TRAIN ONCE FOR BUDGET-ADAPTIVE DEPLOYMENT: LEARNING NESTED SENSOR LAYOUTS FOR PHYSICAL FIELD RECONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Accurately reconstructing physical fields from sparse observations depends on model capability and sensor placement. In practice, sensor placement must also accommodate budgets that may change with resource availability or sensing requirements. However, existing placement methods typically optimize layouts for a single sensor budget, so budget changes require reoptimizing all sensor locations. We therefore propose Budget-Adaptive Sensor Placement (BASP), a differentiable framework that learns nested layouts across budgets in a single training run. BASP first encodes the physical field prior into spatial features, supplementing reconstruction feedback with explicit structural guidance. Then, ordered sensor queries sequentially generate continuous coordinates by combining cross attention to these features with spatial suppression, forming nested sensor layouts. During training, a pretrained, frozen reconstructor jointly supervises reconstruction at a randomly sampled budget and the maximum budget, encouraging early sensors to remain informative across layout sizes. Experiments on four datasets with four reconstructors show BASP achieves lower reconstruction errors than independent coordinate optimization, random and uniform sampling, D-, A-, and E-optimal placement in most settings. Beyond accuracy gains, BASP supports multiple budgets in one training run and enables rapid budget adjustment without relocating retained sensors.

1 INTRODUCTION

Reconstructing physical fields from sparse sensor measurements is a fundamental task across applications in fluid dynamics (Vinuesa et al., 2023; Zhang et al., 2025), structural health monitoring (Plevris & Papazafeiropoulos, 2024; Malekloo et al., 2022), medicine (Wang et al., 2020; Koetzier et al., 2023), and atmospheric physics (Price et al., 2025; Eusebi et al., 2024). In practical deployments, the number of sensors is usually limited by sensor costs, installation requirements, and environmental conditions (Clark et al., 2018). Under limited sensing budgets, reconstruction accuracy depends strongly on where measurements are collected. Accurate physical field reconstruction thus requires both capable reconstruction models and sensor placement strategies that maximize the information obtained from limited measurements (Erichson et al., 2020; Ma et al., 2025).

Traditional placement methods commonly apply D-, A-, and E-optimality criteria to reduced representations of the physical field to identify informative sensor locations (Nakai et al., 2021; Nagata et al., 2022). As illustrated in Figure 1a, these methods select sensor locations based on proper orthogonal decomposition (POD) modes, without feedback from the downstream reconstructor. Their objectives characterize measurement informativeness within a prescribed basis, which may not align with the information needed by a nonlinear neural reconstruction model (Williams et al., 2022). Consequently, layouts favored by these criteria do not necessarily make effective use of the reconstructor's learned capabilities.

Recent methods address this limitation by incorporating reconstruction feedback into sensor placement optimization. Such optimization has been integrated into specific reconstruction models, including pressure reconstruction networks (Zhu et al., 2023; Zou et al., 2025) and Senseiver (Marcato et al., 2024). More general frameworks include PhySense, which optimizes sensor coordinates

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

through a pretrained, frozen generative reconstructor (Ma et al., 2025), and DSPO, which alternates between model training and coordinate optimization (Liu et al., 2025). In this work, we refer to the placement optimization stage of PhySense as FixedCoords, as illustrated in Figure 1b. Although these methods align sensor placement with the reconstruction objective, they typically optimize layouts for a prescribed sensor budget, so adapting to a different budget requires reoptimizing the entire layout and may require relocating deployed sensors. In practice, budgets may change with resource availability or sensing requirements (Santos et al., 2023), creating a need for layouts that accommodate such changes without repeated optimization or relocation of retained sensors.

To address these limitations, we propose Budget-Adaptive Sensor Placement (BASP), a differentiable framework that learns nested sensor layouts across supported budgets in a single placement training run. Its placement network, PlaceNet, first encodes a physical field prior constructed from the training set into shared spatial features, providing explicit structural guidance alongside reconstruction feedback. Ordered sensor queries attend to these features through cross attention, while spatial suppression incorporates preceding selections to discourage redundant placement. The queries sequentially generate continuous sensor coordinates, and each supported budget uses the corresponding prefix of the resulting sequence, making the layouts nested by construction. During training, reconstruction objectives at a randomly sampled budget and the maximum budget jointly optimize PlaceNet through a pretrained reconstructor whose parameters remain frozen. This joint supervision encourages early sensors to remain informative both in smaller layouts and within the complete layout. BASP offers three main advantages:

- **A single training process for multiple budgets.** For each dataset and pretrained reconstructor, BASP learns layouts for all supported budgets in one training run, avoiding separate placement optimization for individual budgets.
- **Budget adjustment without relocating retained sensors.** Each budget selects the required prefix of the same coordinate sequence ordered by deployment priority. Increasing or decreasing the budget therefore only adds or removes sensors, preserving the positions of those retained and allowing adjustments without reconfiguring the entire layout.
- **Improved reconstruction across different model architectures.** Across four datasets and four reconstructors, BASP achieves lower reconstruction errors than FixedCoords, random and uniform sampling, and POD-based D-, A-, and E-optimal placement in most evaluated settings.

2 Related Work

Physical field reconstruction. Neural reconstruction methods can be grouped into deterministic and generative models. Deterministic models learn a direct mapping from sparse observations to a field estimate. Early methods relied on prescribed sensor configurations (Fukami et al., 2019; Erichson et al., 2020), while subsequent approaches accommodate varying sensor locations and numbers through structured observation embeddings or attention (Santos et al., 2023). VoronoiCNN introduced Voronoi tessellation as an input representation for neural field reconstruction, converting scattered measurements into structured grids so that convolutional networks can accommodate varying sensor locations and numbers (Fukami et al., 2021). Building on this representation, RecFNO combines Voronoi tessellation inputs with a Fourier neural operator to reconstruct dense physical fields across spatial resolutions (Zhao et al., 2024b). Generative models learn field distributions and incorporate sparse observations to guide reconstruction. These include diffusion methods, such as S^3GM and DiffusionPDE (Li et al., 2024; Huang et al., 2024), and flow matching methods, such as PhySense and POD-FM (Ma et al., 2025; Wang et al., 2026). Across these reconstruction paradigms, flexible sensor inputs enable placement optimization, but do not themselves ensure informative sensor layouts.

Optimal criterion methods for placement. Traditional sensor placement draws on optimal experimental design, which evaluates sensor layouts according to information acquisition and estimation uncertainty (Pukelsheim, 2006). A optimality minimizes average estimation variance, D optimality maximizes overall information content, and E optimality targets the worst case estimation uncertainty (Nakai et al., 2021). These criteria are commonly applied to reduced representations such

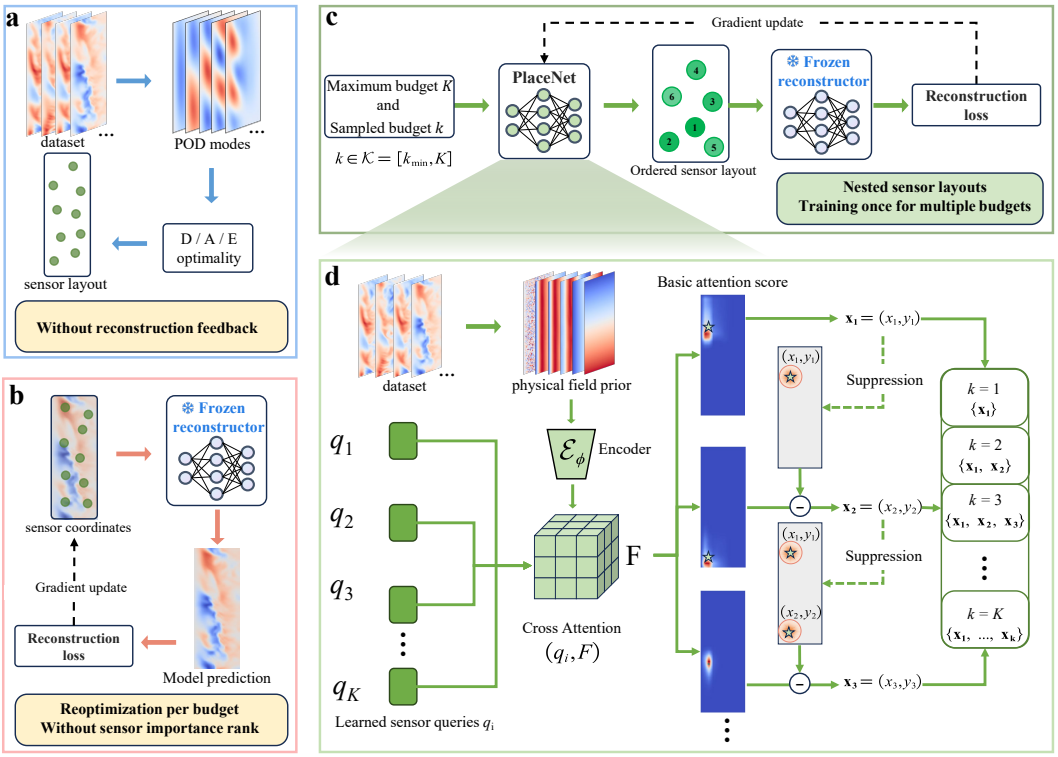

Figure 1: Comparison of sensor placement strategies and overview of BASP. (a) D/A/E optimality criteria select sensors placement from POD modes without reconstruction feedback. (b) FixedCoords directly optimizes sensor coordinates through a pretrained and frozen reconstructor, requiring separate placement optimization for each budget. (c) BASP learns nested layouts across all supported budgets through a single PlaceNet training process per frozen reconstructor, using joint reconstruction supervision at a sampled budget and the maximum budget. (d) PlaceNet encodes the field prior into shared spatial features $\mathbf{F}$. Ordered sensor queries sequentially interact with these features through cross attention to generate continuous coordinates, while spatial suppression based on preceding selections discourages redundant placement.

as POD bases, selecting informative locations without direct feedback from the downstream neural reconstructor (Manohar et al., 2018; Krause et al., 2008).

Neural reconstruction feedback methods for placement. Methods using neural reconstruction feedback can be divided into approaches tailored to specific reconstruction models and general frameworks applicable across reconstruction architectures. The former couple learned coordinate generation, candidate selection, or coordinate optimization with specific reconstruction architectures (Zhu et al., 2023; Zou et al., 2025; Marcato et al., 2024). General strategies include DSPO and the FixedCoords baseline used here. DSPO introduces a bilevel differentiable framework that alternates between reconstruction model training and coordinate optimization for a fixed sensor budget. Each budget requires a separate run of both stages, incurring repeated training costs and producing a reconstructor tailored to the resulting sensor layout (Liu et al., 2025). FixedCoords directly optimizes sensor coordinates using feedback from a pretrained and frozen reconstructor (Ma et al., 2025). It avoids retraining the reconstructor but still requires independent coordinate optimization for each budget.

3 METHOD

Problem setup. Given a pretrained and frozen reconstructor $\mathcal{R}_\theta$, we aim to select sensor locations that enable accurate reconstruction of a physical field u from sparse measurements. We seek nested

layouts across different sensor budgets, allowing sensors to be added or removed without relocating those retained.

