



# A meshless and binless approach to compute statistics in 3D ensemble PTV

Manuel Ratz<sup>1</sup> · Miguel A. Mendez<sup>1</sup>

Received: 12 March 2024 / Revised: 17 August 2024 / Accepted: 24 August 2024  
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2024

## Abstract

We propose a method to obtain super-resolution of turbulent statistics for three-dimensional ensemble particle tracking velocimetry (EPTV). The method is “meshless” because it does not require the definition of a grid for computing derivatives, and it is “binless” because it does not require the definition of bins to compute local statistics. The method combines the constrained radial basis function (RBF) formalism introduced Sperotto et al. (Meas Sci Technol 33:094005, 2022) with an ensemble trick for the RBF regression of flow statistics. The computational cost for the RBF regression is alleviated using the partition of unity method (PUM). Three test cases are considered: (1) a 1D illustrative problem on a Gaussian process, (2) a 3D synthetic test case reproducing a 3D jet-like flow, and (3) an experimental dataset collected for an underwater jet flow at  $Re = 6750$  using a four-camera 3D PTV system. For each test case, the method performances are compared to traditional binning approaches such as Gaussian weighting (Agüí and Jiménez in JFM 185:447–468, 1987), local polynomial fitting (Agüera et al. in Meas Sci Technol 27:124011, 2016), as well as binned versions of RBFs.

## 1 Introduction

Much research has focused on developing image-based three-dimensional and three-component velocity measurements (3D3C Scarano (2013)) in the last two decades. The first popular 3D3C technique is the tomographic particle image velocimetry (PIV) introduced by Elsinga et al. (2006). This extends the planar cross-correlation-based PIV to a three-dimensional setting, where interrogation windows are replaced by interrogation volumes. The main limitation is the computational cost, which scales poorly when moving from 2D to 3D, and the unavoidable spatial filtering produced by a correlation-based evaluation. Recently, 3D particle tracking velocimetry (PTV) has emerged as a promising alternative, offering better computational performances and much higher spatial resolution (Kähler et al. 2012a, b, 2016; Schröder and Schanz 2023).

A key enabler to the success of 3D PTV has been the development of advanced tracking algorithms such as

Shake-the-Box (Schanz et al. 2016) or its open-source variant (Tan et al. 2020). These, together with advancements in the particle reconstruction process (Wieneke 2013; Schanz et al. 2013; Jahn et al. 2021), allow processing images with a particle seeding concentration up to 0.125 particles per pixel (ppp), well above the limits of 0.005 ppp of early tracking methods (Maas et al. 1993). Nevertheless, PTV processing produces randomly scattered data. This poses many challenges to post-processing, from the simple computation of gradients (e.g., to compute vorticity) and flow statistics to more advanced pressure integration. Although post-processing methods based on unstructured grids have been proposed (see Neeteson and Rival (2015); Neeteson et al. (2016)), the most common approach is to interpolate the scattered data onto a uniform grid that allows using traditional post-processing approaches (e.g., finite differences for derivatives, ensemble statistics, modal decompositions, etc.).

When interpolation onto Cartesian grid aims at treating instantaneous fields, for example for derivative computations and/or pressure reconstruction, the most popular approaches are Vic+ and Vic# (Schneiders and Scarano 2016; Scarano et al. 2022; Jeon et al. 2022), constrained cost minimization (Agarwal et al. 2021), the FlowFit algorithm (Schanz et al. 2016; Gesemann et al. 2016) or Meshless Track Assimilation (Sperotto et al. 2024b). These methods require time-resolved data and introduce some physics-based penalty or

✉ Manuel Ratz  
manuel.ratz@vki.ac.be

Miguel A. Mendez  
miguel.alfonso.mendez@vki.ac.be

<sup>1</sup> von Karman Institute for Fluid Dynamics,  
Sint-Genesius-Rode, Belgium

constraint to make the interpolation more robust. Examples are the divergence-free condition on the velocity fields or the curl-free condition for acceleration and pressure fields.

When interpolation onto Cartesian grids aims at computing flow statistics, such as mean fields or Reynolds stresses, the most popular approaches are based on the concept of binning and ensemble PTV (EPTV, Discetti and Coletti (2018)). This method involves dividing the measurement domain into bins, within which local statistics are computed by treating all samples in a bin as part of a local distribution (Kähler et al. 2012a). If sufficiently dense clouds of points are available, these methods can significantly outperform cross-correlation-based approaches in computing Reynolds stresses (Pröbsting et al. 2013; Atkinson et al. 2014; Schröder et al. 2018).

EPTV methods vary in how local statistics, particularly second- or higher-order moments, are computed. A traditional approach, often called “top-hat,” assigns equal weight to all samples within a bin. In contrast, the more advanced Gaussian weighting method by Agüí and Jiménez (1987) assigns greater weight to samples closer to the bin center. Godbersen and Schröder (2020) demonstrated that integrating a fit of individual particle tracks significantly improves convergence. However, this approach requires particle tracks over multiple time steps, obtained either from time-resolved measurements or multi-pulse data (Novara et al. 2016). Agüera et al. (2016) demonstrated that the top-hat approach suffers from unresolved velocity gradients, while Gaussian weighting results in slower statistical convergence. These issues are exacerbated in three-dimensional EPTV, where achieving statistical convergence may require an impractically large number of samples. To address these limitations, Agüera et al. (2016) proposed using local polynomial fits within each bin to regress the mean flow and then compute higher-order statistics on the mean-subtracted fields. This method combines spatial averaging with ensemble averaging, allowing for a larger bin size (which benefits statistical convergence) without compromising the resolution of gradients in the mean flow. However, the mean flow is only locally smooth, does not account for physical priors and provides statistics only at the bin’s centers.

In this work, we aim to extend the concept of blending and integrate it with the meshless framework proposed by Sperotto et al. (2022), recently released in an open-source toolbox called SPICY (super-resolution and pressure from image velocimetry, Sperotto et al. (2024a)). The meshless approach is a new paradigm in PTV data post-processing, where the interpolation step is entirely removed, and all post-processing operations (such as computations of derivatives, correlations or pressure fields) are performed analytically. In the approach proposed by Sperotto et al. (2022), the analytic representation is built using physics-constrained radial basis functions (RBFs). The goal of operating on analytically (symbolically differentiable) fields

bridges assimilation methods in velocimetry with machine learning-based super-resolution techniques, including deep learning (Park et al. 2020), physics-informed neural networks (PINNs, Rao et al. (2020)) and generative adversarial networks (Güemes Jiménez et al. 2022). The primary advantage of the RBF formulation is its linearity with respect to the training parameters, allowing for efficient training and implementation of hard constraints.

The approach proposed in this work employs the constrained RBF framework for spatial averaging, similar to the local polynomial regression by Agüera et al. (2016). However, we use an ensemble trick to avoid the need for defining bins, resulting in an analytic expression for the statistical quantities that is both grid-free and bin-free. The general formulation is presented in Sect. 2. Section 3 outlines the main numerical recipes to implement the RBF constraints while significantly reducing computational costs compared to the original implementation in Sperotto et al. (2022). This is achieved using a simplified version of the well-known partition of unity method (PUM, Melenk and Babuška (1996)) for RBF regression (see also Larsson et al. (2017); Cavoretto and De Rossi (2019, 2020)). The PUM significantly reduces memory and computational demands by splitting the domain into patches, performing RBF regression in each patch, and then merging the solutions into a single regression. The RBF-PUM was recently applied for super-resolution of Shake-the-Box measurements (Li et al. 2021) and mean flow fields in microfluidics (Ratz et al. 2022b), though without penalties or constraints. Section 4 presents the selected test cases for evaluating the algorithm’s performance, while Sect. 5 reviews the algorithms used for benchmarking. Results are presented in Sect. 6, and conclusions and perspectives are discussed in Sect. 7.

## 2 Bin-Free Statistics

We briefly review the fundamentals of radial basis function (RBF) regression in Sect. 2.1. Section 2.2 introduces the ensemble trick to circumvent the need for binning.

### 2.1 Fundamentals of RBF regression and notation

The RBF regression consists of approximating a function as a linear combination of radial basis functions. In this work, we are interested in approximating the components of 3D velocity fields and consider only isotropic Gaussian RBFs:

$$\varphi_k(\mathbf{x}|\mathbf{x}_{c,k}, c_k) = \exp(-c_k^2 \|\mathbf{x} - \mathbf{x}_{c,k}\|^2), \quad (1)$$

where  $\mathbf{x} = (x, y, z) \in \mathbb{R}^3$  is the coordinate where the basis is evaluated,  $\mathbf{x}_{c,k} = (x_{c,k}, y_{c,k}, z_{c,k}) \in \mathbb{R}^3$  and  $c_k$  are, respectively, the  $k$ -th collocation point and the shape parameter of the basis and  $\|\cdot\|$  denotes the  $l_2$  norm of a vector.

At any given point  $\mathbf{x}$ , the velocity field has three entries  $\mathbf{u}(\mathbf{x}) = (u(\mathbf{x}), v(\mathbf{x}), w(\mathbf{x})) \in \mathbb{R}^3$ . The RBF regression using  $n_b$  RBFs can be written as:

$$\mathbf{u}(\mathbf{x}) = \begin{pmatrix} u \\ v \\ w \end{pmatrix} \approx \sum_{k=1}^{n_b} \begin{pmatrix} w_{u,k} \varphi_k(\mathbf{x}|\mathbf{x}_{c,k}, c_k) \\ w_{v,k} \varphi_k(\mathbf{x}|\mathbf{x}_{c,k}, c_k) \\ w_{w,k} \varphi_k(\mathbf{x}|\mathbf{x}_{c,k}, c_k) \end{pmatrix}, \tag{2}$$

where  $w_{u,k}, w_{v,k}, w_{w,k} \in \mathbb{R}^{n_b}$  are the weights associated to each basis. The function approximation (2) can conveniently be evaluated on an arbitrary set of points  $\mathbf{X} = (\mathbf{x}, \mathbf{y}, \mathbf{z})$ , with  $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{R}^{n_p}$  the vectors collecting the coordinates in each point, using the basis matrix  $\Phi_b(\mathbf{X})$ : This collects the value of each RBF on a set of points  $\mathbf{X}$ :

$$\Phi_b(\mathbf{X}) = \begin{pmatrix} \vdots & \dots & \vdots \\ \varphi(\mathbf{X}; \mathbf{x}_{c,1}, c_1) & \dots & \varphi(\mathbf{X}; \mathbf{x}_{c,n_b}, c_{n_b}) \\ \vdots & \dots & \vdots \end{pmatrix}, \tag{3}$$

This matrix allows us to express approximation (2) in a compact notation:

$$\mathbf{U}(\mathbf{X}) = \begin{pmatrix} \mathbf{u}(\mathbf{X}) \\ \mathbf{v}(\mathbf{X}) \\ \mathbf{w}(\mathbf{X}) \end{pmatrix} \approx \begin{pmatrix} \Phi_b(\mathbf{X}) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Phi_b(\mathbf{X}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \Phi_b(\mathbf{X}) \end{pmatrix} \begin{pmatrix} \mathbf{w}_u \\ \mathbf{w}_v \\ \mathbf{w}_w \end{pmatrix}. \tag{4}$$

This block structure is useful when constraints are introduced later on. To ease the notation, we abbreviate (4) to  $\mathbf{U}(\mathbf{X}) \approx \Phi \mathbf{W}$ . Here, it is understood that  $\mathbf{U} \in \mathbb{R}^{3n_p}$  and  $\mathbf{W} \in \mathbb{R}^{3n_b}$  are the vertically concatenated velocity field and weights, respectively.

We assume that training data (e.g., PTV measurements) are available on a set of  $\mathbf{X}_* = (\mathbf{x}_*, \mathbf{y}_*, \mathbf{z}_*) \in \mathbb{R}^{3 \times n_*}$  points and denote these samples as  $\mathbf{U}(\mathbf{X}_*) = \mathbf{U}_* = (\mathbf{u}_*; \mathbf{v}_*; \mathbf{w}_*) \in \mathbb{R}^{3n_*}$  where “;” denotes vertical concatenation. With the basis matrix  $\Phi(\mathbf{X}_*) = \Phi_*$ , the weights minimizing the  $l_2$  norm of the training error are (see for example Hastie et al. (2009); Bishop (2011); Deisenroth et al. (2020)):

$$\mathbf{W} = (\Phi_*^T \Phi_* + \alpha \mathbf{I})^{-1} \Phi_*^T \mathbf{U}_*, \tag{5}$$

where  $\alpha \in \mathbb{R}$  is a regularization parameter and  $\mathbf{I}$  is the identity matrix. The regularization parameter  $\alpha$  ensures that the inversion is possible. Once the weights are computed, the velocity field and its derivatives are available on an arbitrary grid since (2) gives an analytical expression (Sperotto et al. 2022).

### 2.2 From ensembles of RBFs to RBF of the ensemble

Let us consider a statistically stationary and ergodic velocity field  $\mathbf{u}(\mathbf{x})$ . The sample at any location  $\mathbf{x}$  depends on the joint probability density function (pdf)  $f_u(\mathbf{x}, \mathbf{u})$ , so that we can define the mean field from the expectation operator:

$$\langle \mathbf{u} \rangle(\mathbf{x}) = \mathbb{E}\{\mathbf{u}(\mathbf{x})\} = \int_{-\infty}^{\infty} \mathbf{u}(\mathbf{x}) f_u(\mathbf{x}, \mathbf{u}) d\mathbf{u}. \tag{6}$$

The challenge in estimating the mean field in (6) from a set of PTV measurements of the velocity field is that each sample (snapshot) is available on a different set of points. We denote as  $\mathbf{X}^{(j)}$  the set of  $n_p^{(j)}$  points at which the data are available in the  $j$ -th sample of the field (i.e., PTV measurements) and as  $\mathbf{U}^{(j)} = \mathbf{U}(\mathbf{X}^{(j)})$  the associated velocity measurements.

