-1}} & r_{j-1} < r \leq r_j \\ \frac{1}{r_j - r_{j+1}} & r_j < r \leq r_{j+1} \\ 0 & \text{otherwise} \end{cases}. \tag{26}$$

Lastly, we combine these bases to obtain an axisymmetric linear–uniform basis,

$$\rho(x, r) \approx \sum_j^N \rho_j \underbrace{\eta(x)\phi(r)}_{\varphi_j(x, r)}, \tag{27}$$

where φ_j is the resultant 2D basis function. The second-order Tikhonov matrix for this basis has elements

$$L_{i,j} = \begin{cases} -4, & i = j \\ 1, & i \text{ axially adjacent to } j \\ (2r_i \pm h)/(2r_i), & i \text{ radially adjacent to } j \\ 0, & \text{otherwise} \end{cases}, \tag{28}$$

where $\pm$ corresponds to $+$ and $-$ for outer and inner radial neighbors, respectively, and each element is scaled by the grid spacing, h^{-2}, assuming equal spacing in x and r, i.e., $h = \Delta x = \Delta r$. This definition is modified to enforce the free stream refractive index at the outer radius. Effectively, this penalty utilizes the discrete cylindrical Laplacian of ρ to promote reconstructions that are both axially- and radially-smooth.

Appendix A.2: Kernels

Three direct, conventional kernels are utilized to baseline the PINN approach. In this context, "direct" means that the kernel relates the deflection data to a density field (Simpson's 1/3 rule and the two-point method) or vice versa (the Sipkens kernal).[7] "Indirect" refers to methods that require an explicit Poisson solver.

Simpson's rule arises from a traditional approach to numerical integration in which the area under a curve is approximated

---

[7] We incorporate the Sipkens kernel into a unified BOS operator, which relates the density field directly to unprocessed image differences, although unified BOS is not inherent to that scheme.

using the area under parabolic curves. Similarly, the two-point method utilizes piecewise linear interpolation with a quadratic expansion about the singularity. For a smooth target function with sufficient resolution, these approximations are highly similar. Both techniques yield a kernel, $\mathbf{K}$, that relates $\boldsymbol{\delta}_y$ to a radial density field, $\boldsymbol{\rho}$. Separately, the Sipkens kernel, derived in Sipkens et al. (2021), is a discrete forward operator that relaxes the assumption of parallel rays. Sipkens et al. (2021) developed the transform for a variety of bases; we utilize the linear–uniform basis introduced above. The result is deflection operators, $\mathbf{D}_x$ and $\mathbf{D}_y$, that are used to construct a unified BOS measurement model.

Appendix A.2.1: Simpson's 1/3 rule

Simpson's 1/3 rule involves piecewise curve fits to the diverging integral. The kernel has elements

$$K_{i,j} = \begin{cases} 0, & i > j \\ J_{i,j}/2, & j = N+1 \\ J_{i,j}, & i < j \text{ and } j \neq N+1 \end{cases}, \tag{29a}$$

where

$$J_{i,j} = -\frac{1}{3\pi} \frac{3 + (-1)^{j-i+1}}{\sqrt{(j-1)^2 - (i-1)^2}} \tag{29b}$$

and N is the number of equally spaced intervals. It should be noted that the integrand diverges at the lower limit of integration. Therefore, the singularity is extrapolated from the remaining coefficients. This is expressed in the kernel as

$$K_{i,i} = \begin{cases} K_{i,i+1}, & i \leq N \\ 0, & i = N+1 \end{cases}. \tag{30}$$

One key difference between Simpson's rule and the two-point method is that the former requires an even number of intervals due to the curve fitting technique, which uses three points. Nevertheless, Simpson's rule exhibits greater accuracy and faster convergence than integration by many other such rules (e.g., trapezoidal).

Appendix A.2.2: Two-point method

When using the two-point method, the integral is discretized between neighboring radii and the deflection is presumed constant between those two rings (hence the name). The kernel is formed using pairs of projections, resulting in elements

$$K_{i,j} = \begin{cases} 0, & i > j \\ J_{i,j}, & i = j \\ J_{i,j} - J_{i,j-1}, & i < j \end{cases}, \tag{31}$$

where

$$J_{i,j} = \begin{cases} 0, & i > j \\ 2/\pi, & i = j = 0 \\ \frac{1}{\pi} \log \left[\frac{\sqrt{(j+1)^2 - i^2} + j + 1}{\sqrt{(j-1)^2 - i^2} + j} \right] & i \leq j \end{cases}. \tag{32}$$

This formulation has no smoothing. The lack of smoothing can be an advantage over other reconstruction kernels given noise-free data, but it comes the a cost of noise amplification. The resulting deconvolution kernel is an upper triangular matrix, and the divergence of the integral is handled via a quadratic expansion of the projections at the singularity (Cormack 1982).