Overall design. BASP addresses this problem by jointly learning sensor layouts across a set of supported budgets $\mathcal{K}$, with $K = \max \mathcal{K}$, using reconstruction feedback. Its placement network, PlaceNet, maps a fixed physical field prior $\mathbf{p}$ to a sequence of K continuous sensor coordinates ordered by deployment priority. For a requested budget $k \in \mathcal{K}$, the layout consists of the first k coordinates:

$$\mathbf{X}_\phi^{(k)} = \mathcal{P}_\phi(\mathbf{p}; k) = [\mathbf{x}_1, \ldots, \mathbf{x}_k]^{\mathrm{T}} = \left(\mathbf{X}_\phi^{(K)}\right)_{1:k}, \tag{1}$$

where $\mathbf{x}_i$ denotes the location of the ith sensor in the learned deployment order, and ϕ contains all PlaceNet parameters. Each coordinate is generated from the shared prior and preceding selections independently of the final budget, ensuring nested layouts by construction. As illustrated in Figure 1c,d, PlaceNet combines a shared prior encoder, ordered sensor queries with spatial suppression, and reconstruction supervision across budgets.

3.1 Physical Field Prior

To complement reconstruction feedback with explicit spatial guidance, PlaceNet constructs a physical field prior from the training dataset. The prior summarizes characteristic field patterns and their spatial variation, providing statistical cues for sensor queries to identify informative sensing regions. At each grid location, we compute the mean $\boldsymbol{\mu}$, standard deviation $\boldsymbol{\sigma}$, mean absolute field value $\mathbf{a}$, and root mean square spatial gradient magnitude $\mathbf{g}$ across training samples. These statistics characterize the field distribution, variability, and local spatial changes. After normalization, they are concatenated with normalized coordinate maps $\mathbf{C}_y$ and $\mathbf{C}_x$:

$$\mathbf{p} = [\boldsymbol{\mu}, \boldsymbol{\sigma}, \mathbf{a}, \mathbf{g}, \mathbf{C}_y, \mathbf{C}_x] \in \mathbb{R}^{6 \times n_y \times n_x}, \qquad \mathbf{F} = \mathcal{E}_\phi(\mathbf{p}) \in \mathbb{R}^{d \times H_f \times W_f}. \tag{2}$$

Here, $\mathcal{E}_\phi$ denotes the prior encoder, d is the feature dimension, and $H_f \times W_f$ is the encoded spatial resolution. The coordinate maps provide explicit spatial positions for the field statistics. The prior is computed once and remains fixed during placement optimization, while the encoder is trained through reconstruction feedback. The resulting representation $\mathbf{F}$ integrates these statistical and spatial cues and is shared across all sensor queries and budgets.

3.2 Ordered Sensor Queries with Spatial Suppression

We flatten $\mathbf{F}$ into $N = H_f W_f$ feature vectors $\{\mathbf{f}_i\}_{i=1}^N$, associated with normalized spatial coordinates $\boldsymbol{\xi}_i \in [0,1]^2$. PlaceNet maintains an ordered sequence of learnable queries $\mathbf{Q} = (\mathbf{q}_1, \ldots, \mathbf{q}_K)$. At rank r, the query produces attention scores over the shared spatial features:

$$b_{r,i} = \frac{\psi(\mathbf{q}_r)^{\mathrm{T}} \mathbf{f}_i}{\sqrt{d}}, \qquad i = 1, \ldots, N, \tag{3}$$

where ψ is a learnable projection and $\mathbf{b}_r = (b_{r,1}, \ldots, b_{r,N})^{\mathrm{T}}$.

To discourage redundant placement, each query uses a coverage map $\mathbf{C}_{r-1}$ and an occupancy map $\mathbf{O}_{r-1}$, initialized as $\mathbf{C}_0 = \mathbf{O}_0 = \mathbf{0}$. A preliminary coordinate is obtained using coverage suppression alone, while the final coordinate uses both maps:

$$\begin{aligned} \widetilde{\boldsymbol{\beta}}_r &= \mathrm{softmax}\left(\frac{\mathbf{b}_r - \mathbf{C}_{r-1}}{\tau}\right), & \widetilde{\mathbf{x}}_r &= \sum_{i=1}^N \widetilde{\beta}_{r,i} \boldsymbol{\xi}_i, \\ \boldsymbol{\beta}_r &= \mathrm{softmax}\left(\frac{\mathbf{b}_r - \mathbf{C}_{r-1} - \mathbf{O}_{r-1}}{\tau}\right), & \mathbf{x}_r &= \sum_{i=1}^N \beta_{r,i} \boldsymbol{\xi}_i, \end{aligned} \tag{4}$$

where $\tau > 0$ controls the concentration of the attention distributions. The coverage and occupancy maps are then updated around the preliminary and final coordinates, respectively:

$$\begin{aligned} \mathbf{C}_r &= \max\left\{\mathbf{C}_{r-1}, \mathbf{G}_c(\widetilde{\mathbf{x}}_r)\right\}, \\ \mathbf{O}_r &= \max\left\{\mathbf{O}_{r-1}, \mathbf{G}_o(\mathbf{x}_r)\right\}. \end{aligned} \tag{5}$$

Here, $\mathbf{G}_c$ and $\mathbf{G}_o$ are Gaussian suppression maps centered at the preliminary and final coordinates. The coverage map discourages queries from repeatedly targeting the same regions, while the occupancy map additionally suppresses neighborhoods of the final sensor locations, which may differ from the preliminary coordinates. The maximum is evaluated elementwise to retain the strongest suppression at each location, and gradients are stopped through both map updates.

The coordinate $\mathbf{x}_r$ remains continuous and is passed to bilinear sampling without rounding to a grid point. Executing the first k queries yields $\mathbf{X}_\phi^{(k)}$. Because the same recurrence is used for every budget, increasing or decreasing k preserves all coordinates in the shared prefix.

3.3 Joint Optimization across Sensor Budgets

At each training iteration, we sample a field $\mathbf{u} \sim \mathcal{D}_{\text{train}}$ and a budget $k \sim \text{Unif}(\mathcal{K})$. We supervise both the sampled layout $\mathbf{X}_\phi^{(k)}$ and the maximum layout $\mathbf{X}_\phi^{(K)}$ through the same frozen reconstructor. For the sampled sensor budget k, bilinear sampling provides $\mathbf{s}^{(k)} = \mathcal{S}(\mathbf{u}, \mathbf{X}_\phi^{(k)})$. VoronoiCNN and RecFNO directly predict $\widehat{\mathbf{u}}^{(k)} = \mathcal{R}_\theta(\mathbf{s}^{(k)}, \mathbf{X}_\phi^{(k)})$. The placement loss is defined according to the coupled reconstructor:

$$\ell^{(k)} = \begin{cases} \text{MSE}\big(\widehat{\mathbf{u}}^{(k)}, \mathbf{u}\big), & \text{deterministic models,} \\ \text{MSE}\left(\mathbf{v}_\theta(\mathbf{u}_t, t, \mathbf{c}^{(k)}), \mathbf{u} - \mathbf{u}_0\right), & \text{flow matching models.} \end{cases} \tag{6}$$

In our experiments, the deterministic models are VoronoiCNN and RecFNO, while the flow matching models are PhySense and POD-FM. MSE is averaged over the field grid. The conditioning information $\mathbf{c}^{(k)}$ contains the sensor measurements and coordinates, with POD-FM additionally using estimated POD coefficients. The flow matching expectation is estimated using sampled times and Gaussian noise fields during training. The maximum budget loss $\ell^{(K)}$ is computed in the same manner using $\mathbf{X}_\phi^{(K)}$. PlaceNet minimizes the joint objective

$$\mathcal{L}(\phi) = \mathbb{E}_{\substack{\mathbf{u}\sim\mathcal{D}_{\text{train}} \\ k\sim\text{Unif}(\mathcal{K})}} \left[\frac{1}{2}\left(\ell^{(k)} + \ell^{(K)}\right)\right]. \tag{7}$$

The maximum budget term supervises all K queries at every iteration, while the sampled budget term trains the corresponding prefix. Coordinates within the sampled prefix therefore contribute to both objectives, encouraging early sensors to remain informative as the layout expands. The deployment order is learned through this shared reconstruction supervision without explicit importance labels.

Only ϕ is updated during placement optimization. Gradients propagate through the frozen reconstructor, differentiable input encoding, bilinear sampling, and attention readout, while the suppression map updates remain detached. After training, the complete coordinate sequence can be computed once and reused across supported budgets by selecting the corresponding prefix. Budget changes therefore require neither repeated placement optimization nor relocation of retained sensors.

4 Experiments

Benchmark. We evaluate BASP on four physical field datasets and four reconstructors. Cylinder wake contains vorticity fields with periodic vortex shedding behind a circular cylinder (Zhao et al., 2024b). Turbulent flow contains streamwise velocity fields with irregular and aperiodic turbulent dynamics (Fukami et al., 2021). Aerofoil contains streamwise velocity fields around a NACA0012 aerofoil under varying inlet velocities and angles of attack (Zhao et al., 2024a). Heat conduction consists of steady temperature fields governed by nonlinear heat conduction with localized heat sources (Zhao et al., 2024b). We use four reconstructors spanning convolutional networks (VoronoiCNN (Fukami et al., 2021)), neural operators (RecFNO (Zhao et al., 2024b)), and generative models (PhySense (Ma et al., 2025) and POD-FM (Wang et al., 2026)). Each reconstructor is pretrained with varying sensor counts and kept frozen during sensor placement optimization. Detailed dataset descriptions and reconstruction model settings are provided in Appendix A and B.

Table 1: **Reconstruction errors on cylinder wake and turbulent flow.** Relative L_2 errors ($\times 10^{-2}$) on cylinder wake and turbulent flow. # denotes the number of sensors. D/A/E (best) gives the minimum error across the three optimality criteria for each setting. Within each reconstructor and column, the lowest and second lowest displayed errors are bold and underlined

Reconstructor	Method	Cylinder wake				Turbulent flow			
		#10	#15	#20	#30	#25	#50	#100	#200
VoronoiCNN	D/A/E (best)	20.01	20.78	15.15	10.14	33.17	<u>18.71</u>	13.76	<u>11.55</u>
	RS	9.440	5.556	4.259	3.289	48.48	25.19	16.18	12.68
	US	5.849	3.697	4.700	3.512	33.92	18.90	<u>13.39</u>	11.56
	FixedCoords	<u>2.275</u>	<u>1.781</u>	<u>1.914</u>	<u>1.656</u>	<u>30.81</u>	18.73	14.16	11.85
	BASP	**1.830**	**1.679**	**1.593**	**1.507**	**27.65**	**17.41**	**13.36**	**11.25**
RecFNO	D/A/E (best)	7.206	13.41	13.35	13.92	42.36	25.20	17.74	14.42
	RS	7.602	4.209	3.351	2.838	60.17	37.03	23.11	16.63
	US	3.059	2.276	2.429	2.595	41.36	25.86	16.91	14.09
	FixedCoords	**1.504**	**1.388**	**1.302**	**1.287**	<u>37.54</u>	**23.10**	**16.37**	<u>13.74</u>
	BASP	<u>1.534</u>	<u>1.484</u>	<u>1.429</u>	<u>1.297</u>	**37.41**	<u>23.67</u>	<u>16.55</u>	**13.73**
PhySense	D/A/E (best)	37.39	34.09	34.93	30.04	<u>29.20</u>	<u>15.99</u>	12.71	11.05
	RS	22.53	13.64	9.694	5.922	65.34	28.71	15.28	12.05
	US	61.54	5.905	4.050	5.274	46.61	18.09	12.97	11.09
	FixedCoords	<u>4.081</u>	<u>3.678</u>	<u>3.812</u>	<u>3.530</u>	37.80	16.22	<u>12.60</u>	**10.94**
	BASP	**3.826**	**3.554**	**3.534**	**3.175**	**26.12**	**15.73**	**12.57**	<u>10.96</u>
POD-FM	D/A/E (best)	28.19	31.07	25.07	19.02	<u>44.51</u>	**20.54**	**12.30**	**8.993**
	RS	14.99	9.105	6.501	5.053	76.18	40.15	18.84	11.47
	US	25.44	5.886	3.497	4.576	63.82	25.72	<u>13.57</u>	9.648
	FixedCoords	<u>2.955</u>	<u>2.857</u>	<u>2.968</u>	<u>2.752</u>	51.72	24.85	15.47	10.80
	BASP	**2.785**	**2.673**	**2.575**	**2.337**	**41.20**	<u>22.17</u>	13.71	<u>9.556</u>

Baselines. The D, A, and E optimality criteria select sensor locations based on POD modes without reconstruction feedback (Nakai et al., 2021; Nagata et al., 2022). Random sampling (RS) and uniform sampling (US) place sensors randomly and uniformly across the spatial domain. FixedCoords is a general differentiable placement method that directly optimizes sensor coordinates through a pretrained and frozen reconstructor, making it applicable across reconstruction architectures without retraining the reconstructor (Ma et al., 2025). However, it requires separate coordinate optimization for each sensor budget. BASP requires only one placement training process for each dataset and reconstructor combination to generate nested layouts across all supported budgets.

