Optimal Sensor Placement and Movement in Data Assimilation

by

Oleksandr Reshetar

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Mechanical and Mechatronics Engineering

Waterloo, Ontario, Canada, 2025

© Oleksandr Reshetar 2025

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Designing optimal locations for stationary sensors or optimal trajectories for moving sensors within a constrained sensor budget is crucial for Data Assimilation (DA) to reconstruct dynamical systems, such as ocean models or weather forecasts. Commonly used Lagrangian sensors for collecting observational data from the dynamical system may fail to capture important information due to their passive trajectory following the pathlines. This could lead to sensor clustering and undersampling of information-rich regions.

The first goal of this thesis research is to study the performance of Lagrangian sensors and potential improvements on a typical DA method called the Azouani-Olson-Titi (AOT) nudging algorithm. Two dynamical systems were used as testbeds: the one-dimensional Kuramoto-Sivashinsky equation (1D KSE) and the two-dimensional turbulent Navier-Stokes equations (2D NSE).

Computational experiments showed that, depending on the Stokes number of the sensors (e.g., from $St = 0$ for ideal Lagrangian sensors to $St = 1$ for realistic Lagrangian sensors with inertia), the clustering of the sensors degrades the performance of DA. However, introducing random perturbation to the ideal and realistic Lagrangian trajectories can achieve faster convergence for DA, and thus more effective reconstruction than their unperturbed counterparts. These observations suggest that ideal Lagrangian and inertia sensors may not be optimal, as even a simple random perturbation provides improvement. Therefore, it is reasonable to expect a better sensor movement strategy to achieve better convergence of DA.

The second goal is to propose an optimal sensor movement strategy that directs sensors toward information-rich regions of the flow. This is achieved by maximizing the convergence rate of the AOT algorithm, potentially yielding fast and effective reconstruction of a dynamical system. The thesis demonstrates that directed sensors can outperform both Lagrangian and inertia sensors based on AOT reconstruction, particularly in a sparse sensor scenario.

These findings can hopefully contribute to the DA community by showing that (i) ideal Lagrangian and inertia sensors are not necessarily good sensor movement strategies for reconstructing dynamical systems, and (ii) suggest a novel strategy to plan optimal sensor trajectories for moving sensors, which maximizes the convergence rate for the sequential DA.

Acknowledgements

I want to express my heartfelt gratitude to everyone who contributed to the completion of this thesis. First and foremost, I am deeply thankful to my supervisor, Dr. Zhao Pan, for his invaluable mentorship and insightful discussions. His guidance has profoundly shaped my academic and personal development.

I am also grateful to Dr. Jared Whitehead and Dr. Collin Victor for their significant contributions to this work. I am especially thankful to Collin for generously providing the source codes for the 1D KSE and 2D NSE models. My sincere thanks also go to Dr. Stephen Vavasis for his assistance in formal proof of the optimization problems, which laid a strong theoretical foundation for this thesis.

I extend my deepest appreciation to my fiancée, Sofiia, and my family for their unwavering support and encouragement throughout the most difficult moments. I would also like to acknowledge Compute Canada for providing the high-performance computing resources that enabled much of this work. Additionally, I would like to thank the initiative "Summer Program for Students from Ukraine" which provided a unique opportunity to experience Canadian academia and initiated a collaboration between me and Dr. Pan.

Lastly, I extend my sincere support to the people of Ukraine and the Ukrainian forces, whose enduring courage and resilience in the fight for freedom continue to inspire me in these challenging times.

Table of Contents

Author's Declaration ii

Abstract iii

Acknowledgements iv

List of Figures viii

1 Introduction 1

2 Background 5

2.1 Theoretical Setup of DA 5

2.2 Existing DA methods 6

2.2.1 Variational method 6

2.2.2 Statistical method 7

2.2.3 Azounai-Olson-Titi nudging algorithm 9

2.3 Optimal sensor placement 11

2.4 Sensor movement 12

2.5 Research objectives 15

3 Sensor placement and movement for 1D KSE **17**
3.1 Theoretical setup of Kuramoto–Sivashinsky equation 17
3.2 Error convergence analysis for 1D KSE 18
3.3 Minimization problem 20
3.4 Optimal $h(x)$ and the KKT conditions 23
3.5 Target sensor locations 26
3.6 Directed sensors 26
3.7 Lagrangian and the perturbed Lagrangian sensors 27