Appendix A.2.3: Sipkens deflectometry

The Sipkens kernel is a forward operator that can model deflections along arbitrary, non-parallel rays. This formulation can simultaneously operate on deflection data above and below the axis of symmetry, unlike the explicit inverse kernels described above, which improves the stability and resolution of reconstructions. Moreover, the forward kernel can be incorporated into any inverse solver, such as our physics-informed BOS technique. Here, we recall the final form of the linear–uniform Sipkens deflectometry operator using the present notation; a full derivation can be found in the supplementary material of Sipkens et al. (2021).

To start, the kernel requires a 3D description of each ray. Consider the ith ray as it passes by the z-axis at the point $(x_i, y_i, 0)$ with a slope of $m_{x,i}$ in the x-z plane and $m_{y,i}$ in the y-z plane. For ease of notation, we introduce a function that corresponds to the radius of the ith ray at this crossing point,

$$r_{\text{ray},i}(x) = \frac{1}{m_{x,i}} \sqrt{(x - x_i)^2 + [m_{y,i}(x - x_i) + y_i]^2}. \tag{33}$$

Each ray is divided into two, with one on either side of $z = 0$, i.e., a ray approaching the central axis and a ray departing from it. Consequently, the vertical deflectometry operator is split into two parts,

$$D_{y,i,j} = D_{y,i,j}^+ + D_{y,i,j}^-. \tag{34}$$

The "approaching" and "departing" elements have a very similar expression, which we present in a consolidated form,

$$
D_{\mathrm{y},i,j}^{\pm} = \frac{\sqrt{1 + m_{\mathrm{x},i}^2}}{1 + m_{\mathrm{y},i}^2} \sum_{j}^{N} \left(y_{0,i} \left\{ a_{\mathrm{int}}\left[r_{j-1}, r_j, r_2(j)\right] - a_{\mathrm{int}}\left[r_{j-1}, r_j, r_1(j)\right] + a_{\mathrm{int}}\left[r_{j+1}, r_j, r_3(j)\right] \right. \right.
$$
$$
\left. \left. - a_{\mathrm{int}}\left[r_{j+1}, r_j, r_2(j)\right] \right\} \pm m_{\mathrm{y},i} \left[\frac{r_{\mathrm{b}}(j) - r_{\mathrm{b}}(j-1)}{r_j - r_{j-1}} - \frac{r_{\mathrm{b}}(j) - r_{\mathrm{b}}(j+1)}{r_j - r_{j+1}} \right] \right),
$$
(35)

where $\pm$ becomes $+$ for $D_{\mathrm{y},i,j}^+$ and $-$ for $D_{\mathrm{y},i,j}^-$,

$$
a_{\mathrm{int}}\left(r_1, r_2, r_3\right) = \frac{1}{r_2 - r_1} \log \left\{ \mathrm{abs}\left[r_3 + \left(r_3^2 - \frac{y_{0,i}^2}{1 + m_{\mathrm{y},i}^2} \right)^{1/2} \right] \right\},
$$
(36)

and r_{b} modifies the jth radius to incorporate the bounds of the Heaviside function. See (Sipkens et al. 2021) for a discussion of the r_1, r_2, and r_3 functions of j in Eq. (35).

The axial deflection operator is simply

$$
D_{\mathrm{x},i,j} = \mathrm{abs}\left\{ \phi_j\left[r_{\mathrm{ray},i}(x_j)\right] - \phi_j\left[r_{\mathrm{ray},i}(x_{j+1})\right] \right\}.
$$
(37)

Since the axial basis is piecewise uniform, $\mathbf{D}_{\mathrm{x}}$ only has non-zero elements for rays that cross an axial boundary. While this representation is sub-optimal, it is sufficient for the present demonstration. A bi-linear basis will be derived for future use. Equations (35) and (37) are employed to populate $\mathbf{D}_{\mathrm{y}}$ and $\mathbf{D}_{\mathrm{x}}$, which are themselves required to construct the unified BOS measurement model in Eq. (10).

Appendix B: PINNs applied to hyperbolic equations

Here, we briefly review two key developments in the area of physics-informed neural networks applied to hyperbolic equations. This topic directly pertains to PINNs discussed in this work because the Euler equations are hyperbolic.