Implementations. Each reconstructor is pretrained with varying sensor counts and then frozen for all placement methods and evaluated budgets. Field priors and POD bases are computed from training data only. BASP and FixedCoords are optimized using Adam optimizer for 200 epochs per run, with detailed training settings given in the Appendix D. We report the test relative L_2 error, $\varepsilon = \|\hat{u} - u\|_2 / \|u\|_2$, at four sensor budgets for each dataset. For compactness, D/A/E (best) reports the minimum test error among the three criteria separately for each setting. Results for all three criteria are provided in the Appendix E.

4.1 Reconstruction performance and sensor layouts

Cylinder wake. As shown in Table 1, BASP outperforms RS, US, and all three optimality criteria in all 16 settings. It also outperforms FixedCoords for VoronoiCNN, PhySense, and POD-FM, including a 19.6% error reduction for VoronoiCNN with 10 sensors, while FixedCoords performs slightly better for RecFNO. Figure 2a shows BASP sensors distributed along the downstream wake, whereas FixedCoords also places sensors on the domain boundaries and at corners. E optimality instead concentrates sensors immediately behind the cylinder, leaving the farther wake sparsely observed (Appendix E.5). This illustrates a possible mismatch between selection from a prescribed POD basis and the information needs of the neural reconstructor. Increasing the budget from 10 to 15 preserves all ten existing locations and adds five sensors that broaden coverage around the initial

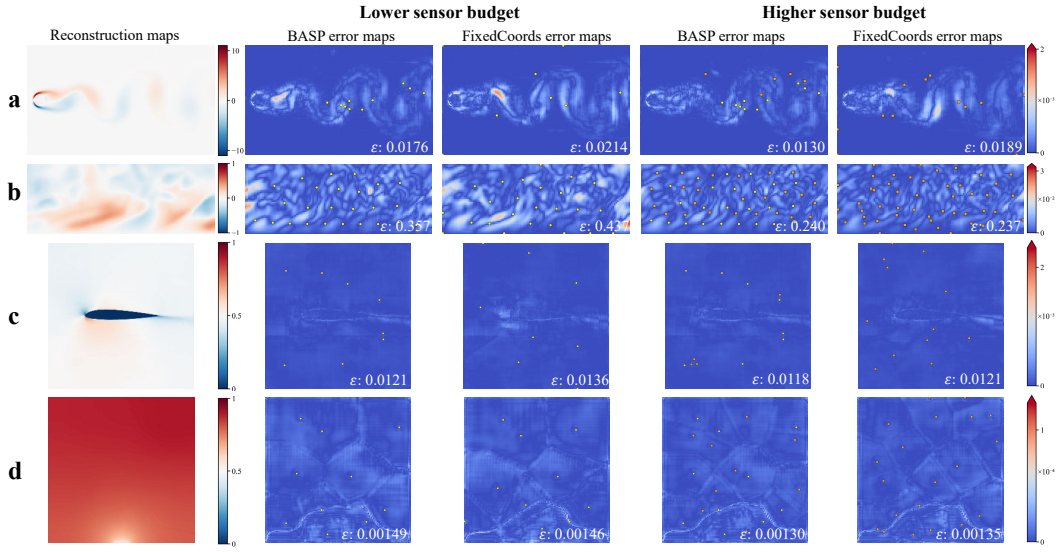

Figure 2: Sensor layouts and reconstruction results for four datasets: (a) cylinder wake, (b) turbulent flow, (c) aerofoil flow, and (d) heat conduction. BASP and FixedCoords share the same pretrained and frozen VoronoiCNN for reconstruction. Yellow circles mark lower-budget sensors; orange squares mark added BASP sensors or the full higher-budget FixedCoords layout. BASP trains once for all budgets and retains existing sensors, whereas FixedCoords optimizes each budget separately. The first column shows BASP reconstructions at the higher budget. ε denotes reconstruction error. Further details and figures are provided in the Appendix E.

set. The same trained placement network supports this budget increase without relocating deployed sensors or repeating placement optimization.

Turbulent flow. The Turbulent flow benchmark tests placement for spatially distributed, aperiodic turbulent structures. On turbulent flow, BASP achieves the lowest error for all four reconstructors at the smallest budget of 25 sensors (Table 1). At this budget, it reduces errors relative to FixedCoords by 30.9% for PhySense and 20.3% for POD-FM. BASP outperforms FixedCoords in 13 of the 16 settings. For POD-FM, however, D/A/E (best) achieves lower errors with 50, 100, and 200 sensors. A possible explanation is their alignment with POD-FM's conditioning mechanism: the reconstructor uses POD coefficients estimated from sparse observations as global guidance, and placement based on the same basis may support their estimation. This result highlights the dependence of placement effectiveness on both field structure and the information used by the reconstructor. Figure 2b shows that both methods provide approximately uniform coverage with local variations in spacing, consistent with the spatially distributed turbulent structures. The 50 sensor BASP layout retains all 25 existing locations and adds sensors in previously uncovered regions, increasing sampling density without relocating retained sensors.

Aerofoil flow. Table 2 shows that BASP outperforms RS, US, and all three optimality criteria in all 16 settings, and FixedCoords in 12. The error reductions include 46.0% relative to US for RecFNO with 12 sensors and 20.3% relative to FixedCoords for VoronoiCNN with eight sensors. The layouts and error maps in Figure 2c reveal distinct spatial distributions among the placement strategies. D/A/E (best) optimality concentrates sensors near the leading edge and in a narrow region extending into the downstream wake, with pronounced reconstruction errors remaining around the aerofoil (Appendix E.5). BASP and FixedCoords distribute sensors more broadly through the surrounding flow, including regions above and below the aerofoil, and achieve smaller errors in the illustrated cases. This comparison suggests that observations distributed across the surrounding flow provide useful information for reconstruction under varying aerodynamic conditions, while concentrating measurements near the aerofoil and wake does not necessarily meet the information needs of the neural reconstructors. For both BASP and FixedCoords, reconstruction errors change only modestly as the budget increases from 8 to 20 sensors across all four models. This suggests that

Table 2: **Reconstruction errors on aerofoil and heat conduction.** Relative L_2 errors ($\times 10^{-2}$) on aerofoil flow and heat conduction. # denotes the number of sensors. D/A/E (best) gives the minimum error across the three optimality criteria for each setting. Within each reconstructor and column, the lowest and second lowest displayed errors are bold and underlined, respectively.

Reconstructor	Method	Aerofoil flow				Heat conduction			
		#8	#12	#16	#20	#10	#20	#30	#40
VoronoiCNN	D/A/E (best)	14.36	14.12	13.00	9.096	0.3151	0.2336	0.2102	0.2073
	RS	6.846	3.947	2.666	2.151	0.4921	0.2936	0.2409	0.2226
	US	2.460	2.521	1.625	1.577	0.4005	0.2440	0.1978	0.1909
	FixedCoords	1.912	1.626	**1.370**	**1.339**	0.2004	**0.1741**	0.1661	0.1665
	BASP	**1.524**	**1.441**	1.429	1.400	**0.2003**	0.1755	**0.1658**	**0.1619**
RecFNO	D/A/E (best)	8.192	7.085	7.271	3.992	0.6790	0.4440	0.4340	0.3745
	RS	0.7206	0.4352	0.3840	0.3624	1.023	0.6470	0.5549	0.5224
	US	0.2973	0.3666	0.2520	0.2585	0.5312	0.4850	0.4722	0.4669
	FixedCoords	0.2234	0.2082	**0.1895**	0.1915	**0.3390**	**0.2497**	**0.2480**	**0.2391**
	BASP	**0.2128**	**0.1979**	0.1926	**0.1869**	0.3423	0.2727	0.2513	0.2430
PhySense	D/A/E (best)	8.606	9.307	9.088	7.723	0.3981	0.2970	0.3156	0.3171
	RS	1.068	0.7637	0.7676	0.7182	0.5741	0.3929	0.3399	0.3121
	US	0.6850	0.7335	0.7359	0.7331	0.3687	0.2881	0.2862	0.2691
	FixedCoords	**0.6174**	0.6197	0.6149	0.6228	0.2640	0.2455	0.2480	0.2379
	BASP	0.6183	**0.6134**	**0.6133**	**0.6083**	**0.2519**	**0.2398**	**0.2383**	**0.2370**
POD-FM	D/A/E (best)	9.870	10.74	9.604	5.965	0.2531	0.2485	0.2479	0.2485
	RS	1.430	0.6810	0.6872	0.5911	0.4331	0.3059	0.2682	0.2659
	US	0.5139	0.5258	0.5296	0.5242	0.3077	0.2573	0.2506	0.2487
	FixedCoords	0.4964	0.4934	0.4893	0.4920	0.2458	0.2434	**0.2428**	**0.2435**
	BASP	**0.4898**	**0.4889**	**0.4883**	**0.4885**	**0.2454**	**0.2425**	0.2430	0.2446

eight optimized measurements already capture much of the information exploited by the frozen reconstructors, leaving limited marginal gains from additional sensors. Consistent with this trend, the 12 sensor BASP layout preserves all eight existing locations and places the four additional sensors close to several retained locations, producing only limited expansion of spatial coverage.

Heat conduction. On heat conduction, BASP and FixedCoords outperform RS, US, and the three optimality criteria throughout the evaluated settings, with comparable reconstruction errors (Table 2). BASP performs better in nine of the 16 settings, while FixedCoords remains better for RecFNO. Figure 2d shows that both methods combine broad coverage with several sensors in the lower region influenced by the heat sink. This distribution is consistent with the varying intensities of localized heat sources and the propagation of their thermal effects throughout the domain, which make both spatial coverage and the allocation of measurements around local thermal variations relevant to reconstruction. When the BASP budget increases from 10 to 20 sensors, all original locations are retained and the additional sensors mainly occupy gaps between them. This complementary expansion is consistent with the learned deployment priority and spatial suppression mechanism, enabling denser coverage without relocating existing sensors. BASP achieves this placement effectiveness through a single training process per reconstructor for all supported budgets, while FixedCoords requires independent coordinate optimization for each budget.