The usual binning-based approach to compute statistics maps the sets of points  $\mathbf{X}^{(j)}$  onto a fixed grid of bins so that all points within the bins can be used to build local statistical estimates. Then, attributing all points within the  $i$ -th bin to a specific position  $\mathbf{x}_i$  allows to remove the spatial dependency of the joint pdf and to move from the expectation operator in (6) to its discrete (sample-based) counterpart. Therefore, at each of the bin locations  $\mathbf{x}_i$  one has:

$$\langle \mathbf{u} \rangle(\mathbf{x}_i) \approx \frac{1}{n_{p,i}} \sum_{j=1}^{n_{p,i}} \mathbf{U}^{(j)}(\mathbf{x}_i), \tag{7}$$

where  $\mathbf{U}^{(j)}(\mathbf{x}_i)$  denotes the mapping of the PTV sample  $\mathbf{U}^{(j)}$  onto the  $i$ -th bin, and  $n_{p,i}$  denotes the number of measurement points available within the bin.

In this work, we propose an alternative path. Introducing the RBF regression (4) into (6) and noticing that the Jacobian is  $d\mathbf{u}/d\mathbf{W} = \Phi(\mathbf{x})$  we have:

$$\begin{aligned} \langle \mathbf{u} \rangle(\mathbf{x}) &= \int_{-\infty}^{\infty} \Phi(\mathbf{x}) \mathbf{W} f_u(\mathbf{x}, \Phi(\mathbf{x})) d\mathbf{u} \\ &= \Phi(\mathbf{x}) \int_{-\infty}^{\infty} \mathbf{W} f_w(\mathbf{W}) d\mathbf{W} = \Phi(\mathbf{x}) \langle \mathbf{W} \rangle, \end{aligned} \tag{8}$$

with  $f_w(\mathbf{W}) = f_u(\mathbf{x}, \mathbf{u}) \Phi(\mathbf{x})$ . The expectation of the weights can be estimated from data more easily than the expectation of the velocity field, because the distribution  $f_w(\mathbf{W})$  does not depend, at least in principle, on the positioning of the data so long as the regression is successful. This implies that the same set of RBFs is used for the regression of all snapshots and that each of these is sufficiently dense to ensure good training.

Assuming that one collects  $n_t$  velocity fields and denoting as  $\mathbf{W}^{(j)}$  the weights of the RBF regression of each of the  $j = [1, \dots, n_t]$  snapshots, one has:

$$\langle \mathbf{W} \rangle \approx \mathbf{W}_A = \frac{1}{n_t} \sum_{j=1}^{n_t} \mathbf{W}^{(j)}, \tag{9}$$

where  $\mathbf{W}^{(j)}$  is the weight obtained when regressing the  $j$ -th snapshot. Introducing (5) is particularly revealing:

$$\langle \mathbf{W} \rangle \approx \frac{1}{n_t} \sum_{j=1}^{n_t} (\Phi_{(j)}^T \Phi_{(j)} + \alpha \mathbf{I})^{-1} \Phi_{(j)}^T \mathbf{u}^{(j)}, \tag{10}$$

where  $\Phi_{(j)} = \Phi(\mathbf{X}^{(j)})$ . All operations in (10) are linear, and some of these can be replaced by operations on the full ensemble of data, which we define as:

$$\mathbf{X}_E = \bigcup_{j=1 \dots n_t} \mathbf{X}_{(j)} \quad \mathbf{U}_E = \bigcup_{j=1 \dots n_t} \mathbf{U}_{(j)}. \tag{11}$$

The ensemble has useful properties. Defining as  $\Phi_E = \Phi(\mathbf{X}_E)$  and expanding the summations in the projections  $\Phi_{(j)}^T \mathbf{U}^{(j)}$  and in the correlations  $\Phi_{(j)}^T \Phi_{(j)}$  from (10), one has:

$$\sum_{j=1}^{n_t} \Phi_{(j)}^T \Phi_{(j)} = \Phi_E^T \Phi_E \in \mathbb{R}^{n_b \times n_b}, \tag{12a}$$

$$\sum_{j=1}^{n_t} \Phi_{(j)}^T \mathbf{U}^{(j)} = \Phi_E^T \mathbf{U}_E \in \mathbb{R}^{n_b \times n_b}. \tag{12b}$$

The goal of the proposed approach is to replace the average of the RBF regression in each snapshot, as requested in (10), with the RBF regression of the ensemble set. This allows replacing  $n_t$  regressions of size  $n_p^{(j)}$  with one single regression of size  $n_{pE}$ . Without aiming for a formal proof, we note that the covariance matrices  $\Phi_{(j)}^T \Phi_{(j)}$  collect the inner products between the bases sampled on the points  $\mathbf{X}^{(j)}$ :

$$\Phi_{(j)}^T \Phi_{(j)}[m, n] = \sum_{j=1}^{n_p^{(j)}} \varphi_m(\mathbf{X}^{(j)}) \varphi_n(\mathbf{X}^{(j)}), \tag{13}$$

and one might expect these to become independent from the specific set  $\mathbf{X}^{(j)}$  at the limit  $n_p^{(j)} \rightarrow \infty$ . The same is true for the inner product  $\Phi_{(j)}^T \mathbf{u}^{(j)}$  in (9).

Therefore, assuming that each snapshot is sufficiently dense, we approximate:

$$\Phi_{(j)}^T \Phi_{(j)} \approx \frac{1}{n_t} \Phi_E^T \Phi_E, \tag{14}$$

and thus use (12b) to write (10) as:

$$\langle \mathbf{W} \rangle \approx \mathbf{W}_E = (\Phi_E^T \Phi_E + \alpha \mathbf{I})^{-1} \Phi_E^T \mathbf{U}_E. \tag{15}$$

With the help of (15), we can therefore compute the mean of a random field through a single RBF regression of the

ensemble of points. The approach uses “meshless” collocation points (see Zhang et al. (2000); Chen et al. (2014); Fornberg and Flyer (2015)) because it does not require the definition of a computational mesh (with nodes, elements and connectivity) to compute derivatives. It is “binless” because the spatial distribution of flow statistics are regressed globally and not computed in local bins.

### 3 Numerical recipes

This section describes the numerical details in the implementation of the RBF regression described in the previous section. Section 3.1 reviews the methods to introduce physics-based constraints while Sect. 3.2 describes the partition of unity method (PUM) to minimize the memory requirements.

#### 3.1 Constrained RBFs

The RBF regression in (5) can be constrained using Lagrange multipliers and the Karush–Kuhn–Tucker (KKT) condition as shown in Sperotto et al. (2022).

The current implementation in SPICY (Sperotto et al. 2024a) allows to set linear constraints and quadratic penalties. These are used to impose or to penalize the violation of linear constraints such as Dirichlet and Neumann conditions, as well as divergence-free or curl-free conditions. Following the notation in (5), the weight vector defining the RBF regression of the data  $(\mathbf{X}_*, \mathbf{U}_*)$  minimizes the following augmented cost function:

$$J^*(\mathbf{w}, \lambda) = \|\mathbf{U}_* - \Phi(\mathbf{X}_*)\mathbf{W}\|_2^2 + \lambda^T (\mathcal{L}(\mathbf{X}_\mathcal{L})\mathbf{W} - \mathbf{c}_\mathcal{L}) + \alpha_\nabla \|\mathbf{D}_\nabla(\mathbf{X}_g)\mathbf{W}\|_2^2. \tag{16}$$

The first term is the least squares error. A minimization solely focused on this term yields the unconstrained solution in (5). The second term is related to hard linear constraints. The vector  $\lambda$  collects the associated Lagrange multipliers: These are additional unknowns to be identified in the constrained regression. The reader is referred to Sperotto et al. (2022) for more details on the shape and formation of these matrices.

The third term in (16) is a quadratic penalty, which in this work is solely used to penalize violations of the divergence-free condition, set on the full set of  $n_g$  points with coordinates  $\mathbf{X}_g$ . The importance of the penalty is controlled by the parameter  $\alpha_\nabla \in \mathbb{R}^+$ . Penalties are soft constraints: They promote but do not enforce a condition and require the a-priori (and not trivial) definition of  $\alpha_\nabla$ . On the other hand, their implementation is computationally much cheaper because penalties bring no new unknowns. The current

implementation allows both constraints and penalties, to offer a compromise between the strength of hard constraints and the limited cost of penalties.

The problem of minimizing (16) can be cast into the problem of solving a linear system of the form:

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{W} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{pmatrix}, \tag{17}$$

where  $\mathbf{A} \in \mathbb{R}^{3n_b \times 3n_b}$  is computed from the training and penalty points,  $\mathbf{B} \in \mathbb{R}^{3n_b \times n_\lambda}$  is computed from the linear constraints,  $\mathbf{b}_1$  is associated with the training data and  $\mathbf{b}_2$  is associated with the constraints. The vector  $\boldsymbol{\lambda} \in \mathbb{R}^{n_\lambda}$  gathers the Lagrange multipliers for which the system must also be solved. The reader is referred to Sperotto et al. (2022) for details on the matrices and efficient numerical methods for the system solution. It is worth stressing that this work solely considered equality constraints (e.g., divergence-free of the mean flow field), although inequality constraints (e.g., positiveness of the Reynolds stresses) could also be included. These require the solution of a quadratic programming problem (Boyd and Vandenberghe 2004; Nocedal and Wright 2006) and are currently under investigation.

In what follows, we introduce the notation  $\tilde{\mathbf{U}}(\mathbf{x}) = \Phi(\mathbf{x})\mathbf{W} = \text{RBF}(\mathbf{U}_*, \mathbf{X}_*)$  to refer to the analytic approximation obtained by solving the constrained regression (17) for the training data  $(\mathbf{X}^*, \mathbf{U}^*)$ .

### 3.2 The partition of unity method (PUM)

An important limitation of the constrained RBF framework is the large memory demand due to the large size and the dense nature of the matrices involved in (17). This problem can be mitigated using compact support bases to make the system sparse and accessible to iterative methods for sparse systems or the partition of unity method (PUM) to divide the problem into smaller blocks and enable direct solvers. We leave a detailed comparison (or possible combination) of the two approaches for future works, and here focus on the second because a preliminary investigation showed that it was faster and generally more accurate.

The PUM was proposed by Melenk and Babuška (1996) in the context of the finite element method, explored for interpolation purposes in Wendland (2002); Cavoretto (2021) and extensively developed by Larsson et al. (2013, 2017); Cavoretto and De Rossi (2019, 2020) for the meshless integration of PDEs. The general idea of RBF-PUM is to split the regression problem in different portions (partitions) of the domain. Different PUM approaches have been proposed; these could be classified into “global” or “local.” A global approach solves one large regression problem (e.g., Larsson et al. (2017)) which is made significantly sparser by the partitioning. A local approach solves many smaller

regression problems (e.g., Marchi and Perracchione (2018)) treating the regression in each portion as independent from the other.

In the context of data assimilation for image velocimetry, the RBF-PUM has been recently used in Li et al. (2021) for smooth gradient computation and in Ratz et al. (2022b) for super-resolution. Recently, the extension of the RBF-PUM to include constraints has been proposed in Li and Pan (2024), following the stable gradient computation by Larsson et al. (2013), and combined with a Lagrangian tracking approach. Our approach differs from Li and Pan (2024)’s in that we use a heuristic treatment of the derivatives at the intersection of the patches, which we found to be more stable.

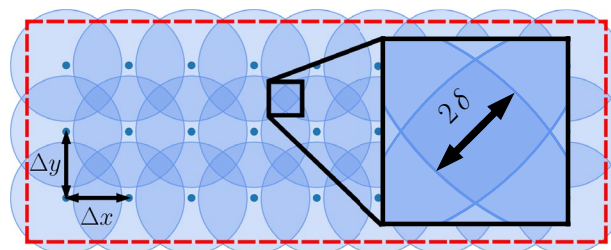
To illustrate the proposed approach, we first briefly recall the PUM with the help of Fig. 1. The measurement domain  $\chi$  is covered by  $M$  spherical patches  $\chi_m$  such that  $\bigcup_{m=1}^M \chi_m \supset \chi$ . In the 2D example of Fig. 1, the rectangular domain (red dashed line) is covered by 27 patches (blue circles) with a regular spacing  $\Delta x$  and  $\Delta y$ . The minimum radius to cover the entire domain is  $r' = \sqrt{\Delta x^2 + \Delta y^2} / \sqrt{2}$ . However, following Larsson et al. (2017), the regression performs better if patches are partially overlapping, that is if the radius  $r'$  is stretched by a factor  $\delta$  to  $r = r'(1 + \delta)$ . This radius is used for every patch  $\chi_m$ . The overlap is visualized in the zoom-in (black solid lines) of Fig. 1.

A weight function  $\Omega_{(m)}$  is assigned to each patch. This function merges the contributions from the overlapping patches and is constructed such that:

$$\sum_{m=1}^M \Omega_{(m)}(\mathbf{x}) = 1, \quad \forall \mathbf{x} \in \chi. \tag{18}$$

The weight functions are generated by applying the method by Shepard (1968) for compactly supported functions, which gives:

$$\Omega_{(m)}(\mathbf{x}) = \frac{\psi_m(\mathbf{x}; c_m)}{\sum_{q=1}^M \psi_q(\mathbf{x}; c_q)}, \tag{19}$$



**Fig. 1** Example of a domain  $\chi$  (red dashed lines) being covered by a total of 27 circular patches (blue circles) on a regularly spaced grid of  $\Delta x, \Delta y$ . The zoom-in (black solid lines) on the right-hand side shows the overlap  $\delta$  between patches

where  $\psi_m(\mathbf{x})$  is a compactly supported generating function, centered on  $c_m$  in the  $m$ -th patch. An example for such a generating function is the Wendland  $C^2$  function (Wendland 1995) which is defined as:

$$\psi_m(\mathbf{x}|c_m, r) = \left(4 \frac{\|\mathbf{x} - c_m\|^2}{r} + 1\right) \left(1 - \frac{\|\mathbf{x} - c_m\|^2}{r}\right)_+^4, \tag{20}$$

where  $r$  is the radius of the function and the subscript  $_+$  is the positive part of a function, i.e.,  $(a)_+ = a$  if  $a > 0$  and  $(a)_+ = 0$  if  $a < 0$ .