Appendix B.1: Artificial viscosity

"Artificial viscosity" was initially introduced into Euler solvers to accommodate discontinuous flow fields by spreading shocks across a finite region that could be resolved by the grid (von Neumann and Richtmyer 1950; Lax 1959). Artificial viscosity entails the deliberate addition of dissipation into the governing equations, often based on the size of the grid, such that the equations can be solved across the shock without introducing excessive errors.

Several researchers have applied this concept to PINNs to provide stability in the context of a hyperbolic physics loss.

Fuks and Tchelepi (2020) were the first to do this for two-phase transport in porous media, i.e., adding viscous damping to a Buckley–Leverett transport model. Coutinho and coworkers (Coutinho et al. 2022) adopted a similar approach and considered the inviscid Burgers' equation in addition to a Buckley–Leverett scenario; Coutinho also introduced a variety of adaptive methods to automatically tune the dissipation term. More recently, Patel et al. (2022) proposed the use of a control volume physics loss with artificial viscosity to solve a 1D shock governed by Burgers' equation, a 1D Sod shock problem, and generic 1D Buckley–Leverett, Euler, and Leblanc problems. Collectively, these papers demonstrate that including artificial viscosity can reduce oscillations about a discontinuity and stabilize optimization of the network.

Most hyperbolic solvers with artificial viscosity require careful selection of the viscous term, which may vary across space and time. Excessive damping is known to yield stable yet invalid solutions (Anderson 1990). This was noted by Liu et al. (2022), leading to their usage of a gradient-dependent ω_{phys} parameter, which diminished in highly compressible regions, instead of an artificial viscosity scheme. Further, Wang et al.'s (2021) analysis of PINNs demonstrates that higher-order gradients dominate the cumulative gradient vector used to update the network parameters. This result suggests that a non-physical diffusion term can obfuscate progress towards the correct, physical solution. Since BOS measurements provide relatively direct access to the real density field, we do not employ artificial viscosity for shock capturing.

Appendix B.2: Entropy pair regularization

Many hyperbolic systems of equations, including the compressible Euler equations, admit multiple "weak" solutions. The physical solution is often selected by applying

a constraint like an entropy condition, which is valid for inviscid flows that exhibit one or more discontinuities. This condition leads to an inequality that relates mathematical entropy to the divergence of an entropy-flux pair (De Lellis and Székelyhidi 2010); the inequality is satisfied by desirable "viscosity solutions" to the governing equations.

Two groups have added an entropy loss to a PINN to learn shock-laden flow. Patel et al. (2022) first implemented this technique, using a complete equation of state that bounds the Euler equations by a specific entropy relation. Separately, Jagtap et al. (2022) used a flux pair coupled with a polytropic equation of state. Patel's formulation resulted in three loss components whereas Jagtap's method culminated in a scalar loss. Both groups reported that entropy pair regularization improved learning for hyperbolic problems compared to a vanilla PINN. However, it should be noted that a unique entropy pair is not guaranteed for a given set of hyperbolic conservation laws (Godlewski and Raviart 2013). Therefore, we chose to forego an entropy pair loss, which proved to be unnecessary for our test cases.

Appendix C: Nonlinear ray tracing

Synthetic data are generated using nonlinear ray tracing within the variable index field. This is accomplished to high accuracy via the fourth-order Runge–Kutta ray tracing algorithm of Sharma et al. (1982). First, Eq. (2) is split into coupled ordinary differential equations. Next, each ray is defined by a starting position, $\mathbf{x}_0 = [x, y, z]^\mathrm{T}$, and a refractive index-scaled direction,

$$\mathbf{v}_0 = n(\mathbf{x})\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}s} \tag{38}$$

where $\mathrm{d}\mathbf{x}/\mathrm{d}s$ is the initial trajectory of the ray. Next, consider the vector valued function

$$\mathbf{f}(\mathbf{x}) = \Delta s\, n(\mathbf{x})\boldsymbol{\nabla} n(\mathbf{x}) = \tfrac{1}{2}\,\Delta s\,\boldsymbol{\nabla} n^2(\mathbf{x}). \tag{39}$$