4.2 Ablation studies

We evaluate component ablations and coordinate optimization controls on cylinder wake and aerofoil. Within each dataset, all variants use the same pretrained and frozen VoronoiCNN and data partitions. Table 3 reports results at every evaluated budget.

Component ablations. We separately remove the physical field prior, spatial suppression, or maximum budget supervision while retaining the remaining components and training settings. Removing the prior increases errors in seven of the eight settings. The degradation is more pronounced on aero-

Table 3: **Component ablations and coordinate optimization controls.** Relative L_2 errors ($\times 10^{-2}$) are reported using the same frozen VoronoiCNN within each dataset. # denotes the number of sensors. The lowest and second lowest errors among all evaluated variants in each column are bold and underlined, respectively. Dashes indicate unevaluated settings.

Variant	Cylinder wake				Aerofoil			
	#10	#15	#20	#30	#8	#12	#16	#20
BASP (full)	<u>1.830</u>	**1.679**	**1.593**	**1.507**	<u>1.524</u>	<u>1.441</u>	1.429	1.400
w/o data dependent prior	**1.779**	1.760	1.758	1.695	2.573	1.935	1.921	1.851
w/o spatial suppression	3.094	3.106	3.294	2.217	5.043	3.554	3.567	3.578
w/o maximum budget supervision	3.842	2.938	2.882	2.323	2.228	1.705	1.695	1.567
OrderedCoords	2.152	2.104	2.004	1.687	1.726	1.541	1.480	1.480
FixedCoords	2.275	1.781	1.914	1.656	1.912	1.626	**1.370**	<u>1.339</u>
FixedCoords (BASP init.)	2.197	1.893	1.764	1.588	**1.474**	**1.392**	<u>1.372</u>	**1.328**
FixedCoords (frozen prefix)	–	<u>1.704</u>	<u>1.607</u>	<u>1.528</u>	–	1.450	1.431	1.458

foil, while cylinder wake also shows increased errors at budgets of 15, 20, and 30 sensors. Removing spatial suppression consistently worsens reconstruction, particularly on aerofoil: even with 20 sensors, the error rises from 1.400×10^{-2} to 3.578×10^{-2}. The degradation persists across budgets, supporting the importance of accounting for previous selections to discourage redundant placement. Removing maximum budget supervision also increases errors in all settings. On cylinder wake with 10 sensors, the error rises from 1.830×10^{-2} to 3.842×10^{-2}, suggesting that supervision at the maximum budget also helps maintain informative early selections. Together, these results support the complementary roles of physical field guidance, spatial suppression, and joint supervision across budgets.

Comparison with coordinate optimization. We first test whether nested coordinate ordering and joint supervision are sufficient to explain BASP's performance. OrderedCoords directly learns an ordered coordinate sequence using the same sampled budget and maximum budget objectives as BASP, enabling a single training run to produce nested layouts for multiple budgets without PlaceNet. However, it yields higher errors than independently optimized FixedCoords layouts at larger budgets and remains less accurate than BASP in all evaluated settings. We then examine whether further coordinate refinement can improve the layouts learned by BASP. FixedCoords (BASP init.) starts from each BASP layout and independently optimizes all coordinates for that budget. This initialization improves upon the original FixedCoords baseline in most settings, suggesting that BASP provides effective starting points for coordinate optimization. On aerofoil, the refined layouts achieve lower errors than BASP at every tested budget, demonstrating room for further improvement when all coordinates are allowed to move. To assess refinement without relocating existing sensors, FixedCoords (frozen prefix) fixes the BASP coordinates from the preceding tested budget and optimizes only added sensors. It consistently yields higher errors than BASP.

5 Conclusion

We present BASP, a framework that learns nested sensor layouts across supported budgets through a single placement training run for each dataset and reconstructor pair. A physical field prior, ordered queries with spatial suppression, and joint supervision at sampled and maximum budgets guide placement through a frozen reconstructor. The resulting sequence supports sensor addition and removal without relocating retained sensors or repeating placement optimization. Across four datasets and four reconstructors, BASP achieves lower reconstruction errors than conventional placement methods in nearly all evaluated settings and independent coordinate optimization in most cases, demonstrating the value of jointly learning reusable layouts. The GPU memory required for backpropagation currently limits extensions to diffusion reconstructors and sensor placement in three dimensions, motivating future work on memory efficient gradient computation.

AI USE STATEMENT

In this work, we used generative AI tools to polish writing. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We will make the source code publicly available on GitHub upon acceptance.

REFERENCES

Emily Clark, Travis Askham, Steven L Brunton, and J Nathan Kutz. Greedy sensor placement with cost constraints. *IEEE sensors journal*, 19(7):2642–2656, 2018.

N Benjamin Erichson, Lionel Mathelin, Zhewei Yao, Steven L Brunton, Michael W Mahoney, and J Nathan Kutz. Shallow neural networks for fluid flow reconstruction with limited sensors. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 476(2238): 20200097, 2020.

Ryan Eusebi, Gabriel A Vecchi, Ching-Yao Lai, and Mingjing Tong. Realistic tropical cyclone wind and pressure fields can be reconstructed from sparse data using deep learning. *Communications Earth & Environment*, 5(1):8, 2024.

Kai Fukami, Koji Fukagata, and Kunihiko Taira. Super-resolution reconstruction of turbulent flows with machine learning. *Journal of Fluid Mechanics*, 870:106–120, 2019.

Kai Fukami, Romit Maulik, Nesar Ramachandra, Koji Fukagata, and Kunihiko Taira. Global field reconstruction from sparse sensors with voronoi tessellation-assisted deep learning. *Nature Machine Intelligence*, 3(11):945–951, 2021.

Jiahe Huang, Guandao Yang, Zichen Wang, and Jeong Joon Park. Diffusionpde: Generative pde-solving under partial observation. *Advances in Neural Information Processing Systems*, 37: 130291–130323, 2024.

Lennart R Koetzier, Domenico Mastrodicasa, Timothy P Szczykutowicz, Niels R van der Werf, Adam S Wang, Veit Sandfort, Aart J van der Molen, Dominik Fleischmann, and Martin J Willemink. Deep learning image reconstruction for ct: technical principles and clinical prospects. *Radiology*, 306(3):e221257, 2023.

Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9(8):235–284, 2008. URL http://jmlr.org/papers/v9/krause08a.html.

Zeyu Li, Wang Han, Yue Zhang, Qingfei Fu, Jingxuan Li, Lizi Qin, Ruoyu Dong, Hao Sun, Yue Deng, and Lijun Yang. Learning spatiotemporal dynamics with a pretrained generative model. *Nature Machine Intelligence*, 6(12):1566–1579, 2024.

Zongyi Li, Nikola Borislavov Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=c8P9NQVtmnO.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Xu Liu, Wei Peng, Xiaoya Zhang, Xiaoyu Zhao, Weien Zhou, Wen Yao, and Xiaoqian Chen. Enhancing deep learning-based field reconstruction with a differentiable learning framework. *Nature Machine Intelligence*, 7(7):1129–1140, 2025.

Yuezhou Ma, Haixu Wu, Hang Zhou, Huikun Weng, Jianmin Wang, and Mingsheng Long. Physense: Sensor placement optimization for accurate physics sensing. *Advances in Neural Information Processing Systems*, 38:35697–35725, 2025.

Arman Malekloo, Ekin Ozer, Mohammad AlHamaydeh, and Mark Girolami. Machine learning and structural health monitoring overview with emerging technology and high-dimensional data source highlights. *Structural Health Monitoring*, 21(4):1906–1955, 2022.

Krithika Manohar, Bingni W Brunton, J Nathan Kutz, and Steven L Brunton. Data-driven sparse sensor placement for reconstruction: Demonstrating the benefits of exploiting known patterns. *IEEE Control Systems Magazine*, 38(3):63–86, 2018.

Agnese Marcato, Eric Guiltinan, Hari Viswanathan, Daniel O'Malley, Nicholas Lubbers, and Javier E Santos. Journey over destination: dynamic sensor placement enhances generalization. *Machine Learning: Science and Technology*, 5(2):025070, 2024.

Takayuki Nagata, Keigo Yamada, Taku Nonomura, Kumi Nakai, Yuji Saito, and Shunsuke Ono. Data-driven sensor selection method based on proximal optimization for high-dimensional data with correlated measurement noise. *IEEE Transactions on Signal Processing*, 70:5251–5264, 2022.

Kumi Nakai, Keigo Yamada, Takayuki Nagata, Yuji Saito, and Taku Nonomura. Effect of objective function on data-driven greedy sparse sensor optimization. *IEEE Access*, 9:46731–46743, 2021.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

Vagelis Plevris and George Papazafeiropoulos. Ai in structural health monitoring for infrastructure maintenance and safety. *Infrastructures*, 9(12):225, 2024.

Ilan Price, Alvaro Sanchez-Gonzalez, Ferran Alet, Tom R Andersson, Andrew El-Kadi, Dominic Masters, Timo Ewalds, Jacklynn Stott, Shakir Mohamed, Peter Battaglia, et al. Probabilistic weather forecasting with machine learning. *Nature*, 637(8044):84–90, 2025.

Friedrich Pukelsheim. *Optimal design of experiments*. SIAM, 2006.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi (eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, volume 9351 of *Lecture Notes in Computer Science*, pp. 234–241, Cham, 2015. Springer. doi: 10.1007/978-3-319-24574-4_28.

Javier E Santos, Zachary R Fox, Arvind Mohan, Daniel O'Malley, Hari Viswanathan, and Nicholas Lubbers. Development of the senseiver for efficient field reconstruction from sparse observations. *Nature Machine Intelligence*, 5(11):1317–1325, 2023.

Ricardo Vinuesa, Steven L Brunton, and Beverley J McKeon. The transformative potential of machine learning for experiments in fluid mechanics. *Nature Reviews Physics*, 5(9):536–545, 2023.

Ge Wang, Jong Chul Ye, and Bruno De Man. Deep learning for tomographic image reconstruction. *Nature machine intelligence*, 2(12):737–748, 2020.

Xin Wang, Dali Li, Laiping Zhang, and Xiaogang Deng. Flow matching with gappy proper orthogonal decomposition guidance for balanced global field reconstruction. *Physics of Fluids*, 38(2): 027113, 02 2026.

Karen Willcox. Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition. *Computers & fluids*, 35(2):208–226, 2006.

Jan Williams, Olivia Zahn, and J Nathan Kutz. Data-driven sensor placement with shallow decoder networks. *arXiv preprint arXiv:2202.05330*, 2022.

Renjie Xiao, Bingteng Sun, Yiling Chen, Lin Lu, Qiang Du, and Junqiang Zhu. Flow field reconstruction via voronoi-enhanced physics-informed neural networks with end-to-end sensor placement optimization. *arXiv preprint arXiv:2603.09371*, 2026.

Qian Zhang, Dmitry Krotov, and George Em Karniadakis. Operator learning for reconstructing flow fields from sparse measurements: an energy transformer approach. *Journal of Computational Physics*, 538:114148, 2025.

Xiaoyu Zhao, Xiaoqian Chen, Zhiqiang Gong, Wen Yao, and Yunyang Zhang. A hybrid method based on proper orthogonal decomposition and deep neural networks for flow and heat field reconstruction. *Expert Systems with Applications*, 247:123137, 2024a.