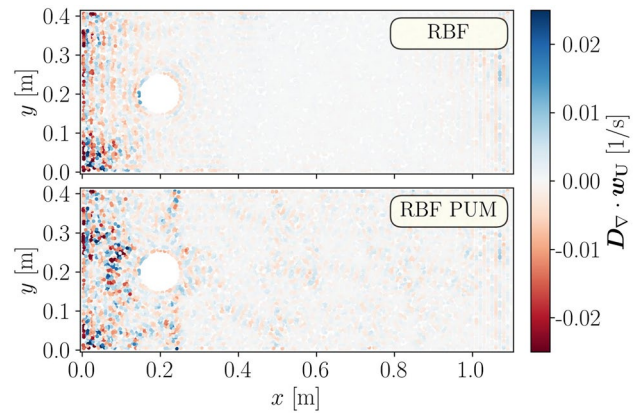
The  $M$  patches are used to identify  $M$  portions of datasets, each contained within the area  $\Omega_{(m)}(\mathbf{x}) \neq 0$  with  $m = 1, \dots, M$ . The partitioning can be interpreted as a partitioning of the linear system (4) and the augmented cost function (16). The partitioning consists in multiplying both the data and the constraints by the local weight function. That is, given the full dataset  $(X_*, U_*)$ , the data used for the local (constrained) regression in patch  $m$  are  $U_{(m)} = \Omega_m(X_{*,m})U_{*,m}$  and the bases used in each patch is  $\Phi_{(m)} = \Omega_m(X_*)\Phi(X_*, X_{c,m})$ , with  $X_{c,m}$  considering only the subset of collocation points inside the  $m$ -th patch. Similarly, all linear constraint operators  $\mathcal{L}(X)$  and their values  $c_{\mathcal{L}}$  in (16) and (17) are weighted by the weight function  $\Omega(X_{\mathcal{L}})$ . Then, each local regression can be carried out solving the local linear system (17) to obtain the local weights  $W_m$ . Finally, given the set of local sets of weights, the analytical expression over the full domain is:

$$\tilde{U}(\mathbf{x}) = \sum_{m=1}^M \Omega_m(\mathbf{x})\Phi(\mathbf{x}, X_{c,m})W_m. \tag{21}$$

To compute derivatives, we use a heuristic treatment that supersedes the product rule and sets all derivatives of the weight functions to zero. Therefore, the partial derivative along  $x$ , for example, reads:

$$\partial_x \tilde{U}(\mathbf{x}) = \sum_{m=1}^M \Omega_m(\mathbf{x})\partial_x \Phi(\mathbf{x}, X_{c,m})W_m. \tag{22}$$

To illustrate the performances of the PUM implementation, we consider the second test case in Sperotto et al. (2022), which is the regression of the flow past a cylinder in laminar conditions. We compare both our local PUM with a classic, global RBF regression. Figure 2 shows the analytical divergence field of the standard RBF regression at the top and the one of RBF-PUM at the bottom. Both use solenoidal and Dirichlet constraints on the boundaries as well as a divergence-free penalty in every training point. The largest differences are at the inlet and close to the cylinder where the gradients are largest. The magnitudes are comparable, and no pattern of the patches is visible. A comparison of the mean flow (not shown here) likewise only shows minor



**Fig. 2** Divergence computed from the analytical RBF representation. Top: constrained, global regression from Sperotto et al. (2022), Bottom: RBF-PUM with locally constrained regressions

differences. The computational time of the RBF-PUM is an order of magnitude shorter. Further gains are possible by solving each of these  $M$  problems in parallel on multiple processors, but we leave these developments to future improvement. In its current implementation, the PUM allowed to process millions of vectors on a modest laptop with 8GB RAM.

## 4 Selected algorithms for benchmarking

### 4.1 Traditional binning approaches

We consider two traditional binning methods, namely the Gaussian weighting by Agüí and Jiménez (1987) and the polynomial fitting by Agüera et al. (2016). These are described in Sects. 4.1.1 and 4.1.2, respectively. These have in common that none of the statistical quantities are expressed as continuous functions. The statistics are only available at the bin’s center, and higher resolution and gradients can only be obtained through further processing. We do not consider the top-hat approach since its shortcomings are well-known (Agüera et al. 2016). While all methods can extract higher-order statistics, we restrict our descriptions to first- and second-order statistics for velocity fields, i.e., the mean flow and Reynolds stresses.

#### 4.1.1 Gaussian weighting

The Gaussian weighting (Agüí and Jiménez 1987) tackles unresolved velocity gradients by weighting points in every bin with a Gaussian. This simple approach gives less impact to points far from the bin center, mitigating the effects of unresolved mean flow gradients. However, weighting

reduces the effective number of samples and thus decreases statistical convergence. In this work, we choose a standard deviation of  $D_b/3$  for the Gaussian weighting functions, with  $D_b$  the bin diameter.

### 4.1.2 Polynomial fitting

The local polynomial fitting of Agüera et al. (2016) fits the ensemble fields within a bin with a polynomial function up to second order, providing a continuous function of the local mean flow. This continuous function is used for two purposes. First, it is evaluated in the bin center to provide the mean velocity in the bin. Second, it is evaluated in all data points within a bin and subtracted to the instantaneous velocities to compute the velocity fluctuations. Higher-order statistics are sampled on the mean-subtracted fields through a top-hat-like approach.

## 4.2 RBF-based approaches

The RBF approaches build on the mathematical background introduced in Sects. 2 and 3, and in particular on the assumption that the expectation operator can be approximated by a regression in space. The framework was implemented with three variants in three algorithms,

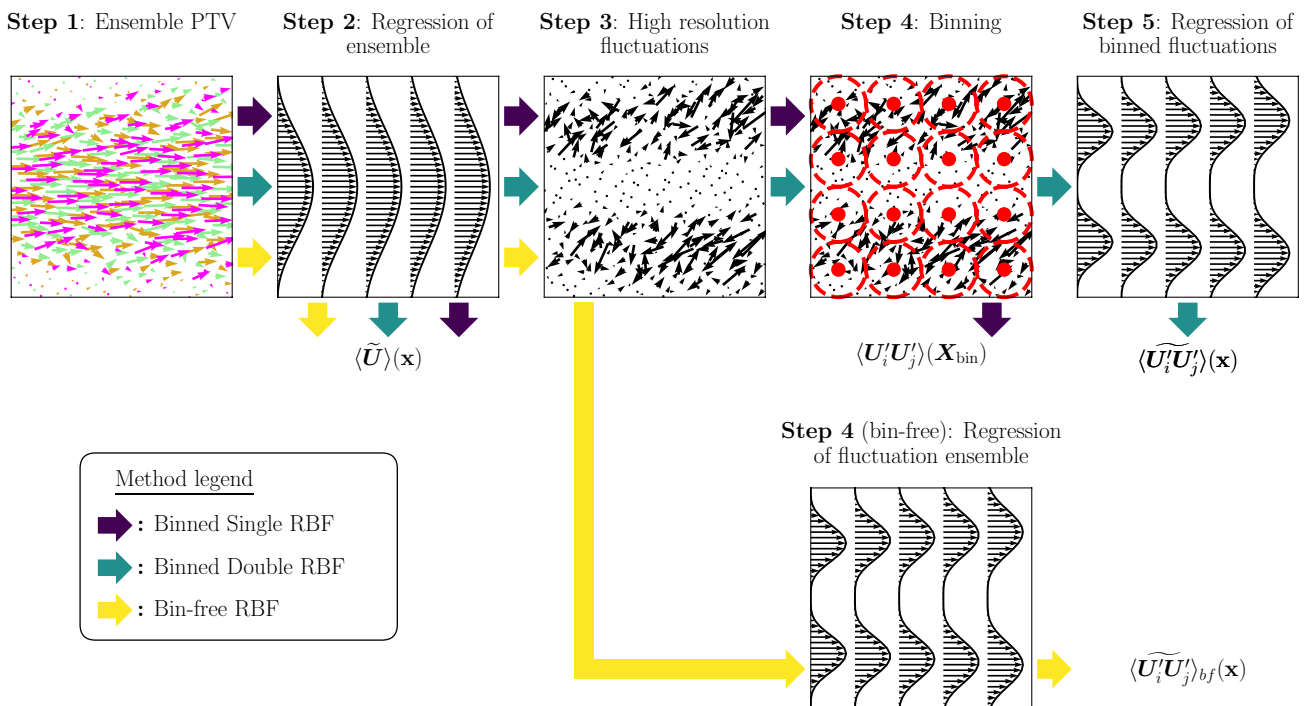
named “Binned Single RBF,” “Binned Double RBF” and “Bin-Free RBF.” These algorithms share several common steps, which are recalled in the flowchart in Fig. 3. The sequence of steps for each method is traced using arrows of different colors, recalled in the legend on the bottom left.

- *Step 1* The starting point for all methods is an ensemble flow field that is assumed to have gathered enough realizations to provide statistical convergence. This is indicated in Fig. 3, using different colors for fields in different snapshots.
- *Step 2* For all methods, the mean flow is computed in the same way using a PUM-based constrained regression RBF of the ensemble. This provides the analytical mean flow field:

$$\langle \tilde{U} \rangle(\mathbf{x}) = \text{RBF}(X_E, U_E). \tag{23}$$

- *Step 3* The function (23) is used to compute the ensemble of velocity fluctuations by subtracting the mean field  $\langle \tilde{U} \rangle(X_E)$  to the ensemble field:

$$U'(\mathbf{x}_E) = U_E - \langle \tilde{U} \rangle(\mathbf{x}_E). \tag{24}$$



**Fig. 3** Flowchart explaining the processing pipeline of the three proposed RBF-based methods. The colors of the arrows correspond to each of the three methods according to the legend. All three methods subtract the global mean field analytically and then extract higher-

order statistics using (1) binning (Binned Single RBFs, purple), (2) binning and RBF regression (Binned Double RBF, teal) or (3) only an RBF regression (Bin-Free RBF, yellow)

This field is then used to compute all the products  $U'_i U'_j(X_E)$ , that are required by all methods in the following steps. This is the last common step for the three methods.

*Binned Single RBF*

- *Step 4.* This method now interrogates the ensemble fields of products  $U'_i U'_j(X_E)$  with a standard binning process. This is the simplest approach and most similar to the one of Agüera et al. (2016), with the only difference being a globally smooth physics-constrained regression instead of a local (locally smooth) polynomial regression. The binning process yields a discrete field of second-order statistics on the binning grid  $X_{bin}$ , i.e.,  $\langle U'_i U'_j \rangle(X_{bin})$ .

*Binned Double RBF*

- *Step 5.* This method builds on the binning grid from Step 4 of Binned Single RBF with a second regression:

$$\langle \widetilde{U'_i U'_j} \rangle(\mathbf{x}) = \text{RBF}(X_{bin}, \langle U'_i U'_j \rangle(X_{bin})). \tag{25}$$

This regression has two purposes. First, it gives an analytical expression for not only the mean but also the Reynolds stresses. Second, it smoothes noisy Reynolds stress fields which occur if the number of samples within a bin is insufficient for convergence. Therefore, fewer samples are required in experiments.

*Bin-Free RBF*

- *Step 4 (Bin-Free).* The bin-free approach deviates from the former two methods after Step 3. This method works on the ensemble fields of products  $U'_i U'_j(X_E)$  without binning, replacing the ensemble operators with the RBF (spatial) regression of the ensemble:

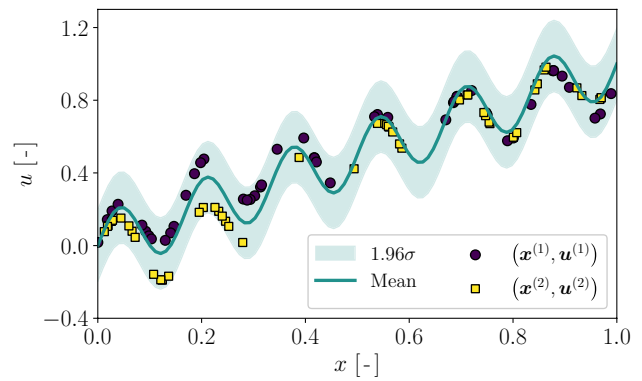
$$\langle \widetilde{U'_i U'_j} \rangle_{bf}(\mathbf{x}) = \text{RBF}(X_E, U'_i U'_j(X_E)), \tag{26}$$

where the subscript “bf” is used to distinguish the output of (26) from the output in (25). The main advantage with respect to the previous approach is to bypass the averaging effects of the binning. However, the computational cost and the complexity of the algorithm are higher, because the number of ensemble points in (26) is larger than the number of bins in (25). Yet, if the same collocation points and shape parameters are reused, computations can be shared for the two successive regressions of Bin-Free RBF.

## 5 Selected test cases

### 5.1 1D Gaussian process

A synthetic 1D test case was designed to illustrate the relevance of the assumption that the average of multiple



**Fig. 4** Test case 1: a 1D Gaussian process. The mean value is shown with a solid line, and the 95% interval is shown by the shaded area. The scattered markers represent two different members of the ensemble  $(x^{(1)}, u^{(1)})$  and  $(x^{(2)}, u^{(2)})$  with  $n_p^{(j)} = 50$

regressions can be approximated by a single regression of the ensemble (see Sect. 2.2).

The 1D dataset is generated by sampling a 1D Gaussian process with average:

$$u(x) = x + \frac{1}{6} \sin\left(\frac{3\pi}{2}x\right), \tag{27}$$

and covariance function:

$$\kappa(x_1, x_2) = \sigma_f \exp(-\gamma(x_2 - x_1)^2), \tag{28}$$

with  $\gamma = 12.5$  and  $\sigma_f = 0.01$ . In a Gaussian process, the covariance function acts as a kernel function measuring the “similarity” between two points.