We evaluate $\mathbf{f}$ at three points along the ray,

$$\begin{aligned}
\mathbf{f}_\mathrm{A} &= \mathbf{f}\big(\mathbf{x}_i\big)\\
\mathbf{f}_\mathrm{B} &= \mathbf{f}\Big[\mathbf{x}_i + \Delta s\Big(\tfrac{1}{2}\mathbf{v}_i + \tfrac{1}{8}\mathbf{f}_\mathrm{A}\Big)\Big],\\
\mathbf{f}_\mathrm{C} &= \mathbf{f}\Big[\mathbf{x}_i + \Delta s\Big(\mathbf{v}_i + \tfrac{1}{2}\mathbf{f}_\mathrm{B}\Big)\Big]
\end{aligned} \tag{40}$$

where the subscript i denotes the current index along the path of the ray. The ray's position and direction are updated,

$$\begin{aligned}
\mathbf{x}_{i+1} &= \mathbf{x}_i + \Delta s\Big[\mathbf{v}_i + \tfrac{1}{6}\big(\mathbf{f}_\mathrm{A} + 2\mathbf{f}_\mathrm{B}\big)\Big],\\
\mathbf{v}_{i+1} &= \mathbf{v}_i + \tfrac{1}{6}\big(\mathbf{f}_\mathrm{A} + 4\mathbf{f}_\mathrm{B} + \mathbf{f}_\mathrm{C}\big)
\end{aligned} \tag{41}$$

and the procedure is repeated until all rays exit the computational domain.

Appendix D: Generation of flow phantoms

This appendix provides additional details about our phantoms.
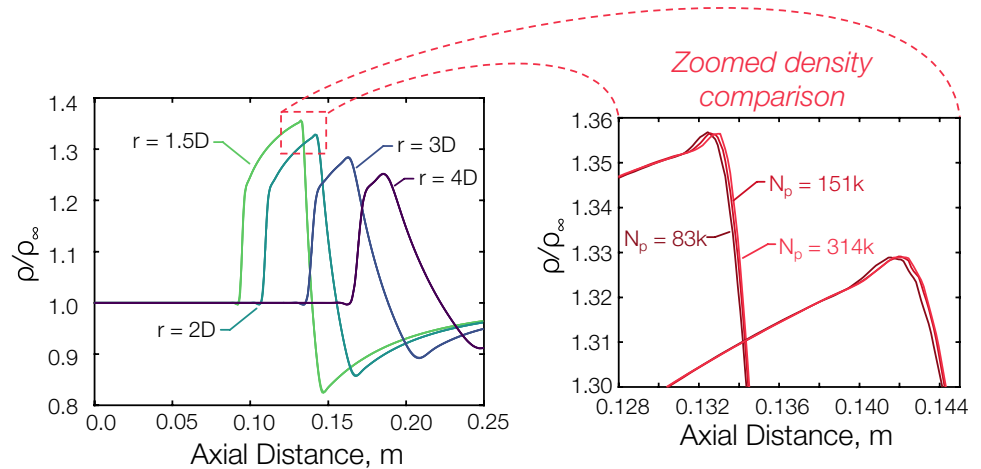
Appendix D.1: Axisymmetric cone cylinder

We simulated the cone cylinder flow using SU2 7.3.0, which is a finite volume solver; convective fluxes are handled with a second-order upwind scheme in space that is TV diminishing when paired with a limiter (Roe 1981). The Venkatakrishnan slope limiter is utilized to combat the oscillatory behavior of higher-order upwind schemes by enforcing a monotonicity condition (Venkatakrishnan 1995). Additionally, gradients are computed using a weighted least squares algorithm. We use a 2D axisymmetric unstructured grid that comprises 297,373 triangular cells, with moderate refinement towards the anticipated shock locations and tripping points. Slip-wall conditions, based on the inviscid flow assumption, are implemented along the cone cylinder and top of the domain, which represents the physical boundary of the wind tunnel.

A grid convergence study was performed to ensure that the solution was independent of the grid. Computations were performed with cells of size 1.25×10^{-4} m, 6.25×10^{-5} m, and 3.125×10^{-5} m. Results from this study are shown in Fig. 15. Radial slices of the non-dimensionalized density field are plotted, with a zoomed view provided to illustrate the close agreement between simulations. Notably, the density, velocity, and energy errors all asymptote, indicating that the shocks are well captured by all three schemes. Errors resulting from the computational grid are thus expected to be marginal. Throughout the paper, we use results computed with the intermediate grid, with 6.25×10^{-5} m cells, to generate synthetic data.