4 Sensor placement and movement for 2D NSE **29**
4.1 Theoretical setup of Navier-Stokes equations 29
4.2 Error convergence analysis for 2D NSE 30
4.3 Minimization problem 33
4.4 Optimal $h(x, y)$ and the KKT conditions 34
4.5 Lagrangian and the perturbed Lagrangian sensors 35

5 Results and Discussion **38**
5.1 1D Kuramoto–Sivashinsky equation 38
5.1.1 Numerical setup 38
5.1.2 Lagrangian and inertia sensors 39
5.1.3 Perturbed Lagrangian and perturbed inertia sensors 42
5.1.4 Directed sensors 43
5.1.5 Error convergence of sensor movement strategies 46
5.2 2D Navier-Stokes equations 47
5.2.1 Numerical Setup 47
5.2.2 Lagrangian and inertia sensors 48
5.2.3 Perturbed Lagrangian and perturbed inertia sensors 50
5.2.4 Directed sensors 53
5.2.5 Error convergence of sensor movement strategies 56

6 Conclusion **57**

6.1 Conclusions 57

6.2 Future Work 58

References **59**

APPENDICES **72**

A Determining target sensor locations **73**

List of Figures

3.1 A flow chart of finding target sensor locations based on the Algorithm 1. . 27

3.2 Clustering of Lagrangian sensors (black circles) in 1D KSE. Green arrows indicate regions where the flow moves to the right, while blue arrows indicate leftward motion. The red circle marks an example of a stagnation point. . 28

4.1 Lagrangian (purple) and perturbed Lagrangian (blue, $c = 0.5$) sensor trajectories. The black circle and black square indicate the start and end locations, respectively. 36

5.1 Error convergence with $N = 1024$ of uniformly initialized Lagrangian sensors. Panels (a) and (b) show the location of Lagrangian sensors at $t = 0$ and at $t = 20$, respectively, after clustering near stagnation points. Panel (c) illustrates the error behavior over time. For a visual demonstration of Lagrangian clustering in the steady KSE flow, access the QR code in (d). 39

5.2 The flow field of the 1D KSE overlaid with the trajectories of $N = 52$ uniformly initialized Lagrangian sensors (a) and zoomed-in view (b). 40

5.3 Error convergence with uniformly initialized inertia sensors. Panels (a–d) correspond to $N = 1024$, $N = 103$, $N = 52$, and $N = 33$, respectively. Lines with different colors are associated with various Stokes numbers, as shown in the legend in (a). 40

5.4 Error convergence with $N = 103$ of uniformly initialized inertia sensors with $St = 1$. Panels (a) and (b) show the locations of the inertia sensors at different times, denoted by colored triangles. The panel in the lower row illustrates the error behavior over time. 41

5.5 Average error convergence (red) with uniformly initialized perturbed Lagrangian sensors over fifty numerical experiments. Panels (a–d) correspond to ($N = 1024, c = 25$), ($N = 103, c = 40$), ($N = 52, c = 40$), and ($N = 33, c = 40$), respectively. 42

5.6 The flow field of the 1D KSE overlaid with the trajectories of $N = 52$ uniformly initialized perturbed Lagrangian sensors with $c = 5$ (a) and zoomed-in view (b). 43

5.7 The flow field of the 1D KSE overlaid with the trajectories of $N = 5$ uniformly initialized perturbed Lagrangian sensors with $c = 40$ (a) and zoomed-in view (b). 43

5.8 Average error convergence (red) with uniformly initialized perturbed inertia sensors ($St = 1$) over fifty numerical experiments. Panels (a–b) correspond to $N = 52$, and $N = 33$, respectively, with $c = 40$. 44

5.9 Error convergence of directed sensors (red, $v_p \approx 0.66$) with $N = 1024$ in (a), $N = 103$ in (b), $N = 52$ in (c), and $N = 33$ in (d). In both dense and sparse sensor scenarios, directed sensors achieved convergence of error to machine precision. 45