The domain  $x$  extends from 0 to 1 and total of  $n_E$  ensembles with  $n_s$  samples are sampled from this process. Figure 4 shows two members of the ensembles  $(x^{(1)}, u^{(1)})$  and  $(x^{(2)}, u^{(2)})$  together with the process average and the 95% confidence interval in shaded area. We verify the validity of assumption (15) by varying the size of the ensemble and the sample size.

### 5.2 3D Synthetic turbulent jet

The second synthetic test case is a three-dimensional, jet-like, turbulent velocity field. This is used to compare the proposed RBF-based methods with classic binning approaches on a case for which the ground truth is available. The synthetic test case is set up in the domain  $(x, y, z) \in [-100, 100] \times [-75, 75] \times [-75, 75]$  voxels (vox). Using cylindrical coordinates  $\mathbf{u} = (u_x, u_r, u_\theta)$ , the mean flow has axial component given by:

$$\langle u_x \rangle(x, r) = \frac{U_0}{2} \left[ 1 + \cos\left(\frac{2\pi r}{\lambda(x)}\right) \right], \tag{29}$$

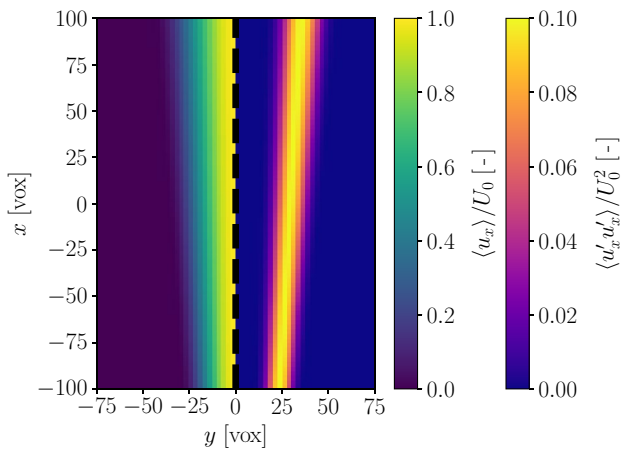
where  $U_0 = 3\text{vox}$  is the maximum displacement,  $r = \sqrt{y^2 + z^2}$  is the radius and  $\lambda(x)$  defines the width of



the profile which increases linearly from 60 to 90 vox. The mean velocity field is zero in the other components, i.e.,  $\langle u_r \rangle = \langle u_\theta \rangle = 0$ . Therefore, this field is not divergence-free and is solely used for demonstration purposes.

Synthetic turbulence is added in a ring with Gaussian noise. The synthetic shear layer is located at  $r = 0.4\lambda(x)$  with a width of  $0.5\lambda(x)$  corresponding to a standard deviation:

$$\sigma_N(x, r) = \frac{3}{2\sqrt{10}} \left[ 1 + \cos\left(\frac{2\pi(r - 0.4\lambda(x))}{0.5\lambda(x)}\right) \right]. \tag{30}$$



**Fig. 5** Test case 2. Exact velocity field of the synthetic jet at  $z = 0$  vox. Axial mean flow (left) and axial normal Reynolds stress (right)

This is used to construct the velocity fluctuations  $u'_x, u'_r$  and  $u'_\theta$  as a multivariate Gaussian  $\mathbf{u}'(\mathbf{x}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \in \mathbb{R}^3$  with mean  $\boldsymbol{\mu}$  and covariance matrix  $\boldsymbol{\Sigma}$  defined as:

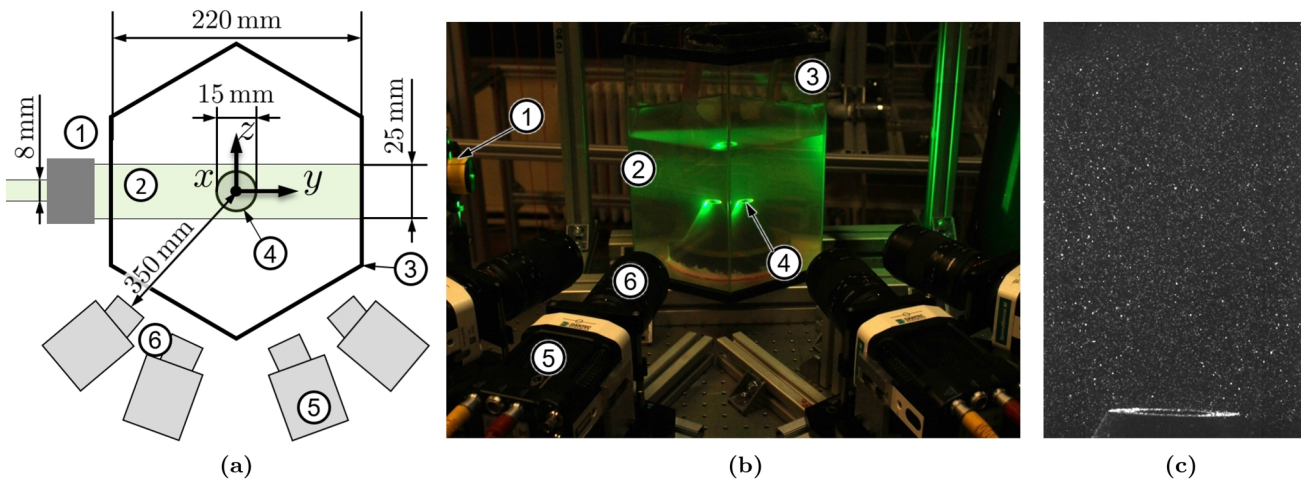
$$\mathbf{u}'(\mathbf{x}) = \begin{pmatrix} u'_x \\ u'_r \\ u'_\theta \end{pmatrix} \sim \mathcal{N} \left( \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{bmatrix} \sigma_N^2 & 0.7\sigma_N^2 & 0 \\ 0.7\sigma_N^2 & \sigma_N^2 & 0 \\ 0 & 0 & \sigma_N^2 \end{bmatrix} \right). \tag{31}$$

That is, the fluctuations  $u'_x$  and  $u'_r$  are correlated while the fluctuation  $u'_\theta$  is not. Figure 5 shows the contour map of the axial mean flow (on the left) and the axial fluctuation  $u'_x$  (on the right).

A total of  $1 \times 10^6$  scattered random points were taken as the velocity field ensemble. We further contaminate these ideal fluctuations by adding uniform noise according to  $\mathbf{u}_n(\mathbf{x}) = \mathbf{u}(\mathbf{x})(1 + \mathbf{q}(\mathbf{x}))$ . Here  $\mathbf{q}(\mathbf{x}) = (q_x(\mathbf{x}), q_y(\mathbf{x}), q_z(\mathbf{x}))$  is a noise vector for each velocity component, where each component is independently sampled from a rectangular distribution in the interval  $[-0.1, 0.1]$ .

### 5.3 3D Experimental turbulent jet

The third test case is a 3D PTV measurement of an underwater jet at the von Karman Institute. The setup of the facility is sketched on the left-hand side of Fig. 6, with a picture of the facility in the center of the Figure. The jet nozzle with a diameter of  $D = 15$  mm was located at the bottom of a hexagonal water tank with a width of 220 mm and a free surface. The nozzle was fixed at the bottom of the tank, and the origin of the coordinate system was set to the center of the nozzle exit. A centrifugal pump was connected to the back of the nozzle with a tube. The effects of the resulting



**Fig. 6** **a** Top-down sketch of the experimental facility with the right-handed coordinate system and **b** image of the facility during the acquisition. Laser light (1) enters the top-hat illumination optics to produce a volumetric illumination (2) which enters the hexagonal tank (3). The illumination is centered above the jet nozzle (4) which

is located at the bottom of the tank. Four high-speed cameras (5) with 100 mm objectives (6) record the jet in an arc that covers approximately  $125^\circ$ ; **c** example of an acquired raw image for the left-most camera

Dean vortices were suppressed by installing a grid with a size of 2 mm inside the nozzle. The inlet length from the grid to the exit of the nozzle was approximately  $4D$  due to spatial constraints. The exit velocity  $U_0$  of the jet was approximately 0.45 m/s, which resulted in a diameter-based Reynolds number of 6750.

The flow was illuminated with a Quantronix Darwin Duo 527-80-M laser with a wavelength of 527 nm and 25 mJ per pulse. The volumetric illumination was achieved with top-hat illumination optics from Dantec Dynamics and entered through the side of the tank. The optics produced a beam with a rectangular cross section with an aspect ratio of 5 : 1, which resulted in an illuminated volume of  $(x \times y \times z) \approx 4.5D \times 2.5D \times 1.5D$ . The resulting scaling factor was approximately 14.7 vox/mm. Red fluorescent microspheres with a diameter ranging from  $4553 \mu\text{m}^3$  and a density of  $1200 \text{ kg/m}^3$  were used as tracer particles. The higher density of the particles allows to vary the seeding concentration by leveraging sedimentation over time. This is particularly helpful for the calibration refinement, which requires much lower seeding concentration (0.005 ppp) than what used during the experiments (0.018 ppp).

The density mismatch was not considered critical to the experiments, since the particles have a terminal velocity of approximately  $u_T = 0.25 \text{ mm/s}$ , that is about a thousandth of the free jet velocity in the free stream. Moreover, the Stokes number was small enough at  $\text{Stk} \approx 5 \times 10^{-3}$  to have tracking errors below 1% Raffel et al. (2018).

Four SpeedSense M310 high-speed cameras with a resolution of  $1280 \times 800 \text{ px}$  were used to observe the flow in the region directly above the jet. The cameras had a distance of approximately 350 mm from the jet center and were arranged in an arc of approximately  $125^\circ$  as shown in Fig. 6. The cameras were equipped with Samyang Macro objectives (F2.8,  $f = 100 \text{ mm}$ ,  $f_\# = 11$ ) and long-pass filters to suppress the reflected laser light. All cameras were used in single-axis Scheimpflug arrangement with an angle of approximately  $3$  and  $12^\circ$  for the interior and exterior cameras, respectively. A total of 2000 time-resolved images were acquired with Dynamic Studio 8.0, at a frequency of 1000 Hz. This corresponds to a maximum displacement of 8 vox for particles in the jet center. An example of an acquired raw image is displayed on the right-hand side of Fig. 6, and Table 1 summarizes the experimental parameters.

The cameras were calibrated with a dotted calibration target (size  $100 \times 100 \text{ mm}^2$ , black dots on white background, diameter 1.5 mm, pitch 2.5 mm). The target was traversed in the range from  $z = \pm 15 \text{ mm}$  throughout the volume by means of a translation stage with micrometric precision. Five images were acquired at equally spaced positions, and a 2nd-degree polynomial in all three axes was used as a calibration model. The resulting calibration error was approximately 0.15 and 0.3 px for the interior and exterior cameras,

**Table 1** Parameters of the experimental setup

Nozzle diameter $D$	15 mm
Central jet velocity $U_0$	0.45 m/s
Reynolds number $\text{Re}_D$	6750
Medium	Water
Camera type	SpeedSense M310
Number of cameras $N_{\text{cam}}$	4
Acquisition frequency $f_{\text{acq}}$	1000 Hz
Camera resolution	$1280 \times 800 \text{ px}$
Camera exposure time $t_{\text{exp}}$	250 $\mu\text{s}$
Scaling factor	14.7 vox/mm
Illum. vol. $(x \times y \times z)$	$67.5 \times 40 \times 24 \text{ mm}^3$ ( $4.5D \times 2.5D \times 1.5D$ )
Lenses	Samyang Macro F2.8/100 mm
Lens aperture $f_\#$	11
Scheimpflug adapter	Single axis
Illumination	Quantronix Darwin Duo 527-80-M laser
Seeding	Fluorescent microspheres
Seeding diameter $d_p$	45–53 $\mu\text{m}$
# tracked particles $n_p$	4000–7000
Seeding density on sensor $N_{\text{ppp}}$	0.018 ppp

respectively. The calibration error was reduced using the procedure outlined by Brückner et al. (2020). For this, a total of 21 statistically independent images were recorded at a seeding concentration of approximately 0.005 ppp. After calibration refinement, the error of every camera was below 0.05 px.

The acquired images were processed with a mean subtraction over all images for each camera. Residual background noise was eliminated by clipping all pixels with an intensity below 60 counts. For each time step, the 3D voxel volume was reconstructed in a domain of approximately  $L_x \times L_y \times L_z = 990 \times 550 \times 285$  in  $x$ ,  $y$  and  $z$  using up to 10 iterations of the SMART algorithm (Atkinson and Soria 2009; Scarano 2013).

For the given parameters, the fraction of ghost particles can be estimated according to Discetti and Astarita (2014):

$$\frac{N_{\text{ghost}}}{N_{\text{true}}} = N_{\text{ppp}} d_\tau L_z (1 - e^{-N_s})^{N_{\text{cam}} - 2}, \quad (32)$$

where  $d_\tau = 2.5 \text{ px}$  was the particle image diameter and  $N_s = N_{\text{ppp}} \pi d_\tau^2 / 4$  the source density. It is important to highlight that the volume was not reconstructed in the full illumination depth of  $1.5D$ , but was reduced to  $1.3D$  because of reduced intensity in the outer regions. The resulting 10% of ghost particles are treated through time-resolved information with predictors based on previous time steps. This increases the accuracy (Malik et al. 1993; Cierpka et al. 2013) and

allows to filter ghost particles which typically have a short track length (Kitzhofer et al. 2009).

After filtering out particles with a track length below 5 time steps, a total of 4000–7000 vectors were computed at each snapshot. Three additional processing steps were applied. First, a normalized median test was used to remove outliers (Westerweel and Scarano 2005). Second, the domain depth was reduced to 1.1*D* because of an insufficient number of particles in the outer region, which negatively affected the RBF regression. Third, we only used data from every third time step, since this provides sufficient statistical convergence and a sufficient level of statistical independence of the snapshots in the shear layer. The resulting dataset consists of  $3.35 \times 10^6$  particles in the ensemble used for the training.