Appendix D.2: Planar expansion fan

The planar expansion fan serves as a useful phantom to evaluate the BOS reconstruction method developed here, given the analytical nature. Specifically, this solution arises through a geometric analysis of a sequence of Mach waves, resulting in the governing differential equation for Prandtl–Meyer flow,

Fig. 15 Axial cuts of the simulated cone shock density field for three computational grids (left). Cuts are shown at selected radii, and a zoomed comparison is presented (right). The CFD results exhibit minimal discrepancies, even across the shocks

$$d\theta = \sqrt{M^2 - 1}\,\frac{dV}{V}, \tag{42}$$

where θ is the turning angle towards the ramp and V is the local flow speed. Equation (42) is expressed in terms of dM/M and integrated to the ramp to obtain the outflow Mach number,

$$\int_{\theta_1}^{\theta_2} d\theta = \int_{M_1}^{M_2} \frac{\sqrt{M^2 - 1}}{1 + \frac{\gamma-1}{2}M^2}\,\frac{dM}{M}. \tag{43}$$

The indefinite solution to the right side of Eq. (43), called the Prandtl–Meyer function, is

$$\nu(M) = \sqrt{\frac{\gamma+1}{\gamma-1}}\tan^{-1}\left[\sqrt{\frac{\gamma-1}{\gamma+1}(M^2-1)}\right] - \tan^{-1}\left(\sqrt{M^2-1}\right), \tag{44a}$$

which can be used to solve Eq. (43) across a differential wedge,

$$\theta - \theta_1 = \nu[M(\theta)] - \nu(M_1). \tag{44b}$$

For a horizontal wall leading up to the ramp, θ_1 is zero and the Mach number is computed as a function of the turning angle. This may be done by solving Eq. (44b) via Newton's method,

$$M^{(k+1)}(\theta) = \frac{\theta + \nu(M_1) - \nu\left[M^{(k)}(\theta)\right]}{\nu'\left[M^{(k)}(\theta)\right]} + M^{(k)}(\theta), \tag{45}$$

where

$$\nu'(M) = \frac{\sqrt{M^2 - 1}}{M\left(1 + \frac{\gamma-1}{2}M^2\right)} \tag{46}$$

is the analytical derivative of the Prandtl–Meyer function. Once the Mach field has been determined, the other field variables are calculated using the isentropic relations,

$$T(\theta) = T_1\left[\frac{1 + \frac{\gamma-1}{2}M_1^2}{1 + \frac{\gamma-1}{2}M(\theta)^2}\right], \tag{47a}$$

$$p(\theta) = p_1\left[\frac{T(\theta)}{T_1}\right]^{\frac{\gamma}{\gamma-1}}, \quad \text{and} \tag{47b}$$

$$\rho(\theta) = \frac{p(\theta)}{R_{\text{gas}}\,T(\theta)}, \tag{47c}$$

where $R_{\text{gas}} = 287$ J/kg K is the gas constant for air.

Although Mach numbers in the fan are calculated as a function of θ, each Mach line is oriented at the corresponding Mach angle,

$$\mu(\theta) = \sin^{-1}\left[M(\theta)^{-1}\right],\tag{48}$$

as drawn in Fig. 8. The local flow speed is simply

$$V = M\,a,\tag{49}$$

where

$$a = \sqrt{\gamma\,R_{\text{gas}}\,T}\tag{50}$$

is the speed of sound. Finally, the flow accelerates tangent to Mach lines throughout the fan such that the streamlines are parallel to the wall up to the forward Mach line, normal to the turning angle throughout the fan, and parallel to the wedge thereafter,

$$u = V\cos(\theta) \quad \text{and}\tag{51a}$$

$$v = -V\sin(\theta).\tag{51b}$$

Declarations

Conflict of interest All author declares no conflict of interest.

References

Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M et al. (2016) TensorFlow: a system for large-scale machine learning, in 12th USENIX symposium on operating systems design and implementation (OSDI 16), pp 265–283

Agrawal AK, Albers BW, Griffin DW (1999) Abel inversion of deflectometric measurements in dynamic flows. Appl Opt 38:3394–3398

Ali MYB, Léon O, Donjat D, Bézard H, Laroche E, Mons V, Champagnat F (2022) Data assimilation for aerothermal mean flow reconstruction using aero-optical observations: a synthetic investigation. In: 56th 3AF international conference on applied aerodynamics, p 11

Anderson JD (1990) Modern compressible flow: with historical perspective, vol 12. McGraw-Hill, New York

Atcheson B, Ihrke I, Heidrich W, Tevs A, Bradley D, Magnor M, Seidel H-P (2008) Time-resolved 3D capture of non-stationary gas flows. ACM Trans Graphics 27:1–9

Atcheson B, Heidrich W, Ihrke I (2009) An evaluation of optical flow algorithms for background oriented schlieren imaging. Exp Fluids 46:467–476

Basir S, Senocak I (2022) Physics and equality constrained artificial neural networks: application to forward and inverse problems with multi-fidelity data fusion. J Comput Phys 111301