5.10 Error convergence rate comparison between the directed sensors (red, $v_p \approx 0.66$) and the designed rate (black) in (3.7) with $N = 1024$ in (a), and $N = 103$ in (b). 45

5.11 The flow field of the 1D KSE overlaid with the trajectories of $N = 33$ directed sensors. 46

5.12 Initial stage of the error convergence of sensor movement strategies with $N = 1024$ in (a), $N = 103$ in (b), $N = 52$ in (c), and later stage with $N = 33$ in (d). Lines with different colors are associated with various sensor movement strategies, as shown in the legend in (a–b). 47

5.13 Error convergence of Lagrangian (blue) and inertia sensors (red) with $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), and $N = 196$ in (d). To see the motion of Lagrangian sensors in turbulent flow, access the QR code in (e). 49

5.14 Error convergence for inertia sensors with $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c) and $N = 196$ in (d). Lines with different colors are associated with various St, as shown in the legend in (a). To see the clustering of inertia sensors with $St = 0.1$, access the QR code in (e). 50

5.15 Lagrangian (magenta) and perturbed Lagrangian sensors with $c = 0.25$ (cyan) in (a), and $c = 4$ (green) in (b). The black circle and square indicate the starting and ending locations, respectively. 51

5.16 Inertia sensors (marked with black dots) with $St = 0.1$ and $N = 196$. Panel (a) illustrates perturbed inertia sensors with noise amplitude $c = 2$, while panel (b) demonstrates a case of larger noise amplitude, $c = 32$. 51

5.17 Error convergence with perturbed Lagrangian sensors and $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), and $N = 196$ in (d). Lines with different colors are associated with various amplitudes, as shown in the legend in (b). 52

5.18 Average error convergence with perturbed Lagrangian sensors (red) over the ten numerical experiments with $N = 900$ and $c = 2$ in (a), and $c = 4$ in (b). 52

5.19 Error convergence with perturbed inertia sensors ($St = 0.1$) and $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), and $N = 196$ in (d). Lines with different colors are associated with various amplitudes, as shown in the legend in (b). 53

5.20 Error convergence with directed sensors, moving towards target locations, determined by the Algorithm 1 with $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), $N = 196$ in (d). 54

5.21 Error convergence with Algorithm 1 and designed error convergence rate, R^*, with $N = 2500$ in (a), and $N = 196$ in (b). The blue and red lines illustrate the actual and designed convergence rates, respectively. 54

5.22 Error convergence with Lagrangian, perturbed Lagrangian ($c = 4$), inertia ($St = 0.1$), perturbed inertia ($St = 0.1, c = 32$) and directed sensors with $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), $N = 196$ in (d). Lines with different colors are associated with various sensor movement strategies, as shown in (a). 55

Chapter 1

Introduction

Reconstructing and predicting dynamical systems pose fundamental challenges across various scientific fields. These challenges arise from imperfect physical models, which are of limited value in predicting the long-term behavior of dynamical systems unless corrected periodically with data from observations. Alternatively, using only observations, which are often noisy and sparse in time and/or space, does not reconstruct and predict the dynamical system in the best possible way. Therefore, combining models with observations could achieve better reconstruction and prediction of the dynamical system [5].

Data Assimilation (DA) is a framework that achieves an optimal reconstruction and prediction of dynamical systems by combining physical models and sensor data [5, 44]. This fusion of the information is critical in many applications. For instance, in weather modeling, satellite images and measurements from weather stations are integrated with atmospheric models to produce weather forecasts [103]. In oceanography, ocean circulation models are combined with observations from in situ ocean drifters (i.e., buoys or Argo floats) to forecast ocean dynamics, including ocean currents and temperature [5]. In biology, synthetic brain tumor images are integrated with mathematical models of tumor evolution to predict the growth and spatial spread of malignant brain cancers [75]. These practical applications showcase the broad applicability of the DA framework in reconstructing and predicting a wide range of dynamical systems.