## 6 Results

### 6.1 A 1D Gaussian process

The main purpose of this illustrative test case was to compare the average of RBF regressions in (9) with the RBF regression of the ensemble in (15). In both cases, we use 25 evenly spaced RBFs with a radius of 0.06, which is defined as the distance at which the RBF reaches half its value. These values are chosen to sufficiently cover the domain and have a well-posed regression for the lowest seeding case. However, the lack of points leads to ill-conditioned matrices, and thus, a strong regularization is needed. The regularization parameter  $\alpha$  in (5) was computed by setting an upper limit to the condition number  $\kappa(\mathbf{H})$  of the matrix  $\mathbf{H} = (\Phi_*^T \Phi_*)$  estimated as follows

$$\kappa(\mathbf{H}) \approx \frac{\lambda_M}{\alpha} \rightarrow \alpha = \frac{\lambda_M(\mathbf{H})}{\kappa_L}, \tag{33}$$

with  $\lambda_M(\mathbf{H})$  the largest eigenvalue of  $\mathbf{H}$  and  $\kappa_L = 10^4$  the upper limit of the condition number. This regularization

approach is used in all regressions in the remainder of this article, each with different values of  $\kappa_L$ .

We consider a set of  $n_{pE}$  samples in the ensemble, varying from  $n_{pE} = 10^3$  to  $10^7$ . To compute the average of RBFs in (9), we assume that the “snapshots” from which each regression is carried out consists of  $n_p$  samples, taken as  $n_p = \{50, 100, 150, 200, 250\}$ . Therefore, the number of regressions is  $n_t = n_{pE}/n_p$ . One could either work with many sparse snapshots (small  $n_p$  and large  $n_t$ ) or fewer dense snapshots (large  $n_p$  and small  $n_t$ ), but for the comparison with the ensemble approach, the same  $n_{pE}$  is kept for all cases. The points are randomly sampled using a uniform distribution.

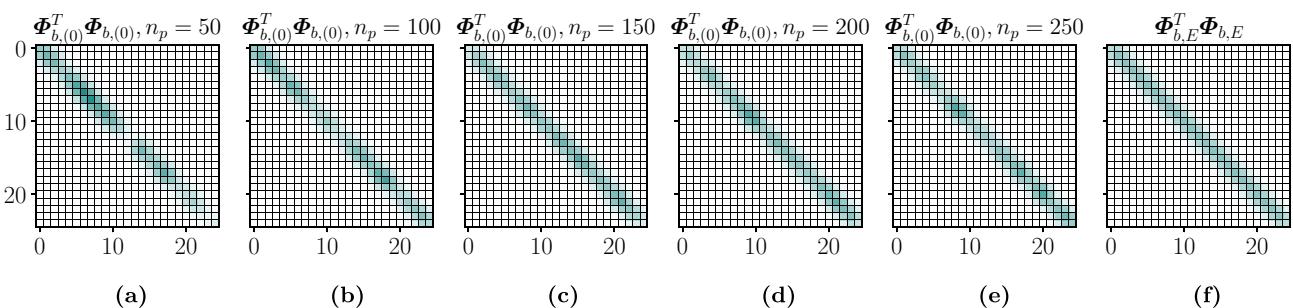
For each snapshot  $i$ , the regression evaluates the basis matrix  $\Phi_{b,(i)}$  and computes the set of weights  $\mathbf{w}^{(i)}$  using the unconstrained RBF regression in (5), i.e.,  $\mathbf{w}^{(i)} = \text{RBF}(\mathbf{x}^{(i)}, \mathbf{u}^{(i)})$ .

Figure 7 compares the matrices  $\Phi_{b,(0)}^T \Phi_{b,(0)}$  for snapshots with  $n_p = \{50, 100, 150, 200, 250\}$  particles each together with the case using the full ensemble of points with  $n_{pE} = 10^7$ . As expected, all matrices have a diagonal band proportional to the width of the RBFs. This is particularly smooth for the ensemble and shows “holes” for the sample matrices, which becomes more pronounced as  $n_p$  is reduced. This is due to the uneven and overly sparse distribution of points in each sample. However, for sufficiently dense snapshots, it is clear that the all inner product matrices  $\Phi_b^T \Phi_b$  converge to a prescribed function. This is the essence of the shift in paradigm from the ensemble averaging of regressions to the regression of the ensemble dataset  $\mathbf{w}_E = \text{RBF}(\mathbf{x}_E, \mathbf{u}_E)$ , with  $(\mathbf{x}_E, \mathbf{u}_E)$  the ensemble dataset.

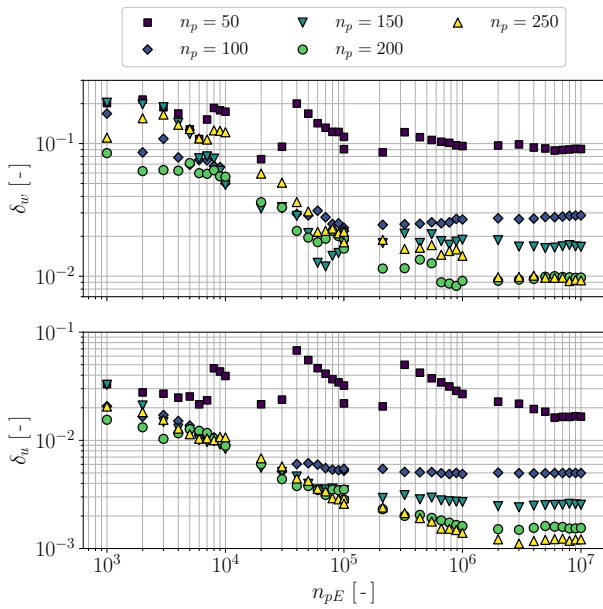
To analyze the impact of the sampling on the comparison between (9) and (15), we define the  $l_2$  discrepancy between the weights and the predictions as

$$\delta_w = \frac{\|\mathbf{w}_A - \mathbf{w}_E\|_2}{\|\mathbf{w}_A\|_2}, \tag{34a}$$

$$\delta_{\langle u \rangle} = \frac{\|\langle \tilde{\mathbf{u}} \rangle_A - \langle \tilde{\mathbf{u}} \rangle_E\|_2}{\|\langle \tilde{\mathbf{u}} \rangle_A\|_2}, \tag{34b}$$



**Fig. 7** Test case 1: matrix structure of  $\Phi_b^T \Phi_b$ . Subfigures a–e show the matrix for the five different sample sizes and subfigure f shows the matrix for the ensemble of points



**Fig. 8** Test case 1: convergence of the relative difference between the RBF regression of the ensemble and the ensemble of the RBF regressions. Difference in the weights (top) and the mean (bottom)

where  $\mathbf{w}_A = \sum_{i=1}^{n_i} \mathbf{w}^{(i)} / n_i$ ,  $\langle \tilde{\mathbf{u}} \rangle_A = \sum_{i=0}^{n_i} \langle \tilde{\mathbf{u}}^{(i)} \rangle / n_i$ , with  $\langle \tilde{\mathbf{u}}^{(i)} \rangle = \Phi_b(\mathbf{x}^{(i)}) \mathbf{w}^{(i)}$  and  $\langle \mathbf{u} \rangle_E = \Phi_b(\mathbf{x}_E) \mathbf{w}_E$ .

Figure 8 plots  $\delta_w$  and  $\delta_u$  in (34a and 34b) as a function of the number of samples in the ensemble ( $n_{pE}$ ) for the five choices of samples per snapshot  $n_p$ . The results show that  $n_p = 50$  is clearly insufficient for the problem at hand. This is due to the fact that (14) does not hold for most of the samples and an average of poor regressions is a poor regression. However, as  $n_p$  increases, convergence is observed with both  $\delta$  dropping smoothly below 1% for  $n_{pE} > 10^4$  regardless of  $n_p$ . Moreover, this comparison shows that the discrepancies on the weight vectors are attenuated in the approximated solution. Although these results depend on the settings of the RBF regression, and in particular on the level of regularization, these results give a practical demonstration on the feasibility of approximating (9) with (15).

### 6.2 3D Synthetic turbulent jet

The purpose of this test case was to compare and benchmark the methods discussed in Sect. 4 on a 3D dataset for which the ground truth is available. We use 121,500 pseudo-random Halton points as collocation points (see Fasshauer (2007) for a discussion on random collocation in meshless RBF methods). This gives approximately 8 particles per basis, in line with the optimal densities identified in Sperotto et al. (2022)). The RBFs use a fixed radius of 30 vox. The PUM used 175 regularly spaced patches with an overlap

of  $\delta = 0.25$ , and no physical constraints were imposed. All methods with binning use spherical bins of different diameters  $D_b$ , also spaced on a regular grid. The Reynolds stress regression has the same RBF and patch placement as the mean flow regression. The RBF processing parameters are summarized in Table 2. All the bin-based approaches use the same binning with size  $D_b$  while the Gaussian weighting has a size of  $\sigma = D_b/3$ . All five methods are compared on the binning grid.

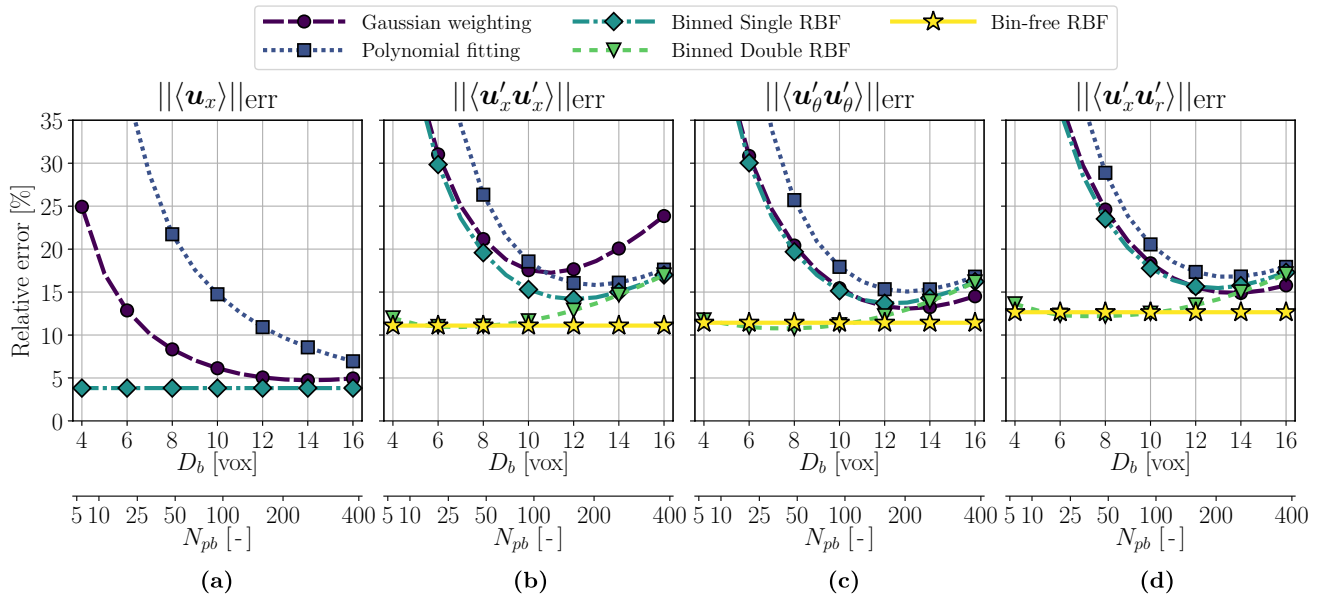
Figure 9 shows the errors for different statistics defined as:

$$\| \mathbf{u} \|_{\text{err}} = \frac{|| \langle \tilde{\mathbf{u}} \rangle - \mathbf{u}_{gt} ||}{|| \mathbf{u}_{gt} ||}, \tag{35}$$

where  $\mathbf{u}$  is either a mean or Reynolds stress and  $\mathbf{u}_{gt}$  the corresponding ground truth.

Figure 9a shows the resulting errors over the binning diameter  $D_b$  of the axial mean flow  $\langle \mathbf{u}_x \rangle (X_{\text{bin}})$ . The abscissa shows the bin diameter and the average number of particles  $N_{pb}$  in each bin. All three RBF-based methods use the same, single regression for the mean which is why they are displayed as one curve. The curve is constant since the regression of the ensemble does not use any binning. For small bin sizes, the error of the Gaussian weighting and polynomial fitting quickly exceeds 15%, although the former has a consistently smaller error. This is because of the small number of points within the bin which are insufficient for averaging and local fitting. At the maximum bin size of 16 vox, the Gaussian weighting reaches an error comparable to the error of the RBF regression whereas the polynomial fitting reaches a minimum of only 8%. This is due to the small gradients in the mean flow; for stronger gradients, the spatial low-pass filtering due to larger bin sizes leads to increased error.