Born M, Wolf E (2013) Principles of optics: electromagnetic theory of propagation, interference and diffraction of light. Elsevier, Amsterdam

S. Cai, Z. Wang, F. Fuest, Y. J. Jeon, C. Gray, and G. E. Karniadakis, Flow over an espresso cup: inferring 3-D velocity and pressure fields from tomographic background oriented Schlieren via physics-informed neural networks, J. Fluid Mech. **915** (2021)

Cai S, Mao Z, Wang Z, Yin M, Karniadakis GE (2022) Physics-informed neural networks (PINNs) for fluid mechanics: a review. Acta Mech Sin 1–12

Casper KM, Beresh SJ, Henfling JF, Spillers RW, Pruett BO, Schneider SP (2016) Hypersonic wind-tunnel measurements of boundary-layer transition on a slender cone. AIAA J 54:1250–1263

Castner R (2012) Exhaust nozzle plume effects on sonic boom. J Aircr 49:415–422

Cook RL, Porter T, Carpenter L (1984) Distributed ray tracing. In: Proceedings of the 11th annual conference on computer graphics and interactive techniques, pp 137–145

Cormack AM (1982) Computed tomography: some history and recent developments. In: Proceedings of symposia in applied mathematics. American Mathematical Society, pp 35–42

Cornick M, Hunt B, Ott E, Kurtuldu H, Schatz MF (2009) State and parameter estimation of spatiotemporally chaotic systems illustrated by an application to Rayleigh-Bénard convection. Chaos Int J Nonlinear Sci 19:013108

Corpetti T, Heitz D, Arroyo G, Mémin E, Santa-Cruz A (2006) Fluid experimental flow estimation based on an optical-flow scheme. Exp Fluids 40:80–97

Coutinho EJR, Dall'Aqua M, McClenny L, Zhong M, Braga-Neto U, Gildin E (2022) Physics-informed neural networks with adaptive localized artificial viscosity. 2203.08802

Dalziel SB, Hughes GO, Sutherland BR (2000) Whole-field density measurements by 'synthetic schlieren'. Exp Fluids 28:322–335

Daun KJ, Thomson KA, Liu F, Smallwood GJ (2006) Deconvolution of axisymmetric flame properties using Tikhonov regularization. Appl Opt 45:4638–4646

Daun KJ, Grauer SJ, Hadwin PJ (2016) Chemical species tomography of turbulent flows: discrete ill-posed and rank deficient problems and the use of prior information. J Quant Spectrosc Radiat Transfer 172:58–74

Davies ER (2004) Machine vision: theory, algorithms, practicalities. Elsevier, Amsterdam

De Lellis C, Székelyhidi L (2010) On admissibility criteria for weak solutions of the Euler equations. Arch Ration Mech Anal 195:225–260

Dolvin D (2008) Hypersonic international flight research and experimentation (HIFiRE) fundamental science and technology development strategy. In: 15th AIAA international space planes and hypersonic systems and technologies conference, p 2581

Economon TD, Palacios F, Copeland SR, Lukaczyk TW, Alonso JJ (2016) SU2: An open-source suite for multiphysics simulation and design. AIAA J 54:828–846

Foi A, Trimeche M, Katkovnik V, Egiazarian K (2008) Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. IEEE Trans Image Process 17:1737–1754

Fuks O, Tchelepi HA (2020) Limitations of physics informed machine learning for nonlinear two-phase transport in porous media. J Mach Learn Model Comput 1

Gardiner W Jr, Hidaka Y, Tanzawa T (1981) Refractivity of combustion gases. Combust Flame 40:213–219

Geerts JS, Yu KH (2017) Systematic application of background-oriented schlieren for isolator shock train visualization. AIAA J 55:1105–1117

Godlewski E, Raviart P-A (2013) Numerical approximation of hyperbolic systems of conservation laws, vol 118. Springer, New York

Goldhahn E, Seume J (2007) The background oriented schlieren technique: sensitivity, accuracy, resolution and application to a three-dimensional density field. Exp Fluids 43:241–249

Gomez M, Grauer SJ, Ludwigsen J, Steinberg AM, Son SF, Roy S, Meyer TR (2022) Megahertz-rate background-oriented schlieren tomography in post-detonation blasts. Appl Opt 61:2444–2458

Grauer SJ, Steinberg AM (2020) Fast and robust volumetric refractive index measurement by unified background-oriented schlieren tomography. Exp Fluids 61:1–17