Xiaoyu Zhao, Xiaoqian Chen, Zhiqiang Gong, Weien Zhou, Wen Yao, and Yunyang Zhang. Recfno: A resolution-invariant flow and heat field reconstruction method from sparse observations via fourier neural operator. *International Journal of Thermal Sciences*, 195:108619, 2024b.

Qiming Zhu, Ze Zhao, and Jinhui Yan. Physics-informed machine learning for surrogate modeling of wind pressure and optimization of pressure sensor placement. *Computational Mechanics*, 71 (3):481–491, 2023.

Junhong Zou, Wei Qiu, Zhenxu Sun, Xiaomei Zhang, Zhaoxiang Zhang, Xiangyu Zhu, and Zhen Lei. Joint optimization of sensor placement and sparse pressure field reconstruction with a two-stage framework for limited data. *Physics of Fluids*, 37(7):075231, 2025.

A Dataset Details

Cylinder wake The first benchmark is the cylinder wake dataset, a canonical test case for sparse flow reconstruction (Zhao et al., 2024b; Santos et al., 2023). It describes a two dimensional unsteady flow past a circular cylinder, where alternating vortex shedding forms a periodic wake downstream. The dataset is generated by solving the incompressible Navier–Stokes equations within a rectangular domain measuring 15.7 cylinder diameters in the streamwise direction and 10 cylinder diameters in the transverse direction. It contains 5,000 snapshots of the vorticity field, each at a resolution of 192×112, with 3,500 snapshots used for training and the remaining 1,500 for testing. The vorticity field is used as the target global field to evaluate the effectiveness of different sensor placement methods.

Turbulent flow The turbulent flow dataset is used to evaluate the proposed method under strongly nonlinear and aperiodic turbulent dynamics. It is generated from a three dimensional simulation of incompressible channel flow at a Reynolds number of 180. The full domain has dimensions $(4\pi\delta,\ 2\delta,\ 2\pi\delta)$ and is discretized on a $256 \times 96 \times 256$ grid, where δ is the channel half width. A two dimensional slice is extracted from the central plane and represented on a 128×48 grid. The spatial coordinates are nondimensionalized by δ, and 10,000 snapshots of the streamwise velocity are collected. This dataset allows us to examine whether the proposed sensor placement strategy can identify informative locations and support accurate reconstruction in complex turbulent flows.

Aerofoil flow The aerofoil dataset is adopted to evaluate BASP under varying flow conditions and aerodynamic structures (Zhao et al., 2024a; Liu et al., 2025). It describes two dimensional incompressible flow around a NACA0012 aerofoil at a high Reynolds number. The data are generated using Reynolds averaged Navier–Stokes simulations. For each case, the angle of attack and inlet velocity are uniformly sampled within $\alpha \in [-10^\circ, 10^\circ]$ and $v_{\mathrm{in}} \in [-50\mathrm{m\,s^{-1}}, 50\mathrm{m\,s^{-1}}]$. The dataset contains 2,000 snapshots and a computational size of 256×256, with 1,400 snapshots used for training and the remaining 600 for testing. The velocity is selected as the reconstruction target in this study. This dataset allows us to assess whether BASP can identify informative sensor locations under diverse operating conditions and accurately reconstruct flow fields containing sharp gradients and complex wake patterns.

Heat conduction The two dimensional steady state heat conduction dataset is used to evaluate BASP on a nonflow physical field with localized heat sources and nonlinear thermal diffusion. The computational domain is a square region $\Omega = [0, 0.1] \times [0, 0.1]$, in which several Gaussian heat sources are distributed. A thermostatic heat sink with a length of 0.01 is located at the center of the bottom boundary, while the remaining boundaries are adiabatic. Different temperature fields are generated by varying the heat source intensity distribution and the temperature of the heat sink. The problem is solved using the finite element method on a 200×200 quadrilateral grid, producing a total of 6,000 temperature field samples. Of these, 4,200 samples are used for training and the remaining 1,800 for testing. This dataset is used to examine whether BASP can identify informative sensor locations and support accurate temperature field reconstruction under varying thermal conditions.

Table 4: Evaluation datasets and sensor budgets. The final column gives the sensor-count ranges used to pretrain the reconstructors.

Dataset / target	Grid	Evaluation budgets	Pretraining range
Cylinder / vorticity	192×112	10, 15, 20, 30	8–32
Channel / streamwise velocity	128×48	25, 50, 100, 200	25–300
Aerofoil / velocity	256×256	8, 12, 16, 20	8–24
Heat / temperature	200×200	10, 20, 30, 40	10–50

B RECONSTRUCTION MODELS

For each dataset, VoronoiCNN, RecFNO, PhySense, and POD-FM are pretrained independently for 800 epochs. Except for the modified VoronoiCNN, the reconstruction backbones and training configurations follow their original implementations. During pretraining, the sensor count is randomly sampled from the corresponding range in Table 4, exposing each reconstructor to varying numbers of observations. After pretraining, the same frozen checkpoint is reused across all placement methods and evaluated budgets within each dataset and reconstructor pair. The model architectures and adaptations to sensor input representations are described below.

VoronoiCNN. VoronoiCNN converts scattered sensor measurements into a structured representation suitable for CNNs (Fukami et al., 2021). In conventional Voronoi tessellation, each grid point receives the measurement of its nearest sensor. To support continuous sensor coordinates, we replace this hard assignment with the differentiable soft Voronoi formulation proposed in Xiao et al. (2026).

$$\widetilde{s}_{\mathrm{hard}}(\mathbf{z}_j) = s_{\arg\min_{1\le i\le m}\|\mathbf{z}_j-\mathbf{x}_i\|_2^2}, \qquad \text{Hard Voronoi.} \tag{8}$$

$$\widetilde{s}_{\mathrm{soft}}(\mathbf{z}_j) = \frac{\displaystyle\sum_{i=1}^{m} s_i \exp\left(-\alpha \left\|\mathbf{z}_j - \mathbf{x}_i\right\|_2^2\right)}{\displaystyle\sum_{i=1}^{m} \exp\left(-\alpha \left\|\mathbf{z}_j - \mathbf{x}_i\right\|_2^2\right)}, \qquad \text{Soft Voronoi.} \tag{9}$$

where $\mathbf{x}_i$ and s_i denote the coordinate and measurement of sensor i, $\mathbf{z}_j$ denotes a grid point, m is the number of sensors, and α controls the assignment sharpness. Rather than using a binary sensor mask, we construct a continuous sensor position map from Gaussian responses centered at the sensor coordinates. This map is concatenated with the soft Voronoi field along the channel dimension. The resulting fixed resolution representation retains both the interpolated measurement distribution and the sensor location information.

The model input comprises two fixed resolution channels containing the soft Voronoi field and the continuous sensor position map. A U-Net (Ronneberger et al., 2015) replaces the original shallow CNN as the reconstruction backbone and maps this representation to the reconstructed field $\widehat{\mathbf{u}}$. Because sensor sets with different sizes and layouts are converted into inputs with identical spatial dimensions during training, the model accommodates varying sensor numbers without architectural changes.

RecFNO. The Fourier neural operator (FNO) learns mappings between function spaces by parameterizing the kernel integral operator directly in the Fourier domain (Li et al., 2021). Its spectral convolution captures global spatial dependencies and reduces the dependence of the learned operator on a specific grid discretization. Building on this framework, RecFNO learns an end to end mapping from sparse observations to dense physical fields (Zhao et al., 2024b). In this study, only the Voronoi embedding is adopted. Consistent with the modified VoronoiCNN described above, the original hard Voronoi input is replaced by the same soft Voronoi rasterization and continuous sensor position map. These two fields are concatenated with the coordinate maps and projected into the initial feature space through a 1×1 convolution. The embedded features are subsequently processed by stacked Fourier layers according to

$$\mathbf{v}_{l+1} = \sigma\left(\mathbf{W}_l \mathbf{v}_l + \mathcal{F}^{-1}\left(\mathbf{R}_l \mathcal{F}(\mathbf{v}_l)\right)\right), \tag{10}$$

where $\mathbf{v}_l$ and $\mathbf{v}_{l+1}$ denote the input and output features of the lth Fourier layer, respectively; $\mathcal{F}$ and $\mathcal{F}^{-1}$ represent the Fourier and inverse Fourier transforms; $\mathbf{R}_l$ contains learnable complex weights acting on the retained Fourier modes; $\mathbf{W}_l$ is a learnable pointwise linear transformation implemented using a 1×1 convolution; and σ denotes a nonlinear activation function. The spectral branch captures global field structures, while the pointwise branch preserves local information. Stacking these operations enables RecFNO to approximate the nonlinear reconstruction operator. Because its learnable spectral parameters act on Fourier modes rather than individual grid points, the model can also transfer the learned mapping across different grid resolutions.

PhySense. Flow matching learns a time dependent velocity field that transports samples from a simple source distribution to a target data distribution along a continuous probability path (Lipman et al., 2023). PhySense applies this generative formulation to sparse physical field reconstruction by conditioning the transport process on sensor coordinates and measurements (Ma et al., 2025). Given a target field $\mathbf{u}_1$ and a Gaussian noise field $\mathbf{u}_0$, PhySense constructs the linear interpolation path

$$\mathbf{u}_t = t\mathbf{u}_1 + (1-t)\mathbf{u}_0, \tag{11}$$

where $t \in [0,1]$ denotes the interpolation time and $\mathbf{u}_t$ represents the intermediate field. The path begins with Gaussian noise $\mathbf{u}_0$ at $t = 0$ and reaches the target field $\mathbf{u}_1$ at $t = 1$. A conditional velocity model $\mathbf{v}\theta$ takes the intermediate field $\mathbf{u}_t$, time t, sensor measurements $\mathbf{s}^{(k)}$, and sensor coordinates $\mathbf{X}^{(k)}$ as inputs and learns the transport direction from $\mathbf{u}_0$ to $\mathbf{u}_1$. During inference, integrating the learned velocity field transforms a Gaussian sample into a physical field conditioned on $\mathbf{s}^{(k)}$ and $\mathbf{X}^{(k)}$.

For regular grid data, PhySense parameterizes the velocity model using a DiT (Peebles & Xie, 2023). The intermediate field is converted into field tokens, while the sensor coordinates and measurements are jointly encoded as conditioning tokens. Cross attention enables the field tokens to aggregate information from the sparse observations, whereas self attention captures global dependencies within the evolving field. Sensor numbers and locations are randomized during training so that a single reconstructor can accommodate different sensing configurations. Although the complete PhySense framework subsequently freezes the reconstructor and optimizes sensor coordinates, only its pretrained reconstruction model is retained in this study, with sensor configurations supplied by the placement methods being compared.

POD-FM. POD-FM integrates an improved gappy proper orthogonal decomposition (POD) module with flow matching to provide global physical guidance for sparse field reconstruction (Wang et al., 2026). POD applies singular value decomposition to the mean centered training snapshots and retains the first r dominant modes $\mathbf{\Phi}_r$, which provide a low dimensional representation of the principal spatial structures (Willcox, 2006). Using these modes, the physical field is approximated as

$$\mathbf{u} \approx \boldsymbol{\mu} + \mathbf{\Phi}_r \mathbf{a}, \tag{12}$$

where $\boldsymbol{\mu}$ is the mean field and $\mathbf{a}$ denotes the POD coefficient vector. Given k sensor measurements $\mathbf{s}^{(k)}$ at coordinates $\mathbf{X}^{(k)}$, gappy POD estimates the coefficients $\widehat{\mathbf{a}}$ by fitting this representation to the observations at the sensor locations. POD-FM further applies column normalization and adaptive regularization during coefficient estimation to improve numerical stability under sparse and noisy observations. Rather than using the resulting POD approximation as the final reconstruction, the estimated coefficients $\widehat{\mathbf{a}}$ are employed as a physically interpretable global prior that communicates the dominant field structures beyond the directly observed regions.