The effectiveness of the DA often depends on the quality and quantity of sensor data. DA can typically achieve effective reconstruction of the dynamical system with noise-free, abundant sensor data. However, such data quality and quantity are often unattainable in real-world scenarios. For instance, the quality of data is often limited due to measurement noise and errors [5]. Additionally, the quantity of data can be limited when sensors are

expensive to manufacture, maintain, or deploy [74]. Another challenge with data quantity could be the high-dimensionality of the dynamical system, such as the ocean [114], where achieving dense sensor coverage is almost impossible. Therefore, careful planning and optimizing sensor placement are necessary to achieve an effective DA reconstruction with noisy and sparse sensors.

Optimal Sensor Placement (OSP) is a classic task in planning and optimizing sensor locations that maximize the informativeness of sensor data for a given dynamical system [45, 64, 73, 116, 127]. This is typically achieved by optimizing a performance criterion, such as mutual information or system observability [76], to improve state estimation, prediction, or control accuracy [91]. The outcome of the resulting optimization is typically a set of stationary sensor locations that capture most of the key features of the dynamical system.

While OSP typically focuses on finding stationary sensor locations, they are not necessarily optimal for dynamical systems in all scenarios, and in such cases, mobile sensors could improve DA reconstruction. Being stationary can be viewed as a constraint for sensors, limiting their observational area. This limited observational area could miss key features of a dynamical system (which are information-rich to a DA algorithm), particularly when sensors are sparse and the dynamical system is chaotic. Therefore, removing the stationary constraint and allowing the sensors to move could improve the DA reconstruction [54, 73].

Lagrangian sensors are a natural strategy for moving sensors to collect data from dynamical systems. These sensors are often used in fields such as oceanography or weather modeling, taking the form of drifting ocean buoys or weather balloons [31, 38, 66, 105]. Using Lagrangian sensors has the following benefits. First, Lagrangian sensors have low manufacturing costs, making their mass deployment practically feasible to collect more observational data [95]. Second, they passively follow the flow of dynamical systems. This passive movement requires minimal energy for locomotion and often provides intuitive measurements [54]. For example, Lagrangian sensors can effectively track the trajectories of substances or organisms in the ocean, such as oil spills or fish larvae, that are transported by ocean currents [15]. Finally, observing the collective motion of many Lagrangian sensors, one can intuitively identify Lagrangian Coherent Structures (LCS), which are the foundational geometrical features in the flow [60, 61]. These structures are often helpful to reveal skeletons in the flow (e.g., shear layers and/or eddies, and transport barriers) [23]. Due to these benefits, Lagrangian sensors are often used to study transport, stirring, mixing in fluid flows [102], and in DA problems to reconstruct dynamical systems [30, 93, 129].

Although Lagrangian sensors are a low-cost solution to collect observational data, they come with several limitations. First, following the flow (e.g., along pathlines of moving

fluids), but not necessarily the key features, might miss important information about the dynamical system [79]. This missed information could be essential for effective DA reconstruction. Second, the nature of Lagrangian sensors makes them sensitive to their initial locations [61, 94], which can lead to inconsistent DA performance. An example demonstrating this inconsistency is when these sensors are deployed near LCS transport barriers, such as attracting or repelling structures. For instance, Lagrangian sensors initialized near repelling regions may drift into entirely different flow domains. In contrast, those placed near the same attracting areas tend to cluster, resulting in redundant observations over time.

Finally, ideal Lagrangian trajectories are often impractical, as realistic Lagrangian sensors carry inertia, potentially causing undesired behavior. For instance, in two-dimensional turbulent flows, sensors with even small inertia commonly exhibit clustering in regions characterized by high strain and low vorticity [39, 106]. This clustering may reduce the diversity of information available to DA, as some regions (e.g., high-strain regions) may become oversampled, while others (e.g., high-vorticity regions) remain unobserved.