For the Reynolds stresses, the low-pass filtering due to the binning is more evident. The errors on the stresses  $\langle \mathbf{u}'_x \mathbf{u}'_x \rangle (X_{\text{bin}})$ ,  $\langle \mathbf{u}'_\theta \mathbf{u}'_\theta \rangle (X_{\text{bin}})$  and  $\langle \mathbf{u}'_x \mathbf{u}'_r \rangle (X_{\text{bin}})$  are shown in subfigures (b)–(d). For the axial Reynolds stress, the spatial inhomogeneities lead to increased errors for larger bins for all methods except the Bin-Free RBFs. For the axial normal stress, the effects of unresolved mean flow gradients become apparent as the error of the Gaussian weighting strongly increases for large bin sizes. For the other stresses, the mean flow gradients are not as impactful and the weighting mitigates the spatial inhomogeneities. The error trends for the polynomial fitting, Binned Single and Binned Double RBF collapse for bin sized above 12 vox, because there are no convergence problems and the method of mean subtraction has little influence. However, for bin sizes below 10 vox, the error quickly reaches values above 15% because of poor statistical convergence. For the Binned Double RBF, the unconverged Reynolds stresses are smoothed, preserving the



**Fig. 9** Test case 2. Comparison of the errors for different binning diameters. The five curves correspond to the Gaussian weighting (●, Agüí and Jiménez (1987)), local polynomial fitting (■ Agüera et al. (2016)), Binned Single RBF (◆), Binned Double RBF (▼) and Bin-

Free RBF (★). The error norms are the same as the ones defined in Eq. (34a), and in the first figure, the Binned Double and Bin-Free RBF have the same line as Binned Single RBF

**Table 2** RBF parameters of all three methods for the 3D synthetic jet with Reynolds stress computation

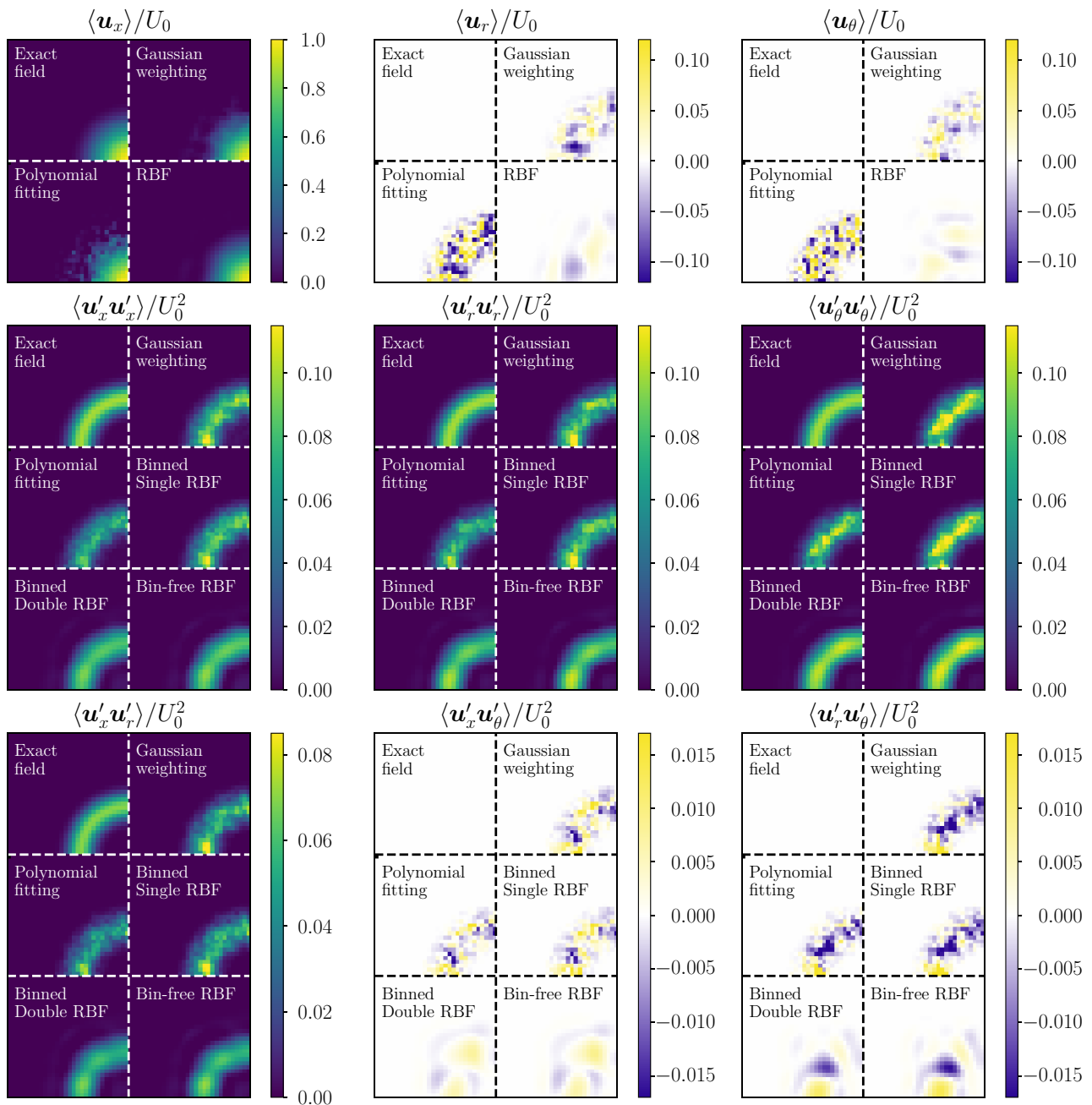
Regression of mean flow	
Number of training points $n_*$	$1 \times 10^6$ (scattered)
Binning diameter $D_b$	4–16 vox
Number of points per bin $N_{p,b}$	7–477
Number of RBFs $n_b$	121,500 (Halton)
RBF radius	30 vox
Condition number $\kappa_L$	$10^{12}$
Number of patches $M$ ( $x \times y \times z$ )	175 ( $7 \times 5 \times 5$ )
Overlap $\delta$	0.25
Noise level $q$	Uniform, 10%
Regression of Reynolds stresses	
Number of training points $n_*$	288,000 ( $80 \times 60 \times 60$ )
Binned Double RBF ( $x \times y \times z$ )	$1 \times 10^6$ (scattered)
Bin-Free RBF	$1 \times 10^6$ (scattered)
Binning diameter $D_b$	4–16 vox
Number of points per bin $N_{p,b}$	7–477
RBF placement	Regular
Number of RBFs $n_b$	121,500 (Halton)
RBF radius	30 vox
Condition number $\kappa_L$	$10^{12}$
Number of patches $M$ ( $x \times y \times z$ )	175 ( $7 \times 5 \times 5$ )
Overlap $\delta$	0.25

error between 11 and 13% for all bin sizes between  $D_b = 4$ –10 vox.

The Bin-Free RBF outperforms all methods with a constant error of 11%, which is the best error achieved by the Binned Double RBF. The fact that the Binned Double RBF converges to the Bin-Free RBF at small binning diameters is not surprising, considering that both approximate local statistics. For small diameters (around 8 vox), the binning only produces a poor approximation of the local statistics and the subsequent regression yields a strong improvement.

A very similar trend is visible in the tangential Reynolds stress in the third column of Fig. 9. The errors for the polynomial fitting and all RBF-based methods appear almost identical to the axial stress. In comparison, the Gaussian weighting reaches its smallest value for the largest bin size. This is because there is no unresolved mean flow gradient which affects the Gaussian weighting. The weight again mitigates spatial inhomogeneities but the error is still 2% larger than the smallest value of Binned Double and Bin-Free RBF. The correlation between the radial and axial component  $u'_x u'_r$  in the fourth column has the same trend as the tangential Reynolds stress. However, all errors are slightly increased by about 2% w.r.t. the other two stresses. The exact reason for this is not known. Yet, the correlation is equally well recovered by all five methods and none of them shows additional advantages in this case.

For  $D_b = 10$  vox, Fig. 10 shows a slice through the jet at  $x = 2.5$  vox. The first, second and third row contain the mean



**Fig. 10** Test case 2. Resulting fields of the slice at  $x = 0$  vox for  $D_b = 10$  vox. Top row: Mean velocity fields. Middle row: Normal stress fields. Bottom row: Shear stress fields. The four panels of the mean fields show from top left to bottom right: The analytical solution, the Gaussian weighting (Agüí and Jiménez

(1987),  $\sigma = D_b/0.33$ ), local polynomial fitting (Agüera et al. 2016) and the solution of the RBF regression. For the Reynolds stresses, there are three panels for the RBFs corresponding to the three different algorithms outlined in Sect. 4.2

flow, normal stress and shear stress, respectively. The sub-figures of the mean flow contain four panels which are from top left to bottom right: the field of the analytical solution, the Gaussian weighting (Agüí and Jiménez 1987), the local polynomial fitting (Agüera et al. 2016) and the mean from the RBF regression. The results of all three mean flow components are similar between all methods. The shape of the

mean profile is recovered well, and the fields appear slightly noisy in the regions of high shear. As expected from the error curves in Fig. 9, the polynomial fitting and Gaussian weighting appear more noisy than the RBF regression. Furthermore, the spikes of the former two methods are random, whereas the RBF regression yields a globally smooth expression.

The subfigures of the Reynolds stresses additionally contain two panels at the bottom showing the result of the Binned Double and Bin-Free RBF method. The effects of not subtracting the local mean are evident in the second row, which shows the three normal stresses. In the core of the jet, in the bottom right region of the panel, the Gaussian weighting has a nonzero axial normal stress  $\langle u'_x u'_x \rangle$  in regions where it should be zero. We attribute this to mean flow gradients within the bin. The other four methods are not affected by this. Moreover, we again highlight the smoothing properties of the second RBF regression observed in the contours of the normal stresses obtained by the Binned Double and Bin-Free RBF.

The same observations hold for the shear stresses in the bottom row of the figure. All methods recover the correlation well. The Gaussian weighting is most severely affected by convergence issues whereas the top-hat approach, polynomial fitting and Binned Single RBF have almost the same shear stress fields.

To conclude this section, the methods based on two successive RBF regressions perform the best for the analyzed test case. For small binning diameters, Binned Double RBF and Bin-Free RBF yield almost the same result as the binning only introduces a slight modulation. Besides the lowest error, the RBF regressions also give continuous expressions of the statistics which enables super-resolution and analytical gradients for all Reynolds stresses.

### 6.3 3D Experimental turbulent jet

The regression of the mean flow field was done with  $n_b = 77175$  RBFs, placed with pseudo-random Halton points as in the previous test case. Considering the measurement volume of  $V \approx 4000 \text{ mm}^3$ , this yields an RBF density of  $\rho_b = n_b/V \approx 1.9$  bases per  $\text{mm}^3$ . For a uniform distribution of points, using geometric probability one could thus estimate an expected average distance of  $\mathbb{E} = (4/3)^{1/3} \rho_b^{-1/3} \approx 0.9 \text{ mm}$  between bases, enabling sufficient overlapping if these have a radius of  $0.5D$ . Divergence-free constraints were imposed in 14,130 points on the outer hull of the measurement domain, and a penalty of  $\alpha_v = 1$  was applied in the whole flow domain. The imposed constraints do not significantly impact the  $l_2$  norm of the error, but allows for better derivatives and improve the computation of derived quantities such as pressure (Sperotto et al. 2022). In total, 1300 patches were used for the PUM, again with an overlap of  $\delta = 0.25$ . For the computation of the Reynolds stresses, 760,725 bins with a diameter of  $0.15D$  were placed on a regular grid of  $161 \times 105 \times 45$  points in  $x \times y \times z$ . This yielded an average of 65 vectors within each bin. The second

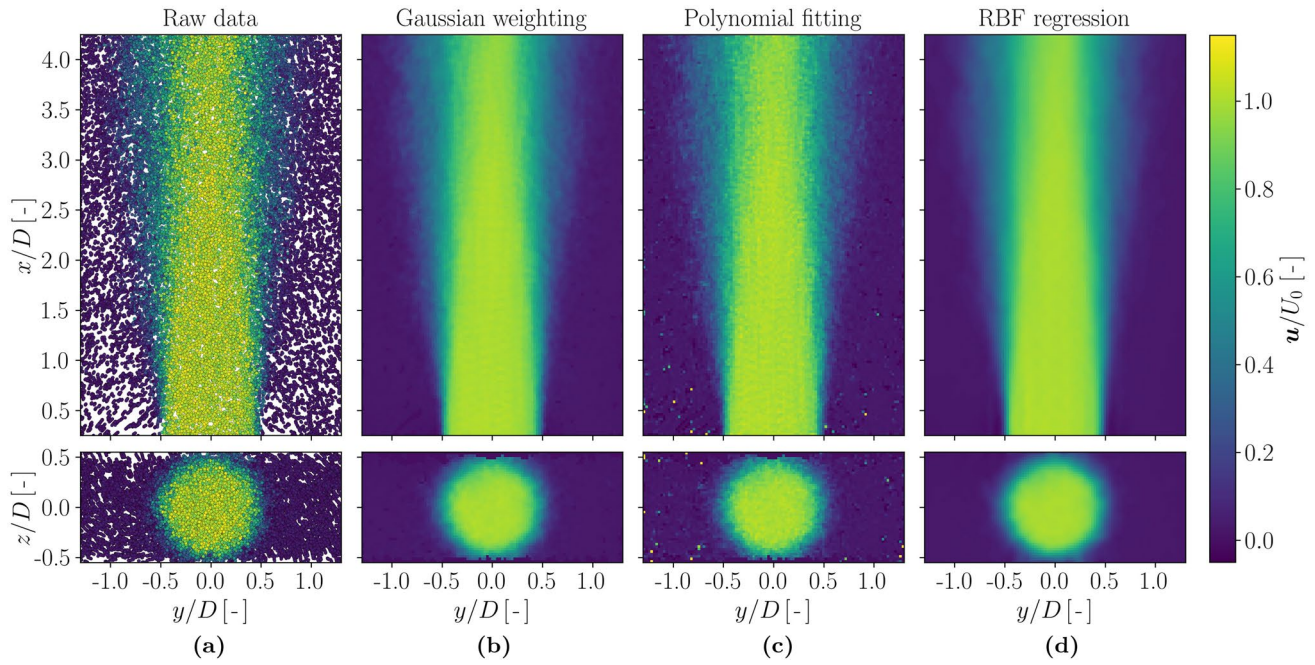
regression reused the same basic RBF and PUM settings. All processing parameters are summarized in Table 3.