The estimated POD coefficients are encoded together with $\mathbf{s}^{(k)}$ and $\mathbf{X}^{(k)}$ to form the conditioning information for the generative reconstructor. The reconstruction component follows the same DiT based flow matching architecture as PhySense (Ma et al., 2025), where cross attention introduces the conditioning information into the evolving field representation and the learned velocity field transports Gaussian noise toward the target physical field. Random sensor layouts are used during training to accommodate different sensing configurations. Therefore, the principal distinction from PhySense lies in the additional POD coefficient guidance, which supplements local sensor observations with a compact global representation of the physical field.

C SENSOR PLACEMENT METHODS

D, A, and E optimality criteria These criteria formulate sensor placement as an optimal experimental design problem constructed from a POD basis Manohar et al. (2018); Nakai et al. (2021). The POD modes were computed from the training snapshots. For each sensor budget k, the first $r = k$ dominant modes were retained in $\mathbf{\Phi}_k$. Let $\mathcal{I}_j$ denote the set of locations selected after j steps. At selection step j, where $1 \leq j \leq k$, the sampled POD matrix and the observable information matrix are defined as

$$\mathbf{A}_{\mathcal{I}_j} = \mathbf{C}_{\mathcal{I}_j} \mathbf{\Phi}_k, \qquad \mathbf{F}_{\mathcal{I}_j} = \mathbf{A}_{\mathcal{I}_j} \mathbf{A}_{\mathcal{I}_j}^{\mathsf{T}}, \tag{13}$$

where $\mathbf{C}_{\mathcal{I}_j}$ is the sampling matrix that extracts the rows associated with the grid locations indexed by $\mathcal{I}_j$. Because $j \leq k = r$ throughout selection, only this row space formulation was used. Under independent homoscedastic Gaussian measurement noise and full row rank, D optimality maximizes $\log\det(\mathbf{F}_{\mathcal{I}_j})$, reducing the observable confidence region volume. A optimality minimizes $\mathrm{tr}(\mathbf{F}_{\mathcal{I}_j}^{-1})$, reducing the summed variance in the observable POD coefficient subspace. E optimality maximizes $\lambda_{\min}(\mathbf{F}_{\mathcal{I}_j})$, strengthening the least observable direction and limiting the largest directional variance. At $j = k = r$, all retained POD coefficients are observable. The D sequence used deterministic column pivoted QR applied to $\mathbf{\Phi}_k^{\mathsf{T}}$, which is equivalent to greedy log determinant maximization. The A sequence used exact greedy scoring with block inverse updates. The E sequence used forward greedy selection over all unused candidates. Each budget was processed independently, so layouts were not constrained to be nested. These criteria used only the POD modes and received no feedback from the downstream reconstruction model. The layouts were evaluated using the same pretrained reconstructor as the remaining placement methods, with all reconstructor parameters fixed. D selection uses column pivoted QR on the transposed POD basis, A selection uses exact greedy scoring with block inverse updates, and E selection uses forward greedy scoring over unused candidates. Each budget is processed independently. These layouts depend on the training POD basis and receive no feedback from the downstream reconstructor.

Random and uniform sampling Random sampling (RS) places sensors randomly within the computational domain. During evaluation, RS independently samples a new sensor layout for each test sample and reconstructs the field using measurements at those locations. Uniform sampling (US) distributes sensors regularly across the domain at the prescribed budget. Both methods determine sensor locations without reconstruction feedback.

FixedCoords FixedCoords directly parameterizes the sensor layout for a prescribed budget k as a trainable coordinate matrix $\mathbf{X}^{(k)} \in \mathbb{R}^{k\times 2}$ (Ma et al., 2025). It can be coupled with each of the reconstruction models considered here, while keeping all reconstructor parameters θ fixed. Let $\ell_\theta(u, \mathbf{X})$ denote the placement loss evaluated from a training field u and a sensor layout $\mathbf{X}$. This is the field prediction mean squared error for VoronoiCNN and RecFNO, and the flow matching velocity prediction loss for PhySense and POD-FM. For the latter two models, the expectations below also include the sampled interpolation time and Gaussian noise field. For each budget, FixedCoords minimizes

$$\min_{\mathbf{X}^{(k)}\in\Omega^k} \mathbb{E}_{u\sim\mathcal{D}_{\mathrm{train}}} \left[\ell_\theta \left(u, \mathbf{X}^{(k)} \right) \right], \tag{14}$$

where Ω specifies the coordinate bounds of the computational domain. Starting from an independently initialized layout, each optimization iteration samples measurements from the training fields at the current coordinates, constructs the sensor representation required by the reconstructor, and evaluates the placement loss. Gradients reach the coordinates through both the sampled measurements and the coordinate information supplied to the model. Adam updates the coordinate parameters, followed by projection to keep them within the computational bounds. The reconstructor weights are not updated. After optimization, the resulting coordinates are reused for all test fields at that budget. Each dataset, reconstructor, and budget combination has a separate optimization run. Consequently, changing the sensor count requires optimizing another layout, and the resulting coordinates need not preserve locations from a smaller budget. The BASP initialization and frozen prefix variants are additional controls described with the ablation studies.

C.1 BASP IMPLEMENTATION DETAILS

Construction of the physical field prior. Given the training fields $\{u^{(n)}\}_{n=1}^{N}$ after dataset preprocessing, we compute four statistical channels pointwise:

$$\begin{aligned}
\boldsymbol{\mu} &= \frac{1}{N}\sum_{n=1}^{N} u^{(n)}, \\
\boldsymbol{\sigma} &= \left[\frac{1}{N}\sum_{n=1}^{N}\left(u^{(n)}-\boldsymbol{\mu}\right)^2\right]^{1/2}, \\
\mathbf{a} &= \frac{1}{N}\sum_{n=1}^{N}\left|u^{(n)}\right|, \\
\mathbf{g} &= \left[\frac{1}{N}\sum_{n=1}^{N}\left\{\left(\delta_y u^{(n)}\right)^2+\left(\delta_x u^{(n)}\right)^2\right\}\right]^{1/2}.
\end{aligned} \tag{15}$$

These channels represent the mean, population standard deviation, mean absolute value, and root mean square gradient magnitude, respectively. Spatial differences δ_y and δ_x use unit grid spacing, with central differences at interior points and forward or backward differences at boundaries. Each statistical map is subsequently normalized by the spatial 99th percentile of its absolute values and clipped to $[-1, 1]$ for the mean or $[0, 1]$ for the other channels.

For grid indices starting at zero, the two coordinate channels are

$$C_y(i,j) = \frac{i}{n_y - 1}, \qquad C_x(i,j) = \frac{j}{n_x - 1}. \tag{16}$$

The four normalized statistical maps and two coordinate maps are concatenated to form $\mathbf{p} \in \mathbb{R}^{6\times n_y \times n_x}$. The prior is computed once from the training set and remains fixed during placement optimization. The physical field prior $\mathbf{p}$ is passed through a convolutional encoder consisting of three sequential blocks. Each block contains a 3×3 convolution with padding of one grid cell, followed by batch normalization and a ReLU activation. The output channel numbers are 64, 128, and 256, with convolution strides of 2, 2, and 1, respectively. The resulting feature map is

$$\mathbf{F} = \mathrm{Encoder}(\mathbf{p}) \in \mathbb{R}^{256\times h_f \times w_f}, \qquad h_f = \left\lceil \frac{n_y}{4} \right\rceil, \quad w_f = \left\lceil \frac{n_x}{4} \right\rceil. \tag{17}$$

The encoder parameters are learned during placement optimization, whereas the input prior remains fixed.

Learning ordered sensor queries. We represent the ordered sensor queries as a trainable matrix $\mathbf{Q} \in \mathbb{R}^{K\times d}$, where K is the maximum sensor budget and d was the channel dimension. Each entry of $\mathbf{Q}$ is independently initialized from a standard normal distribution. Before interacting with the spatial features, each query is transformed by a shared multilayer perceptron comprising two fully connected layers of width d, with a ReLU activation between them. For a sensor budget k, the first k queries are selected and processed in their index order to generate sensor locations. The query matrix, query projection, and encoder are jointly optimized using Adam through backpropagation of the reconstruction loss averaged over the sensor budgets selected at each training step, while the pretrained reconstruction network remains fixed. The query indices define the sensor order, and training with different budgets learns the corresponding prefixes of this shared query sequence.

D TRAINING AND EVALUATION PROTOCOL

All experiments involving VoronoiCNN and RecFNO are conducted on an NVIDIA RTX 4070 Ti SUPER GPU, while those involving PhySense and POD-FM use an NVIDIA RTX 5090 GPU. This includes reconstructor pretraining, sensor placement optimization, and evaluation. For each dataset and reconstructor pair, all placement methods use the same hardware and are evaluated with the same frozen reconstructor checkpoint.

Table 5: Sensor placement training settings. BASP trains PlaceNet once per dataset and reconstructor pair, supporting every integer sensor budget within the listed range. FixedCoords is optimized independently for each listed budget. Epochs refer to each training or optimization run, and # denotes the number of sensors.

Dataset	Method	Epochs	Batch size	Initial learning rate	Optimizer	Sensor budgets
Cylinder wake	BASP	200	32	5×10^{-4}	Adam	#8–32
	FixedCoords			1.0		#10, #15, #20, #30
Turbulent flow	BASP	200	32	5×10^{-4}	Adam	#25–300
	FixedCoords			1.0		#25, #50, #100, #200
Aerofoil flow	BASP	200	32	5×10^{-4}	Adam	#8–24
	FixedCoords			1.0		#8, #12, #16, #20
Heat conduction	BASP	200	32	5×10^{-4}	Adam	#10–50
	FixedCoords			1.0		#10, #20, #30, #40

Table 5 summarizes the sensor placement training settings and budgets. BASP uses cosine learning rate decay, and the FixedCoords learning rate follows its original open source implementation (Ma et al., 2025). During placement optimization, reconstructor parameters remain frozen, while gradients propagate through their sensor inputs. The D, A, and E optimality baselines are constructed from centered POD modes computed using the training snapshots, with the number of retained modes set equal to the corresponding sensor budget.

All placement methods within each dataset and reconstructor pair are evaluated using the same pretrained checkpoint, test partition, and sensor budgets listed in Table 4. Reconstruction accuracy is measured by the relative L_2 error:

$$\varepsilon = \frac{\|\hat{u} - u\|_2}{\|u\|_2}, \tag{18}$$

where u and $\hat{u}$ are the reference and reconstructed fields. For PhySense and POD-FM, this metric is computed from the final generated fields, while the flow matching velocity prediction loss is used for placement optimization.

In Tables 1 and 2, D/A/E (best) reports the smallest test error among the three separately evaluated optimality criteria for each dataset, reconstructor, and budget; individual results are provided in the extended results. Qualitative comparisons use the same frozen VoronoiCNN within each dataset, with errors above the field maps referring to the displayed samples. The component ablations and coordinate optimization controls are conducted on cylinder wake and aerofoil using the same frozen VoronoiCNN checkpoint as full BASP within each dataset.