This thesis investigates how the above limitations of Lagrangian and sensors with inertia (i.e., following the flow and clustering, but not necessarily observing the key features) impact the performance of a typical DA method, called the AOT algorithm, in reconstructing dynamical systems. Next, to address these limitations, this work proposes alternative sensor movement strategies that improve the AOT performance. One such strategy involves adding a random perturbation to both Lagrangian sensors and sensors carrying inertia with random noise, which could prevent clustering. If this random perturbation facilitates a more effective AOT algorithm, it would imply that ideal Lagrangian and sensors with inertia are not necessarily optimal. Another strategy focuses on finding optimal trajectories for moving sensors to collect flow data from information-rich regions. This is achieved by maximizing the convergence rate of the AOT algorithm, thereby achieving fast and effective reconstruction of dynamical systems.

The rest of the thesis is structured as follows. Chapter 2 reviews the background on DA, including the AOT algorithm, and existing sensor placement and movement strategies. Chapter 3 develops a theoretical framework for guiding sensors to follow key features of the one-dimensional chaotic partial differential equation (PDE) called the Kuramoto–Sivashinsky equation (1D KSE). Chapter 4 extends the framework from Chapter 3 to the two-dimensional turbulent Navier-Stokes equations (2D NSE). Both Chapters 3 and 4 propose adding random perturbations to ideal Lagrangian and inertia-driven trajectories, which serve as an intriguing example of why Lagrangian trajectories may not be the best choice for DA. Chapter 5 validates proposed sensor movement strategies and provides a comparison among them. The thesis concludes with Chapter 6, which outlines future

developments in the topic of optimal sensor placement and sensor movement strategies for achieving effective DA reconstruction.

Chapter 2

Background

This chapter first reviews the general mathematical formulation of the DA framework for reconstructing dynamical systems. Then, it provides a brief overview of two classical and one state-of-the-art implementation of DA. This chapter also reviews optimal sensor placement/movement problems, solving which might benefit the DA performance in reconstructing dynamical systems under a constrained sensor budget. Lastly, it discusses traditional sensor movement strategies, such as Lagrangian and inertia sensors, and emphasizes how their passive movement could inhibit a successful DA reconstruction.

2.1 Theoretical Setup of DA

To motivate the use of DA, consider a dynamical system governed by the following PDE:

$$\begin{aligned} \frac{du}{dt} &= F(u) \\ u(x,0) &= u_0 \end{aligned} \quad , \tag{2.1}$$

where u denotes the system's state variable evolving in time t, F represents the system dynamics (typically known), and u_0 is the unknown initial conditions. In principle, if the exact initial conditions u_0 were known, one could reconstruct the true dynamical system (2.1) without DA [83]. However, precise knowledge of u_0 is rarely available in practical applications.

To compensate for this lack of knowledge, sparsely distributed (in space and time) observations of the state u are incorporated into the DA framework. These observations

are defined through the mapping

$$u_o = \mathbf{H}[u] + \epsilon_o,$$

where $\mathbf{H}$ is the observation operator, which selects the locations where u is observed. ϵ_o denotes a white noise process with zero mean. For simplicity, this thesis assumes sensor data are observed without noise (i.e., $\epsilon_o = 0$) and continuous in time, as in [54, 89].

Once the observations are incorporated into the DA framework, a successful assimilation process converges the discrepancy between the system estimate $\hat{u}$ and observations to zero as time goes to infinity:

$$\epsilon(t) = \|\mathbf{H}[\hat{u}(t)] - u_o(t)\| \to 0, \quad \text{as } t \to \infty. \tag{2.2}$$

After the projected system estimate $\mathbf{H}[\hat{u}]$ has converged to the observations u_o in (2.2), one could hope that the system estimate $\hat{u}$ converges to the true dynamical system u.

The above convergence rate of $\epsilon(t)$ reflects how the error between the estimate and observations behaves over time, which is an indicator of DA performance. This rate often depends on several factors, including the profile of the dynamical system, the system's dimensionality, the specific DA method used, and the placement or trajectories of sensors. While the first two factors are typically fixed for a specific problem, the latter, such as the DA method and sensor location/trajectories, leave room for manipulation to achieve better DA performance. Existing DA methods will be briefly reviewed in Section 2.2, while the main focus of this thesis, sensor placement/movement strategies, will be discussed in detail in Sections 2.3–2.4.