Grauer SJ, Unterberger A, Rittler A, Daun KJ, Kempf AM, Mohri K (2018) Instantaneous 3D flame imaging by background-oriented schlieren tomography. Combust Flame 196:284–299

Grauer SJ, Mohri K, Yu T, Liu H, Cai W (2023) Volumetric emission tomography for combustion processes. Prog Energy Combust Sci 94:101024

Haghighat E, Amini D, Juanes R (2022) Physics-informed neural network simulation of multiphase poroelasticity using stress-split sequential training. Comput Methods Appl Mech Eng 397:115141

Hayase T (2015) Numerical simulation of real-world flows. Fluid Dyn Res 47:051201

Heineck JT, Banks DW, Smith NT, Schairer ET, Bean PS, Robillos T (2021) Background-oriented schlieren imaging of supersonic aircraft in flight. AIAA J 59:11–21

Hirose Y, Ishikawa K, Ishimoto Y, Nagashima T, Ota M, Udagawa S, Inage T, Kiritani H, Fujita K, Fujita K et al (2019) The quantitative density measurement of unsteady flow around the projectile. J Flow Control Meas Visual 7:111

Horn BK, Schunck BG (1981) Determining optical flow. Artif Intell 17:185–203

Howard M, Fowler M, Luttman A, Mitchell SE, Hock MC (2016) Bayesian Abel inversion in quantitative X-ray radiography. SIAM J Sci Comput 38:B396–B413

Jagtap AD, Mao Z, Adams N, Karniadakis GE (2022) Physics-informed neural networks for inverse problems in supersonic flows. J Comput Phys 466:111402

Jin X, Cai S, Li H, Karniadakis GE (2021) NSFnets (Navier–Stokes flow nets): physics-informed neural networks for the incompressible Navier-Stokes equations. J Comput Phys 426:109951

Kadri-Harouna S, Dérian P, Héas P, Mémin E (2013) Divergence-free wavelets and high order regularization. Int J Comput Vision 103:80–99

Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. arXiv preprint 1412.6980

Kogelschatz U, Schneider W (1972) Quantitative schlieren techniques applied to high current arc investigations. Appl Opt 11:1822–1832

Kolehmainen V, Somersalo E, Vauhkonen P, Vauhkonen M, Kaipio J (1998) A Bayesian approach and total variation priors in 3D electrical impedance tomography. In: Proceedings of the 20th annual international conference of the IEEE Engineering in Medicine and Biology Society, (IEEE), pp.1028–1031

Kolhe PS, Agrawal AK (2009) Abel inversion of deflectometric data: comparison of accuracy and noise propagation of existing techniques. Appl Opt 48:3894–3902

Lax P (1959) Systems of conservation laws. Tech. rep, Los Alamos National Lab

Liu L, Liu S, Yong H, Xiong F, Yu T (2022) Discontinuity computing with physics-informed neural network. 2206.03864

Lucas BD, Kanade T et al (1981) An iterative image registration technique with an application to stereo vision. In: DARPA image understanding workshop, pp 121–130

Mao Z, Jagtap AD, Karniadakis GE (2020) Physics-informed neural networks for high-speed flows. Comput Methods Appl Mech Eng 360:112789

Molnar JP, Grauer SJ (2022) Flow field tomography with uncertainty quantification using a Bayesian physics-informed neural network. Meas Sci Technol 33:065305

Mons V, Du Y, Zaki TA (2021) Ensemble-variational assimilation of statistical data in large-eddy simulation. Phys Rev Fluids 6:104607

Nicolas F, Todoroff V, Plyer A, Le Besnerais G, Donjat D, Micheli F, Champagnat F, Cornic P, Le Sant Y (2016) A direct approach for instantaneous 3D density field reconstruction from background-oriented schlieren (BOS) measurements. Exp Fluids 57:1–21

Ota M, Leopold F, Noda R, Maeno K (2015) Improvement in spatial resolution of background-oriented schlieren technique by introducing a telecentric optical system and its application to supersonic flow. Exp Fluids 56:1–10

Patel RG, Manickam I, Trask NA, Wood MA, Lee M, Tomas I, Cyr EC (2022) Thermodynamically consistent physics-informed neural networks for hyperbolic systems. J Comput Phys 449:110754

Raffel M (2015) Background-oriented schlieren (BOS) techniques. Exp Fluids 56:1–17

Raffel M, Willert CE, Kompenhans J et al (1998) Particle image velocimetry: a practical guide, vol 2. Springer

Raissi M, Perdikaris P, Karniadakis GE (2019) Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J Comp Phys 378:686–707