Computational cost. Table 6 reports the estimated placement optimization times for VoronoiCNN. Each run uses 200 epochs, excluding reconstructor pretraining. We compare the cost of obtaining layouts for every integer budget in the full supported range. BASP is trained once per dataset and reconstructor pair, while FixedCoords requires a separate optimization for each budget. The total FixedCoords time is estimated by multiplying its mean time per budget by the number of supported budgets.

Across all four datasets, one BASP training run takes less time than the estimated total for FixedCoords over the full range. PlaceNet training takes substantially longer on turbulent flow because this dataset requires an ordered sequence of up to 300 sensors. The longer sequence involves more sequential attention and spatial suppression updates, increasing the computational cost of each training iteration.

E Extended reconstruction results

This section supplements Tables 1 and 2 with the individual results of the D, A, and E optimality criteria and additional comparisons of sensor layouts and spatial reconstruction errors. The quantita-

tive panels cover all four reconstructors and evaluated budgets. The qualitative panels use the same frozen VoronoiCNN within each dataset and show the two smallest evaluated budgets. They include E optimality, uniform sampling, and random sampling in addition to BASP and FixedCoords.

Table 6: Estimated placement optimization time in minutes with 200 epochs per run. BASP is trained once per dataset and reconstructor pair for all supported budgets. FixedCoords reports the mean time per budget and the estimated total across all supported budgets: 25 for cylinder wake, 276 for turbulent flow, 17 for aerofoil flow, and 41 for heat conduction. Time saved is the estimated FixedCoords total minus the BASP training time. Time reduction expresses this saving as a percentage of the estimated FixedCoords total.

Reconstructor	Dataset	BASP One run (min)	FixedCoords		Time saved (min)	Time reduction (%)
			Mean per budget (min)	Full range estimate (min)		
VoronoiCNN	Cylinder wake	46.99	16.64	416.00	369.01	88.70
	Turbulent flow	388.85	21.52	5939.52	5550.67	93.45
	Aerofoil flow	45.65	20.04	340.68	295.03	86.60
	Heat conduction	117.14	40.93	1678.13	1560.99	93.02

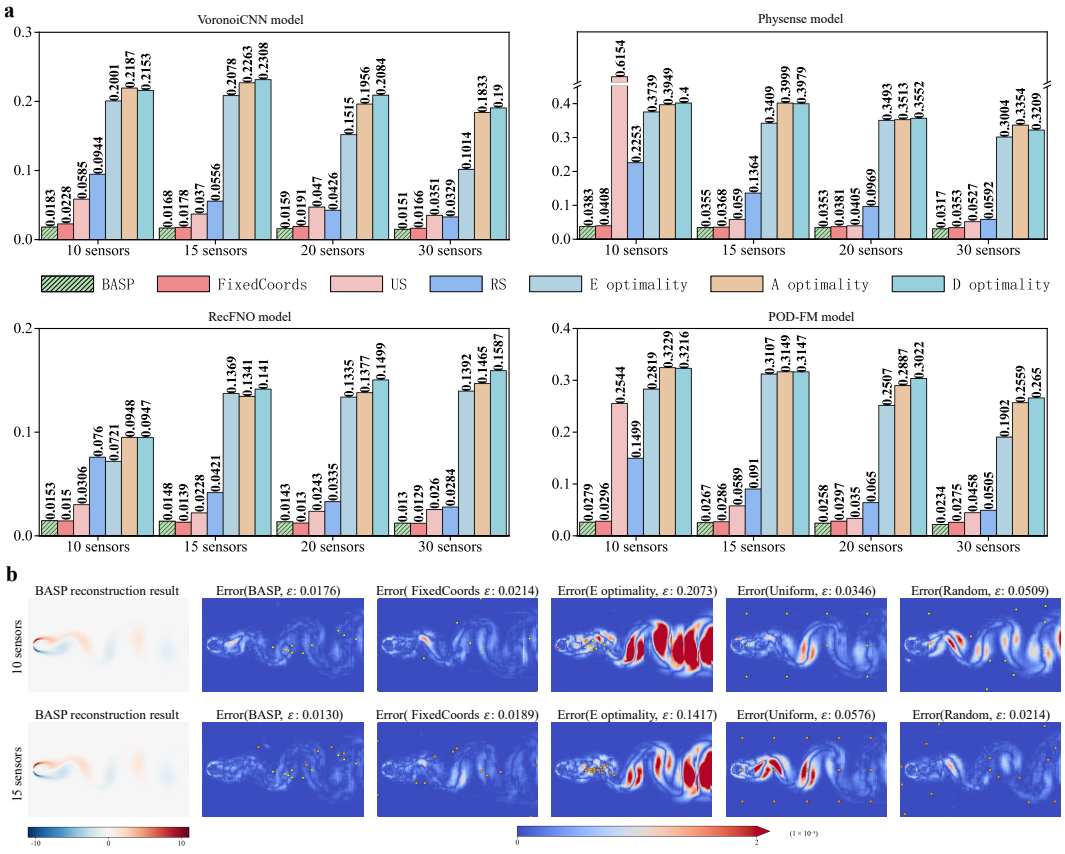

Figure 3: **Extended results on cylinder wake.** (a) Test relative L_2 errors of all seven placement methods with VoronoiCNN, PhySense, RecFNO, and POD-FM at budgets of #10, #15, #20, and #30. The PhySense panel uses a broken vertical axis.(b) BASP vorticity reconstructions in the first column and VoronoiCNN error maps for BASP, FixedCoords, E optimality, US, and RS at #10 and #15. Yellow circles mark lower-budget sensors; orange squares mark the five added BASP sensors or the complete higher-budget layouts of the other methods. The error ε above each map refers to the displayed sample.

E.1 CYLINDER WAKE

Figure 3 shows that the errors of the D, A, and E layouts do not consistently decrease as sensors are added, and remain above those of BASP across the evaluated reconstructors and budgets. BASP also outperforms FixedCoords for VoronoiCNN, PhySense, and POD-FM, while FixedCoords retains a small advantage for RecFNO. The spatial comparisons help explain the contrast between the placement strategies. E optimality concentrates observations close to the cylinder, leaving substantial errors farther downstream. BASP distributes sensors along the wake, while FixedCoords also places several sensors on the domain boundaries. The BASP extension from 10 to 15 sensors preserves the original locations and adds five complementary observations around the retained sensors. These examples illustrate how layouts derived from a POD criterion can differ from those favored by a neural reconstructor.

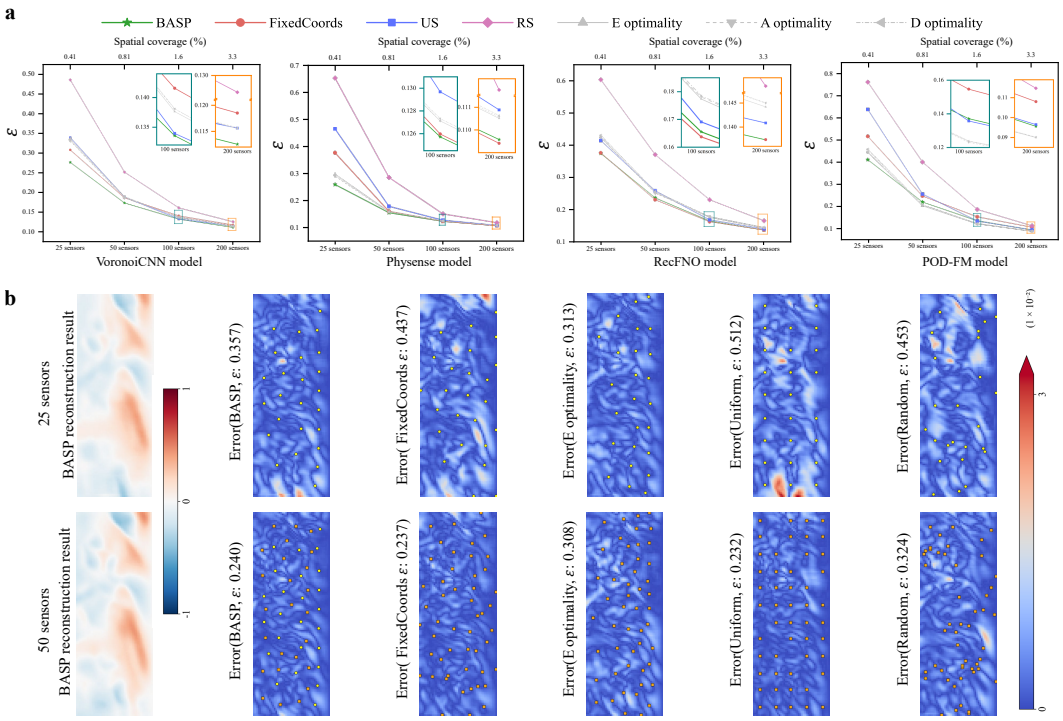

Figure 4: **Extended results on turbulent flow.** (a) Test relative L_2 errors of all seven placement methods with VoronoiCNN, PhySense, RecFNO, and POD-FM at budgets of #25, #50, #100, and #200. Insets enlarge the results at #100 and #200. (b) BASP streamwise velocity reconstructions in the first column and VoronoiCNN error maps for BASP, FixedCoords, E optimality, US, and RS at #25 and #50. Yellow circles mark lower-budget sensors; orange squares mark the 25 added BASP sensors or the complete higher-budget layouts of the other methods. The error ε above each map refers to the displayed sample.

E.2 TURBULENT FLOW

BASP achieves the lowest test error for all four reconstructors at 25 sensors, while the relative performance at larger budgets depends on the reconstruction model (Figure 4). For POD-FM, all three optimality criteria outperform BASP and FixedCoords at 50, 100, and 200 sensors. This may reflect the compatibility between selection from POD modes and the modal coefficients used to guide POD-FM. The layouts provide broad spatial coverage, with BASP inserting 25 additional sensors between retained locations when the budget increases to 50. The ranking in a representative field can differ from the aggregate ranking: E optimality has a lower error than BASP in the displayed 25-sensor example, despite its higher test error in the quantitative comparison. The aggregate results and representative maps therefore provide complementary views of placement performance.

E.3 AEROFOIL FLOW

The complete comparison in Figure 5 reveals a substantial gap between the three optimality criteria and the layouts optimized through reconstruction feedback. In the qualitative examples, E optimality concentrates sensors near the leading edge and in a narrow downstream region, while BASP and FixedCoords sample a broader portion of the surrounding flow and reduce the prominent errors around the aerofoil. The effect of additional sensors varies across reconstructors. For PhySense and POD-FM, both BASP and FixedCoords show little variation across budgets, whereas FixedCoords with VoronoiCNN continues to improve and yields lower errors than BASP at 16 and 20 sensors. The BASP extension from eight to 12 sensors places four additional sensors near several retained locations, producing only limited expansion of spatial coverage.