Figure 11 shows slices of the velocity field from the PTV data  $\mathbf{u}/U_0$  and the computed mean  $\langle \mathbf{u} \rangle / U_0$  for each algorithm. The slices are, respectively, taken from two planes at  $z/D = 0$  and  $x/D = 2$ . The raw data in a thin volume around the slice are shown as a scatter plot in subfigure (a) while subfigures (b)–(d) show the velocity on the binning grid. All three methods capture the spreading of the symmetric jet well although the RBF regression appears smoother, particularly in the shear layer. The horizontal slice at  $z/D = 0$  further confirms this lack of convergence as the bins on the domain boundary are particularly noisy. In contrast, the RBF solution shows a smooth behavior, as the divergence-free flow acts as a regularization which prevents sharp, noisy spikes.

Figure 12 shows two mean velocity profiles, extracted at  $z/D = 0, x/D = 2$  (left) and  $z/D = 0, x/D = 3$  (right). It can be very well seen that the profiles for all three methods almost collapse. The profiles are not symmetric around the central axis but this asymmetry is equal between all methods, so we attribute it to the jet facility and not the methods. The RBF method yields the best performance in the aforementioned regions of low particle seeding. While the other two methods produce spikes in the mean flow due to problems in the statistical convergence, the RBFs yield a smooth profile of the axial mean velocity.

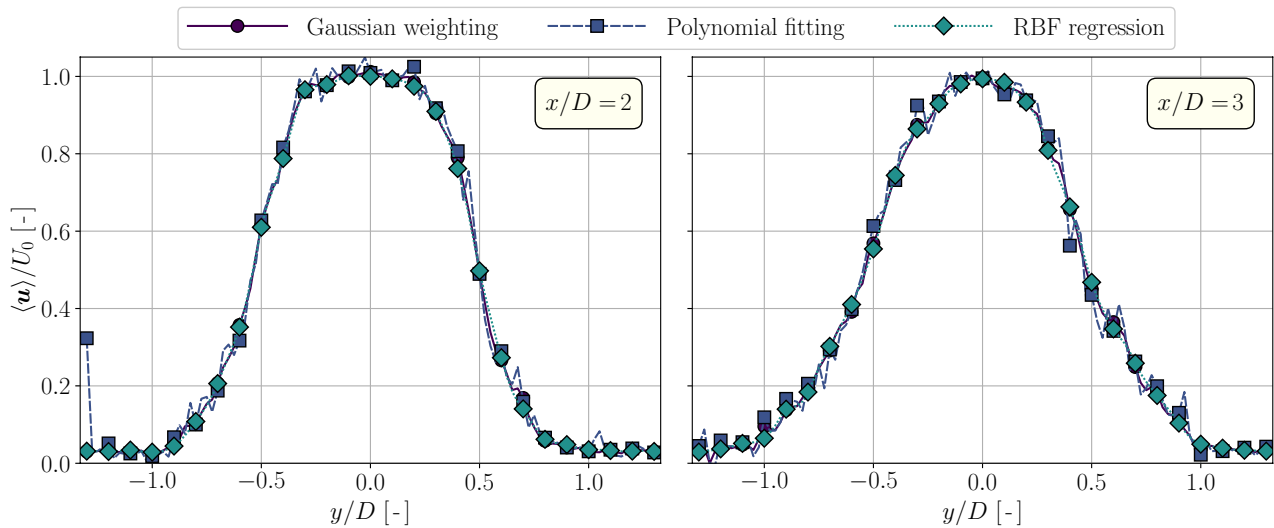
**Table 3** RBF parameters of all three methods for the 3D experimental jet with Reynolds stress computation

Regression of mean flow	
Number of training points $n_p$	$3.35 \times 10^6$ (scattered)
Number of RBFs $n_b$	77,175 (Halton)
RBF radius	$0.5D$ (110 vox)
Number of solenoidal constraints $n_v(x \times y \times z)$ , outer hull	14130 ( $81 \times 53 \times 23$ )
Divergence penalty $\alpha_v$	1
Condition number $\kappa_L$	$10^{12}$
Number of patches $M(x \times y \times z)$	1300 ( $20 \times 13 \times 5$ )
Overlap $\delta$	0.25
Regression of Reynolds stresses	
Number of training points $n_p$	
Binned Double RBF	760725 ( $161 \times 105 \times 45$ )
Bin-Free RBF	$3.35 \times 10^6$ (scattered)
Number of RBFs $n_b$	77175 (Halton)
RBF radius	$0.5D$ (110 vox)
Binning diameter $D_b$	$0.15D$ (44 vox)
Condition number $\kappa_L$	$10^{12}$
Number of patches $M(x \times y \times z)$	1300 ( $20 \times 13 \times 5$ )
Overlap $\delta$	0.25



**Fig. 11** Test case 3. Resulting fields of the axial flow component in a slice through the regression volume. Top row: vertical slice at  $z/D = 0$ . Bottom row: horizontal slice at  $x/D = 2$ . Subfigure **a** shows the scattered training data  $\mathbf{u}_*$  in a thin volume around the slice and

subfigures **b–d**, respectively, show the mean field  $\langle \mathbf{u} \rangle / U_0$  from: the Gaussian weighting (Agüí and Jiménez (1987),  $\sigma = D_b/0.33$ ), local polynomial fitting (Agüera et al. 2016) and the RBF regression of the ensemble



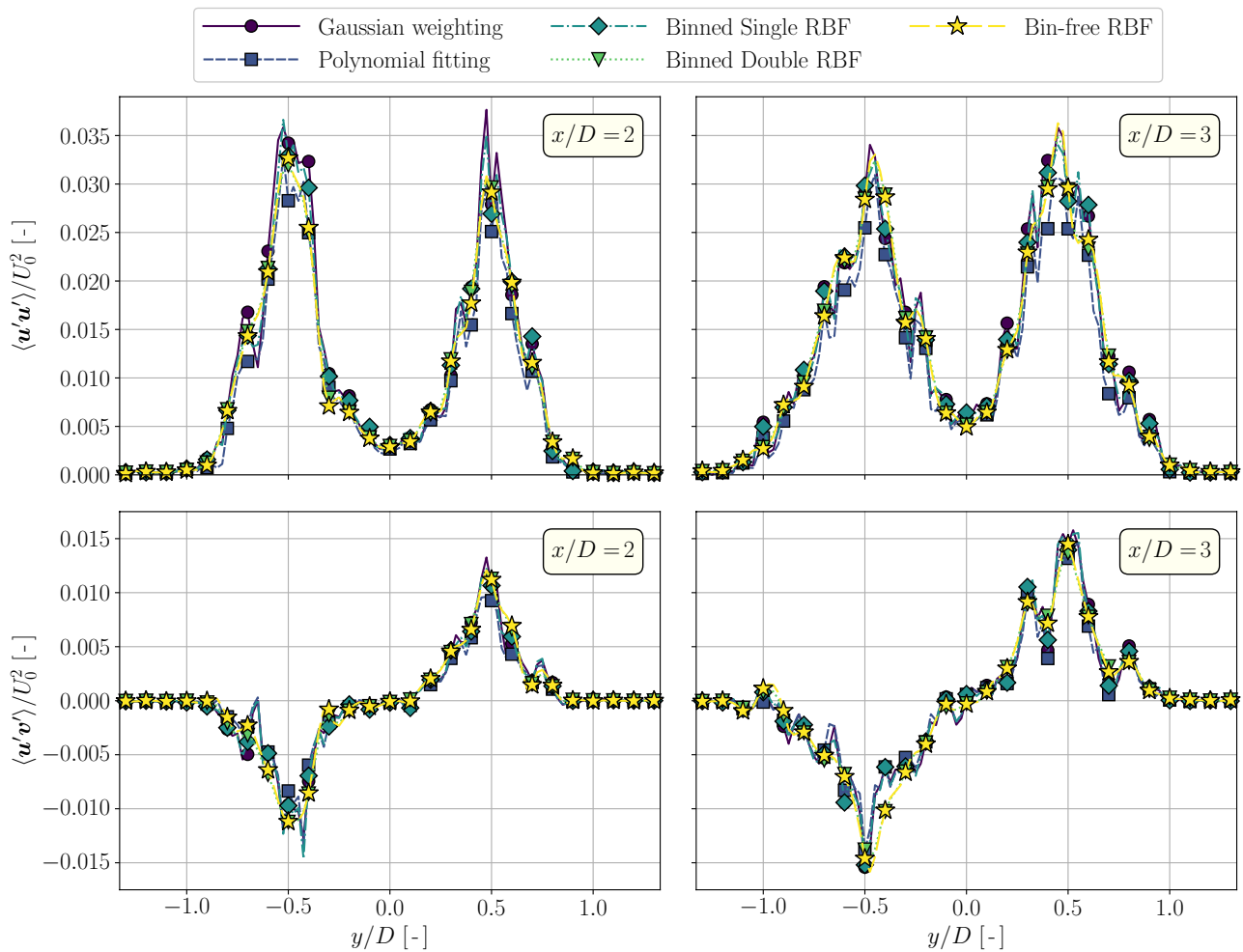
**Fig. 12** Test case 3. Resulting mean velocity profile  $\langle \mathbf{u} \rangle / U_0$  extracted at  $z/D = 0$  and  $x/D = 2$  (left) and  $x/D = 3$  (right). The three curves correspond to the Gaussian weighting (●, Agüí and Jiménez (1987)),

local polynomial fitting (■, Agüera et al. (2016)) and the RBF regression of the ensemble (◆)

The Reynolds stress profiles in Fig. 13 show the same characteristics as the mean flow. We show the normal stress  $\langle \mathbf{u}'\mathbf{u}' \rangle / U_0^2$  and the shear stress  $\langle \mathbf{u}'\mathbf{v}' \rangle / U_0^2$  in the top and bottom row, respectively. All methods give results which agree with theoretical expectations: The stresses

are largest in the shear layer and expanding with the jet. Furthermore, the normal stress is an even function while the shear stress is an odd function. Yet, the Reynolds stresses appear more noisy than the mean flow, as convergence is slower for higher-order statistics. This is





**Fig. 13** Test case 3. Resulting Reynolds stress profile  $\langle u'u' \rangle / U_0^2$  (top) and  $\langle u'v' \rangle$  (bottom) extracted at  $z/D = 0$  and  $x/D = 2$  (left) and  $x/D = 3$  (right). The five curves correspond to the Gaussian weighting (●, Agüí and Jiménez (1987)), local polynomial fitting (■, Agüera

et al. (2016)), Binned Single RBF (◆), Binned Double RBF (▼) and Bin-Free RBF (★)

particularly visible in the Gaussian weighting approach, which shows significant spikes in each of the four sub-figures with differences in the peak amplitude compared to the other methods. In contrast, polynomial fitting and Binned Single RBF have almost the same curve. The lack of convergence is mainly responsible for the non-smooth profile rather than the specific method of mean subtraction. The Binned Double RBF and Bin-Free RBF yield smoother curves compared to the other methods but still struggle in specific areas, like  $(x/D, y/D) = (3, 0.35)$  for  $\langle u'v' \rangle$  where the profiles have an unexpected kink. Yet, this kink is also visible for all other methods and likely stems from unfiltered outliers or a general lack of points in this region.

To conclude, the two successive RBF regressions give the best results also for the experimental test case. In regions with sparse or noisy data, the regularization yields a smooth solution and matches the binning-based approaches in all other regions.

### 7 Conclusions and perspectives

We propose a meshless and binless method to compute statistics in turbulent flows in ensemble particle tracking velocimetry (EPTV). We use radial basis functions (RBFs) to obtain a continuous expression for first- and second-order moments. We showed through simple derivations that an RBF regression of a statistical field is

equivalent to performing spatial averaging in bins. We expanded this idea and showed averaging the weights from multiple regressions can be approximated with a single, large regression of the ensemble of points. The test case of a 1D Gaussian process served as numerical evidence to prove the convergence of the weights and the solution. The resulting matrix is very large, and the computational cost of inverting is prohibitive. Therefore, we employ the partition of unity method (PUM) and the RBFs to reduce the computational cost significantly. Together, both approaches result in analytical statistics at a low cost, even for large-scale problems.

We proposed three different RBF-based approaches and compared them with existing methods, namely Gaussian weighting (Agüí and Jiménez 1987) and local polynomial fitting (Agüera et al. 2016). The proposed methods range from simple ideas based on existing literature (Agüera et al. 2016) to a fully mesh- and bin-free method which uses two successive RBF regressions. On a synthetic test case, the RBF-based methods outperformed the methods from existing literature in both first- and second-order statistics, with the bin-free method having the lowest error. Therefore, besides giving an analytical expression, the bin-free methods also require less data for convergence, which is highly relevant for experimental campaigns.

The same conclusions hold on experimental data, with the RBF approaches producing the best results. All methods show a qualitative agreement with literature expectations with the binning-based approaches having more noise. Insufficient convergence within a bin results in spikes, whereas the methods with a second regression yield a smooth curve with almost no outliers. Therefore, the two successive regressions have the double merit of providing smooth and noise-free analytical regression that can be used for super-resolution of the flow statistics.

Ongoing work focuses on integrating the pressure Poisson equation in the Reynolds-averaged Navier–Stokes framework to obtain the mean pressure field. This can be done with a mesh-free integration following the initial velocity regression, or by coupling both steps in a nonlinear method.

**Acknowledgements** The authors kindly acknowledge Alessia Simonini for her help in setting up and conducting the experiments. The authors also acknowledge David Hess from Dantec Dynamics for his support in setting up the measurement system and helping during the post-processing with Dynamic Studio.

**Author contributions** M.R. contributed to methodology, software, validation, experiments, formal analysis, investigation, data curation, writing—original draft and visualization. M.M. was involved in conceptualization, methodology, software, investigation, resources, writing—review and editing, supervision, project administration and funding acquisition.

**Data availability** Datasets generated during the current study are available from the corresponding author on reasonable request.