Rajendran LK, Bane SP, Vlachos PP (2019) Dot tracking methodology for background-oriented schlieren (BOS). Exp Fluids 60:1–13

Rajendran LK, Zhang J, Bhattacharya S, Bane SP, Vlachos PP (2020) Uncertainty quantification in density estimation from background-oriented Schlieren measurements. Meas Sci Technol 31:054002

Roe PL (1981) Approximate Riemann solvers, parameter vectors, and difference schemes. J Comput Phys 43:357–372

Saredi E, Ramesh NT, Sciacchitano A, Scarano F (2021) State observer data assimilation for RANS with time-averaged 3D-PIV data. Comput Fluids 218:104827

Schmidt B, Sutton J (2019) High-resolution velocimetry from tracer particle fields using a wavelet-based optical flow method. Exp Fluids 60:1–17

Schmidt B, Sutton J (2020) Improvements in the accuracy of wavelet-based optical flow velocimetry (wOFV) using an efficient and physically based implementation of velocity regularization. Exp Fluids 61:1–17

Schmidt BE, Woike MR (2021) Wavelet-based optical flow analysis for background-oriented schlieren image processing. AIAA J 59:3209–3216

Sharma A, Kumar DV, Ghatak AK (1982) Tracing rays through graded-index media: a new method. Appl Opt 21:984–987

Sims JL (1964) Tables for Supersonic Flow Around Right Circular Cones at Zero Angle of Attack, vol 3004. Office of Scientific and Technical Information, National Aeronautics and Space Administration

Sipkens T, Grauer S, Steinberg A, Rogak S, Kirchen P (2021) New transform to project axisymmetric deflection fields along arbitrary rays. Meas Sci Technol 33:035201

Sommersel O, Bjerketvedt D, Christensen S, Krest O, Vaagsaether K (2008) Application of background oriented schlieren for quantitative measurements of shock waves from explosions. Shock Waves 18:291–297

J. Stam and E. Languénou, Ray tracing in non-constant media. In: Eurographics workshop on rendering techniques. Springer, pp 225–234

Szeliski R (2010) Computer vision: algorithms and applications. Springer, New York

Tokgoz S, Geisler R, Van Bokhoven L, Wieneke B (2012) Temperature and velocity measurements in a fluid layer using background-oriented schlieren and PIV methods. Meas Sci Technol 23:115302

Vauhkonen M, Vadász D, Karjalainen PA, Somersalo E, Kaipio JP (1998) Tikhonov regularization and prior information in electrical impedance tomography. IEEE Trans Med Imaging 17:285–293

Venkatakrishnan V (1995) Convergence to steady state solutions of the Euler equations on unstructured grids with limiters. J Comput Phys 118:120–130

Venkatakrishnan L, Meier G (2004) Density measurements using the background oriented schlieren technique. Exp Fluids 37:237–247

Vinnichenko NA, Plaksina YY, Pushtaev AV, Uvarov AV (2022) Obtaining velocity and pressure distributions in natural convection flows using experimental temperature fields. Appl Therm Eng 118962

von Neumann J, Richtmyer RD (1950) A method for the numerical calculation of hydrodynamic shocks. J Appl Phys 21:232–237

Walsh KT, Fielding J, Long MB (2000) Effect of light-collection geometry on reconstruction errors in Abel inversions. Opt Lett 25:457–459

Wang S, Teng Y, Perdikaris P (2021) Understanding and mitigating gradient flow pathologies in physics-informed neural networks. SIAM J Sci Comput 43:A3055–A3081

Wang S, Yu X, Perdikaris P (2022) When and why PINNs fail to train: a neural tangent kernel perspective. J Comput Phys 449:110768

Wang Q, Wang M, Zaki TA (2022) What is observable from wall data in turbulent channel flow? J Fluid Mech 941

Wei C, Schwarm KK, Pineda DI, Spearrin RM (2021) Volumetric laser absorption imaging of temperature, CO and CO_2 in laminar flames using 3D masked Tikhonov regularization. Combust Flame 224:239–247

Yamagishi M, Yahagi Y, Ota M, Hirose Y, Udagawa S, Inage T, Kubota S, Fujita K, Ohtani K, Nagai H (2021) Quantitative density measurement of wake region behind reentry capsule (Improvements in accuracy of 3D reconstruction by evaluating the view-angle of measurement system). J Fluid Sci Technol 16:JFST0021

Yuan J, Ruhnau P, Mémin E, Schnörr C (2005) Discrete orthogonal decomposition and variational fluid flow estimation. In: International conference on scale-space theories in computer vision. Springer, pp 267–278