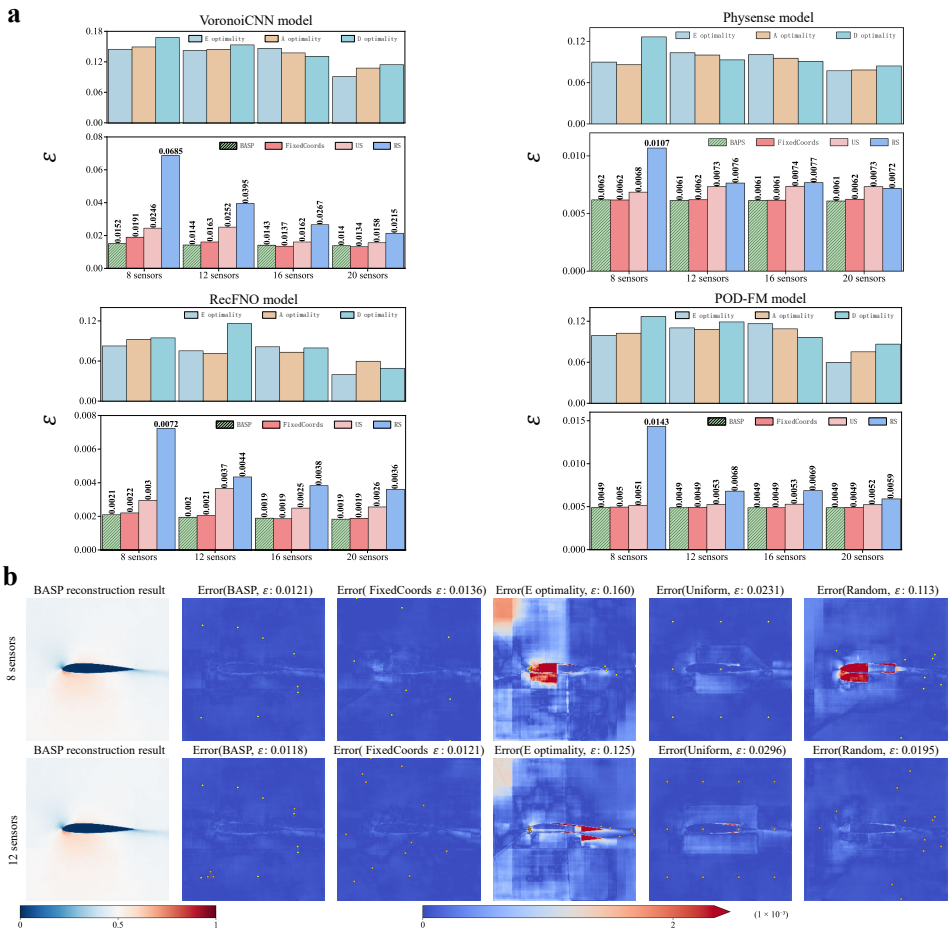

Figure 5: **Extended results on aerofoil flow.** (a) Test relative L_2 errors of all seven placement methods with VoronoiCNN, PhySense, RecFNO, and POD-FM at budgets of #8, #12, #16, and #20. The D, A, and E results are shown in separate upper panels because their errors occupy a different range. (b) BASP velocity reconstructions in the first column and VoronoiCNN error maps for BASP, FixedCoords, E optimality, US, and RS at #8 and #12. Yellow circles mark lower-budget sensors; orange squares mark the four added BASP sensors or the complete higher-budget layouts of the other methods. The error ε above each map refers to the displayed sample.

E.4 HEAT CONDUCTION

Figure 6 shows that BASP and FixedCoords outperform the conventional placement methods throughout the evaluated settings. Their errors are close, with FixedCoords performing better for RecFNO and BASP for PhySense at every tested budget. In the displayed 10-sensor case, uniform and random sampling leave larger errors near the heat sink at the lower boundary, particularly for

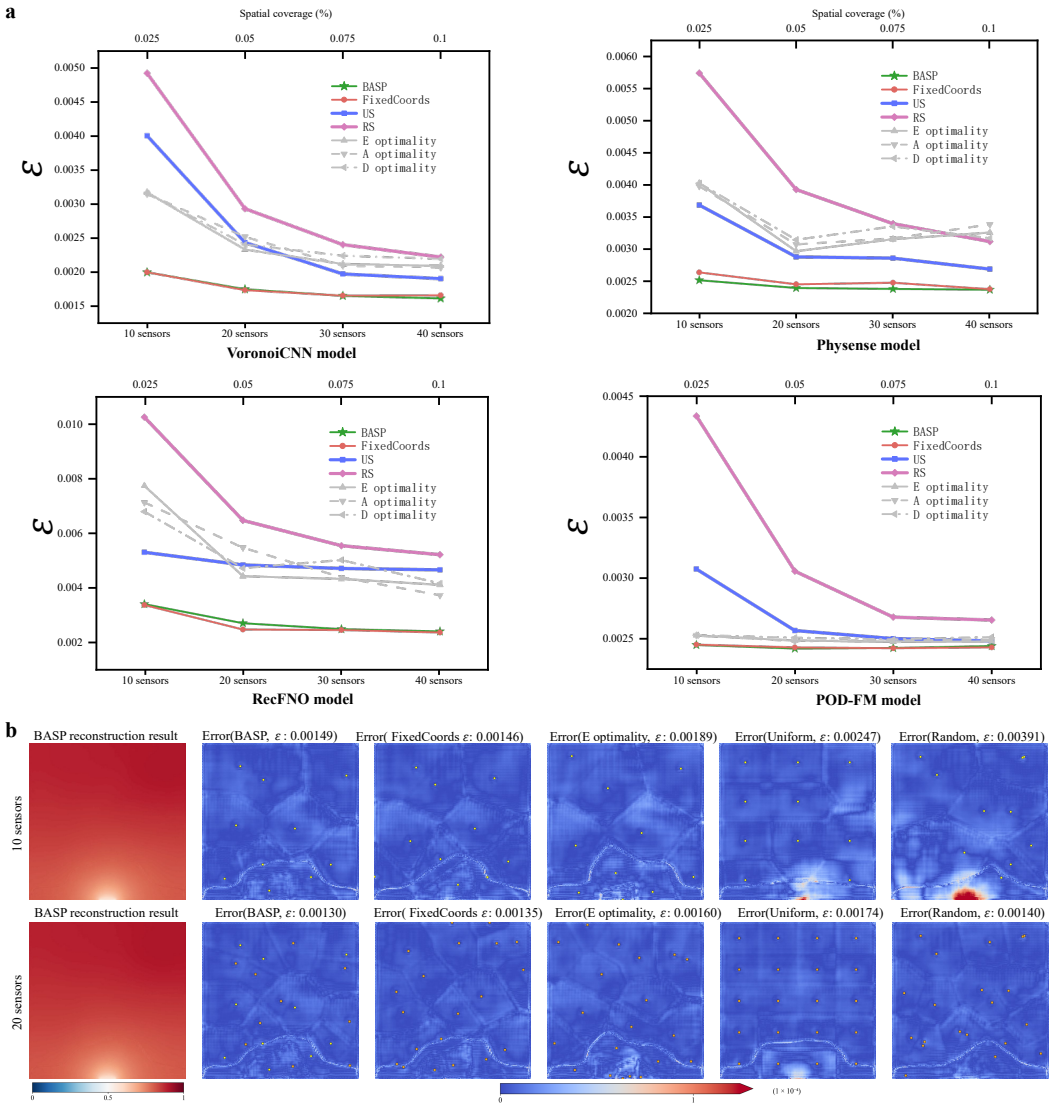

Figure 6: **Extended results on heat conduction.** (a) Test relative L_2 errors of all seven placement methods with VoronoiCNN, PhySense, RecFNO, and POD-FM at budgets of #10, #20, #30, and #40. (b) BASP temperature reconstructions in the first column and VoronoiCNN error maps for BASP, FixedCoords, E optimality, US, and RS at #10 and #20. Yellow circles mark lower-budget sensors; orange squares mark the ten added BASP sensors or the complete higher-budget layouts of the other methods. The error ε above each map refers to the displayed sample.

random sampling. BASP and FixedCoords combine broad coverage with measurements in the lower region influenced by the heat sink. Increasing the BASP budget from 10 to 20 preserves all existing sensors and fills gaps between them. This example illustrates how the learned sequence can increase spatial coverage while retaining the initial layout.

E.5 Sensor layouts from optimality criteria

Figure 7 complements the separate D, A, and E error curves by showing the layouts selected by each criterion at the two smallest evaluated budgets. Each layout is constructed from the training POD basis independently for its prescribed budget, without feedback from the neural reconstructor. Together with the reconstruction results above, these layouts allow the spatial distributions selected by the criteria to be examined alongside their performance across reconstruction architectures. No

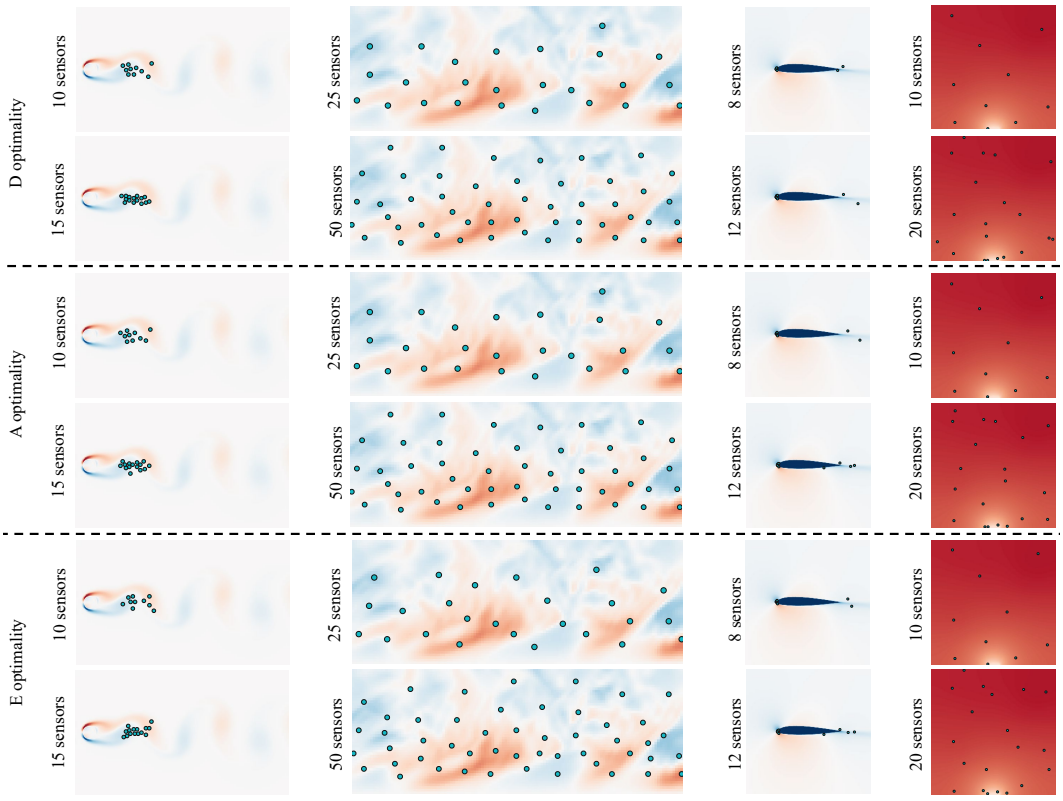

Figure 7: **Sensor layouts selected by the D, A, and E optimality criteria.** Columns show cylinder wake, turbulent flow, aerofoil flow, and steady state heat conduction from left to right. The upper, middle, and lower blocks correspond to D, A, and E optimality. Within each block, the two rows show #10 and #15 for cylinder wake, #25 and #50 for channel flow, #8 and #12 for aerofoil flow, and #10 and #20 for heat conduction. Markers indicate sensor locations overlaid on representative physical fields.

single optimality criterion consistently achieves the lowest reconstruction error across all datasets, reconstructors, and budgets.