## Declaration

**Conflict of interest** The authors declare no conflict of interest.

## References

- Agarwal K, Ram O, Wang J, Lu Y, Katz J (2021) Reconstructing velocity and pressure from noisy sparse particle tracks using constrained cost minimization. *Exp Fluids* 62(4):75. <https://doi.org/10.1007/s00348-021-03172-0>
- Agüera N, Cafiero G, Astarita T, Discetti S (2016) Ensemble 3D PTV for high resolution turbulent statistics. *Measur Sci Technol*. <https://doi.org/10.1088/0957-0233/27/12/124011>
- Agüí JC, Jiménez J (1987) On the performance of particle tracking. *J Fluid Mech* 185:447–468. <https://doi.org/10.1017/S0022112087003252>
- Atkinson C, Soria J (2009) An efficient simultaneous reconstruction technique for tomographic particle image velocimetry. *Exp Fluids* 47:553–568. <https://doi.org/10.1007/s00348-009-0728-0>
- Atkinson C, Buchmann N, Amili O, Soria J (2014) On the appropriate filtering of PIV measurements of turbulent shear flows. *Exp Fluids* 55:1654. <https://doi.org/10.1007/s00348-013-1654-8>
- Bishop CM (2011) *Pattern Recognition and Machine Learning*. Springer, Berlin
- Boyd S, Vandenberghe L (2004) *Convex Optimization*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511804441>
- Brücker C, Hess D, Watz B (2020) Volumetric calibration refinement of a multi-camera system based on tomographic reconstruction of particle images. *Optics* 1:114–135. <https://doi.org/10.3390/opt1010009>
- Cavoretto R (2021) Adaptive radial basis function partition of unity interpolation: a bivariate algorithm for unstructured data. *J Sci Comput*. <https://doi.org/10.1007/s10915-021-01432-z>
- Cavoretto R, De Rossi A (2019) Adaptive meshless refinement schemes for RBF-PUM collocation. *Appl Math Letters* 90:131–138. <https://doi.org/10.1016/j.aml.2018.10.026>
- Cavoretto R, De Rossi A (2020) Error indicators and refinement strategies for solving Poisson problems through a RBF partition of unity collocation scheme. *Appl Math Comput* 369:124824. <https://doi.org/10.1016/j.amc.2019.124824>
- Chen W, Fu ZJ, Chen CS (2014) *Recent Advances in Radial Basis Function Collocation Methods*, 1st edn. Springer, Berlin. <https://doi.org/10.1007/978-3-642-39572-7>
- Cierpka C, Lütke B, Kähler CJ (2013) Higher order multi-frame particle tracking velocimetry. *Exp Fluids*. <https://doi.org/10.1007/s00348-013-1533-3>
- Deisenroth MP, Faisal AA, Ong CS (2020) *Mathematics for Machine Learning*. Cambridge University Press, Cambridge
- Discetti S, Astarita T (2014) The detrimental effect of increasing the number of cameras on self-calibration for tomographic PIV. *Measur Sci Technol* 25:084001. <https://doi.org/10.1088/0957-0233/25/8/084001>
- Discetti S, Coletti F (2018) Volumetric velocimetry for fluid flows. *Measur Sci Technol*. <https://doi.org/10.1088/1361-6501/aaa571>
- Elsinga G, Scarano F, Wieneke B, Oudheusden B (2006) Tomographic particle image velocimetry. *Exp Fluids* 41:933–947. <https://doi.org/10.1007/s00348-006-0212-z>
- Fasshauer GE (2007) *Meshfree approximation methods with MATLAB*, vol 6. World Scientific, Singapore
- Fornberg B, Flyer N (2015) Solving PDEs with radial basis functions. *Acta Numer* 24:215–258
- Gesemann S, Huhn F, Schanz D, Schröder A (2016) From Noisy Particle Tracks to Velocity, Acceleration and Pressure Fields

- using B-splines and Penalties. In: 18th International Symposium on Applications of Laser Techniques to Fluid Mechanics
- Godbersen P, Schröder A (2020) Functional binning: improving convergence of eulerian statistics from Lagrangian particle tracking. *Measur Sci Technol*. <https://doi.org/10.1088/1361-6501/ab8b84>
- Giúemes Jiménez A, Sanmiguel Vila C, Discetti S (2022) Super-resolution generative adversarial networks of randomly seeded fields. *Nat Mach Intell* 4:1–9. <https://doi.org/10.1038/s42256-022-00572-7>
- Hastie T, Tibshirani R, Friedman J (2009) *The Elements of Statistical Learning*. Springer, New York. <https://doi.org/10.1007/978-0-387-84858-7>
- Jahn T, Schanz D, Schröder A (2021) Advanced iterative particle reconstruction for Lagrangian particle tracking. *Exp Fluids*. <https://doi.org/10.1007/s00348-021-03276-7>
- Jeon Y, Müller M, Michaelis D (2022) Fine scale reconstruction (VIC#) by implementing additional constraints and coarse-grid approximation into VIC+. *Exp Fluids*. <https://doi.org/10.1007/s00348-022-03422-9>
- Kähler CJ, Astarita T, Vlachos PP, Sakakibara J, Hain R, Discetti S, Foy RRL, Cierpka C (2016) Main results of the 4th International PIV Challenge. *Exp Fluids* 57(79):1–71. <https://doi.org/10.1007/s00348-016-2173-1>
- Kitzhofer J, Kirmse C, Brücker C (2009) High Density, Long-Term 3D PTV Using 3D Scanning Illumination and Telecentric Imaging. In: Wolfgang Nitsche CD (ed) *Imaging Measurement Methods for Flow Analysis*, vol 106. Springer, Berlin, pp 125–134. [https://doi.org/10.1007/978-3-642-01106-1\\_13](https://doi.org/10.1007/978-3-642-01106-1_13)
- Kähler CJ, Scharnowski S, Cierpka C (2012) On the resolution limit of digital particle image velocimetry. *Exp Fluids* 52(6):1629–1639. <https://doi.org/10.1007/s00348-012-1280-x>
- Kähler CJ, Cierpka C, Scharnowski S (2012) On the uncertainty of digital PIV and PTV near walls. *Exp Fluids* 52:1641–1656. <https://doi.org/10.1007/s00348-012-1307-3>
- Larsson E, Lehto E, Heryudono A, Fornberg B (2013) Stable computation of differentiation matrices and scattered node stencils based on gaussian radial basis functions. *SIAM J Sci Comput* 35:2096–2119. <https://doi.org/10.1137/120899108>
- Larsson E, Shcherbakov V, Heryudono A (2017) A least squares radial basis function partition of unity method for solving PDEs. *SIAM J Sci Comput*. <https://doi.org/10.1137/17M1118087>
- Li L, Pan Z (2024) Three-dimensional time-resolved Lagrangian flow field reconstruction based on constrained least squares and stable radial basis function. *Exp Fluids*. <https://doi.org/10.1007/s00348-024-03788-y>
- Li L, Sellappan P, Schmid P, Hickey JP, Cattafesta L, Pan Z (2021) Lagrangian Strain- and Rotation-Rate Tensor Evaluation Based on Multi-pulse Particle Tracking Velocimetry (MPTV) and Radial Basis Functions (RBFs). In: 14th International Symposium on Particle Image Velocimetry
- Maas HG, Gruen A, Papantoniou D (1993) Particle tracking velocimetry in three-dimensional flows—Part I: photogrammetric determination of particle coordinates. *Exp Fluids* 15:133–146. <https://doi.org/10.1007/BF00190953>
- Malik N, Dracos T, Papantoniou D (1993) Particle tracking velocimetry in three-dimensional flows—Part II: particle tracking. *Exp Fluids* 15:279–294. <https://doi.org/10.1007/BF00223406>
- Marchi SD, Perracchione E (2018) *Lectures on radial basis functions*. University of Padua (Italy), Tech. rep
- Melenk J, Babuška I (1996) The partition of unity finite element method: basic theory and applications. *Comput Methods Appl Mech Eng* 139(1–4):289–314. [https://doi.org/10.1016/s0045-7825\(96\)01087-0](https://doi.org/10.1016/s0045-7825(96)01087-0)
- Neeteson N, Rival D (2015) Pressure-field extraction on unstructured flow data using a voronoi tessellation-based networking algorithm: a proof-of-principle study. *Exp Fluids*. <https://doi.org/10.1007/s00348-015-1911-0>
- Neeteson N, Bhattacharya S, Rival D, Michaelis D, Schanz D, Schröder A (2016) Pressure-field extraction from Lagrangian flow measurements: first experiences with 4D-PTV data. *Exp Fluids*. <https://doi.org/10.1007/s00348-016-2170-4>
- Nocedal J, Wright SJ (2006) *Numerical Optimization*, 2nd edn. Springer-Verlag, New York. <https://doi.org/10.1007/978-0-387-40065-5>
- Novara M, Schanz D, Reuther N, Kähler C, Schröder A (2016) Lagrangian 3D particle tracking in high-speed flows: Shake-The-Box for multi-pulse systems. *Exp Fluids*. <https://doi.org/10.1007/s00348-016-2216-7>
- Park JH, Choi W, Yoon GY, Lee SJ (2020) Deep learning-based super-resolution ultrasound speckle tracking velocimetry. *Ultrasound Med Biol* 46(3):598–609. <https://doi.org/10.1016/j.ultrasmedbio.2019.12.002>
- Pröbsting S, Scarano F, Bernardini M, Pirozzoli S (2013) On the estimation of wall pressure coherence using time-resolved tomographic PIV. *Exp Fluids* 54:1–15. <https://doi.org/10.1007/s00348-013-1567-6>
- Raffel M, Willert CE, Scarano F, Kähler C, Wereley ST, Kompenhans J (2018) *Particle Image Velocimetry—A Practical Guide*, 3rd edn. Springer International Publishing, Berlin. <https://doi.org/10.1007/978-3-319-68852-7>
- Rao C, Sun H, Liu Y (2020) Physics-informed deep learning for incompressible laminar flows. *Theor Appl Mech Letters* 10(3):207–212. <https://doi.org/10.1016/j.taml.2020.01.039>
- Ratz M, Fiorini D, Simonini A, Cierpka C, Mendez M (2022a) Analysis of an unsteady quasi-capillary channel flow with time-resolved PIV and RBF-based super-resolution. *J Coat Technol Res*. <https://doi.org/10.1007/s11998-022-00664-4>
- Ratz M, König J, Mendez M, Cierpka C (2022b) Radial basis function regression of Lagrangian three-dimensional particle tracking data. In: 20th International Symposium on Applications of Laser and Imaging Techniques to Fluid Mechanics
- Scarano F (2013) Tomographic PIV: principles and practice. *Measur Sci Technol* 24:012001. <https://doi.org/10.1088/0957-0233/24/1/012001>
- Scarano F, Schneiders J, González Saiz G, Sciacchitano A (2022) Dense velocity reconstruction with VIC-based time-segment assimilation. *Exp Fluids*. <https://doi.org/10.1007/s00348-022-03437-2>
- Schanz D, Gesemann S, Schröder A, Wieneke B, Novara M (2013) Non-uniform optical transfer functions in particle imaging: calibration and application to Tomographic reconstruction. *Measur Sci Technol* 24:024009. <https://doi.org/10.1088/0957-0233/24/2/024009>
- Schanz D, Gesemann S, Schröder A (2016) Shake-The-Box: Lagrangian particle tracking at high particle image densities. *Exp Fluids*. <https://doi.org/10.1007/s00348-016-2157-1>
- Schneiders J, Scarano F (2016) Dense velocity reconstruction from tomographic PTV with material derivatives. *Exp Fluids*. <https://doi.org/10.1007/s00348-016-2225-6>
- Schröder A, Schanz D (2023) 3D Lagrangian particle tracking in fluid mechanics. *Annu Rev Fluid Mech* 55(1):511–540. <https://doi.org/10.1146/annurev-fluid-031822-041721>
- Schröder A, Schanz D, Novara M, Philipp F, Geisler R, Agocs J, Knopp T, Schroll M (2018) Investigation of a high Reynolds number turbulent boundary layer flow with adverse pressure gradients using PIV and 2D- and 3D-Shake-The-Box. In: 19th International Symposium on the Application of Laser and Imaging Techniques to Fluid Mechanics
- Shepard D (1968) A Two-Dimensional Interpolation Function for Irregularly-Spaced Data. In: *Proceedings of the 1968 23rd ACM National Conference*, Association for Computing Machinery, New York, pp 517–524

- Sperotto P, Pieraccini S, Mendez MA (2022) A meshless method to compute pressure fields from image velocimetry. *Measur Sci Technol*. <https://doi.org/10.1088/1361-6501/ac70a9>
- Sperotto P, Ratz M, Mendez MA (2024a) SPICY: a Python toolbox for meshless assimilation from image velocimetry using radial basis functions. *J Open Source Softw*. <https://doi.org/10.21105/joss.05749>
- Sperotto P, Watz B, Hess D (2024b) Meshless track assimilation (MTA) of 3D PTV data. *Measur Sci Technol* 35:086005. <https://doi.org/10.1088/1361-6501/ad3f36>
- Tan S, Salibindla A, Masuk AUM, Ni R (2020) Introducing OpenLPT: new method of removing ghost particles and high-concentration particle shadow tracking. *Exp Fluids*. <https://doi.org/10.1007/s00348-019-2875-2>
- Wendland H (1995) Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Adv Comput Math* 4:389–396
- Wendland H (2002) Fast evaluation of radial basis functions: methods based on partition of unity. In: Chui CK (ed) *Approximation Theory X: Wavelets, Spline, and Applications*, Vanderbilt Univ. Press, Nashville, pp 473–483
- Westerweel J, Scarano F (2005) Universal outlier detection for PIV data. *Exp Fluids* 39:1096–1100. <https://doi.org/10.1007/s00348-005-0016-6>
- Wieneke B (2013) Iterative reconstruction of volumetric particle distribution. *Meas Sci Technol* 24(2):024008. <https://doi.org/10.1088/0957-0233/24/2/024008>
- Zhang X, Song K, Lu M (2000) Meshless methods based on collocation with radial basis functions. *Comput Mech* 26:333–343. <https://doi.org/10.1007/s004660000181>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.