2.2 Existing DA methods

2.2.1 Variational method

The variational method frames DA as an optimization problem grounded in optimal control theory and variational calculus. The core idea of the variational method is to define a cost function that quantifies the discrepancy between the system estimate and observational data [5]. This cost function is then minimized iteratively to obtain the trajectory of the estimate that best fits the available observations.

Among the most widely used variational methods are three-dimensional variational data assimilation (3D-Var) [87] and four-dimensional variational data assimilation (4D-Var) [120]. The 3D-Var formulation minimizes the cost function at a single analysis time,

yielding a static correction of the estimated state. In contrast, 4D-Var extends the assimilation over a time window by incorporating the full temporal evolution of the model and observations, allowing the state estimate to reflect time-dependent dynamics. This incorporation makes 4D-Var especially valuable in numerical weather prediction, where capturing time-dependent processes is essential.

Although the variational method could achieve effective DA reconstruction, it faces significant computational challenges and is not optimal in all scenarios. These computational challenges arise primarily from minimizing the cost function by computing adjoint models, which becomes particularly demanding in strongly nonlinear, high-dimensional systems, posing a barrier to their use in real-time applications [125]. Additionally, classical variational methods, such as 3D-Var and 4D-Var, assume that both observation and model errors are Gaussian. However, errors in nonlinear dynamical systems are often non-Gaussian, which could degrade the performance of DA [19].

2.2.2 Statistical method

The statistical method of DA is grounded in Bayesian inference, which aims to maximize knowledge of the posterior probability distribution of the system estimate. Since having full knowledge of the probability distribution is typically infeasible, statistical methods determine the best approximation of the statistical parameters of the system, such as its mean and/or its variance. This is often achieved through combining prior estimates and new observational data to minimize the uncertainty about the posterior system estimate [3, 5]. One of the classical methods that combines time-varied measurements with prior system estimates is the Kalman Filter (KF) [72].

The KF provides an analytically optimal solution for linear systems with Gaussian-distributed errors. It minimizes the posterior error covariance, yielding a statistically optimal estimate $\hat{u}$ of the true system state u through recursive assimilation of observations u_o. Specifically, this recursive assimilation consists of two steps:

$$\text{Forecast: } \hat{u}_{l+1}^f = F_{l+1}(\hat{u}_l),$$

where $\hat{u}_{l+1}^f$ is forecast of the state at time step $l+1$ and

$$\text{Update: } \hat{u}_{l+1} = \hat{u}_{l+1}^f + \mathbf{K}\left[u_o - \mathbf{H}\hat{u}_{l+1}^f\right], \tag{2.3}$$

where $\hat{u}$ is the system estimate, and $\mathbf{K}$ is the Kalman gain matrix. The gain matrix in the above equation controls the relative weighting of the system estimate and the observations: greater weight is given to the observations when they are more accurate than the

system estimate and vice versa. Thus, the matrix $\mathbf{K}$ is designed to optimally balance the contributions of $\hat{u}^f$ and u_o, minimizing the overall uncertainty in the reconstructed state.

Although the classical KF computes the optimal gain matrix in linear low-dimensional systems under Gaussian noise, its performance is limited in more complex scenarios. These limitations include high computational demands for high-dimensional systems, restrictions to linear systems, and a reliance on the assumption of Gaussian noise. For instance, KF calculates the optimal gain matrix by a set of matrix operations, such as multiplication and inversion, which become computationally intractable for high-dimensional systems. Additionally, the assumptions of a linear system and Gaussian noise are rarely satisfied in practical applications involving complex geophysical or environmental processes [126]. In such processes, the filter may fail to accurately capture the nonlinear interactions or non-Gaussian error structures inherent in the system, leading to suboptimal state estimates.

Several extensions of the KF have been developed to handle nonlinear systems and reduce computational demands. For instance, the first attempt to handle nonlinear systems was proposed in [67, 117] by developing the Extended Kalman Filter (EKF). The core idea of EKF is to linearize the system dynamics around the current estimate. This is done by approximating the nonlinear functions with their first-order Taylor series expansion, replacing them with linearized models at each update step. However, this linear approximation can break down when the time between updates is long, allowing the model's nonlinear behavior to evolve in ways the linearized version cannot capture [5]. Additionally, similar to the KF, EKF operates with full-rank covariance matrices, which could become expensive in high-dimensional systems. Lastly, the EKF typically assumes the model and measurement error covariance matrices are Gaussian, which may be unrealistic in strongly nonlinear environments.

Particle Filters (PF) were later proposed to handle strongly nonlinear scenarios and non-Gaussianity. These filters approximate the full posterior distribution of dynamical systems by propagating a set of particles through the model and weighting them according to their consistency with the observations [78]. However, the performance of PF strongly depends on the number of particles; a large number is typically required for effective reconstruction in high-dimensional or highly nonlinear systems, which may significantly increase computational cost.

To address the computational challenges of high-dimensional systems, the Ensemble Kalman Filter (EnKF) was introduced in [42]. The EnKF can be viewed as a reduced-order version of the KF and, like the PF, represents the system state using a collection of state vectors, or ensemble members. The variability of these ensemble members implicitly captures the forecast error statistics, eliminating the need to propagate large covariance

matrices. This significantly reduces computational demands in high-dimensional settings. However, the accuracy of the EnKF degrades when the ensemble size is too small or when the system exhibits strong nonlinear behavior [27, 43, 130]. Increasing the ensemble size can better represent nonlinear effects, but at the cost of higher computational expense.

2.2.3 Azounai-Olson-Titi nudging algorithm

Since minimizing a cost function in the variational method and having a large number of particles/ensembles in statistical methods could be computationally demanding, the nudging method—also known as Newtonian relaxation—was introduced in [4]. This method adds a penalty term to the evolution model of the system estimate, based on the discrepancy between the observations (i.e., input data) and the estimate, analogous to the Luenberger observer or the Kalman filter in control theory [72, 88]. Specifically, the nudging looks for a solution $\hat{u}$ to the equation:

$$\frac{d\hat{u}}{dt} = F[\hat{u}] + \kappa(I_h[u_o] - \hat{u}), \tag{2.4}$$

where κ is a constant nudging parameter that is proportional to the model-observation misfit, and I_h is an operator that interpolates observations on the same grid space as $\hat{u}$ with a spatial distance h between the locations of observation. This equation can be viewed as a simplified version of KF in (2.3), where the gain matrix is replaced with a constant parameter κ, which significantly reduces computational costs. Additionally, instead of applying the discrete updates on the system estimate $\hat{u}$ in (2.3), which occur only when new observations become available, the nudging method applies a continuous correction that steers the estimate smoothly toward the observational data.

The main limitation of the conventional nudging equation (2.4) is that it allows observations to influence the estimate across the entire spatial domain, including regions where no observations are available. This might cause inconsistent convergence between nudging and the true solution (2.1). In particular, even if the estimate $\hat{u}$ matches the true solution u at some time t_k, the nudging equation will still evolve differently from the true system because of the extra term $\kappa(I_h(u_0) - \hat{u}) \neq 0$. To avoid this problem, u_0 and $\hat{u}$ must be mapped and compared in the same observation space, ensuring that when $u(t_k) = \hat{u}(t_k)$, the evolution of both systems remains the same [27].

A recently proposed Azounani-Olson-Titi (AOT) algorithm solves the above problem [2, 6, 7, 16, 25, 32, 36, 46, 47, 53, 69, 81, 97], by introducing a feedback control term that

applies nudging only at observed locations:

$$\frac{d\hat{u}}{dt} = F[\hat{u}] + \kappa(I_h\{u_o\} - I_h\{\hat{u}\}), \tag{2.5}$$

where $I_h(u)$ and $I_h(\hat{u})$ are now compared on same resolution. Introducing the interpolation operator also enabled a rigorous proof showing that (2.5) is guaranteed to converge to important nonlinear systems, such as KSE, NSE, or Allen-Cahn equations, given large enough κ and sufficiently small h (or equivalently, for a sufficiently dense sensor distribution) [7, 82, 89].

The AOT algorithm is perhaps the simplest form of DA that can effectively reconstruct the dissipative dynamical system [122]. Dissipative means that the energy of the dynamical system decreases over time due to mechanisms such as friction, viscosity, or diffusion. These dissipative mechanisms are essential for damping errors in small spatial scales (i.e., high Fourier modes) between the AOT algorithm and the dynamical system. As a result, the long-term behavior of the dynamical system is ultimately determined by the large spatial scales (i.e., low Fourier modes) as small spatial scales die out. Even if small scales persist and continue transferring energy to larger scales, a process known as the inverse energy cascade, the latter ones ultimately determine the fundamental structure of the flow [99]. Thus, the goal of the successful AOT algorithm is to observe these large spatial scales from the true solution and to integrate them properly in the AOT equation (2.5) for the effective reconstruction of the dynamical system.

The success of an AOT-based reconstruction depends on the proper selection of the following terms: the nudging parameter κ, the interpolation operator I, and the number of sensors and their locations and/or trajectories, which influences h, and the behavior of I_h. For instance, a small nudging parameter may not exhibit enough forcing to nudge the system estimate $\hat{u}$ to the true solution u. In contrast, choosing a large nudging parameter could cause numerical instabilities, which arise from injecting energy into small-scale errors. This injection may overcome system dissipative forces, which dampen these small-scale errors, thereby preventing the AOT solution from converging to the reference solution [7].

Once the nudging parameter κ is appropriately selected to inject enough forcing into the AOT equation in (2.5) and to avoid numerical instabilities, the focus of designing a good AOT algorithm could shift to employing a good interpolation operator I. The choice of the interpolation operator is subject to the approximation property [7], which provides a bound on the error between an arbitrary function and its interpolation:

$$\|f - I_h(f)\|^2_{L^2(\Omega)} \leq c_0 h^2 \|f\|^2_{H^1(\Omega)}, \tag{2.6}$$

where $f \in H^1(\Omega)$, c_0 is a dimensionless constant and I_h are linear interpolants from space $I_h : H^1 \to L^2$. Commonly used linear interpolants such as piecewise interpolation [54], cubic, spline, or projection on low Fourier modes [1, 55] guarantee the error convergence in (2.2) given large enough κ and small enough h. Non-linear interpolation operators do not provide such a guarantee theoretically; however, in numerical simulations, the error could still decay [26, 80].

While finding the optimal nudging parameter and interpolation operator is an important task to improve the AOT performance, they are often not effective if the sensor data is sparse and/or does not provide important information about the dynamical system; hence, designing a sensor location/trajectory that captures important information essential to improve the performance of the AOT algorithm. Designing better sensor locations/trajectories is often done in OSP and sensor movement frameworks that place and move sensors to locations that contain rich information about the dynamical system. This placement/movement is particularly critical under a constrained sensor budget. These OSP and sensor movement methods are discussed in Section 2.3 and Section 2.4, respectively.

2.3 Optimal sensor placement

OSP is a classic task to determine optimal sensor locations in many scientific fields, such as controls [21, 29, 119, 123], where sensor locations impact observability and controllability of the system, or signal reconstruction [74, 86, 91], where sensor locations affect the accuracy of state estimation. The goal of OSP is to select a subset of available sensor locations that capture the important information about the dynamical system. However, finding optimal sensor locations could be computationally challenging in high-dimensional systems and hence remains an open problem.

Computational challenges in finding optimal sensor locations arise from the combinatorial nature of selecting sensor locations: for a system with n possible measurement locations and only N available sensors, the brute force approach must evaluate

$$\binom{n}{N} = \frac{n!}{N!(n-N)!}$$

possible sensor configurations [92, 96, 104]. Therefore, using a brute force approach to achieve OSP in high-dimensional scenarios is often infeasible.

Instead of exhaustive search, most existing OSP methods use greedy algorithms or low-rank representations of the dynamical system to obtain a near-optimal solution [96]. These

methods typically optimize submodular (or near-submodular) criteria of the surrogate objective function, such as observability [73], condition number [91, 131], determinant [109], or frame potential [104]. Other studies also focused on minimizing error metrics, including root mean square error [124] and reconstruction error [85]. More recent works also applied neural networks to determine sensor locations for improved flow reconstruction [40, 128]. The concepts behind these OSP methods are often transferable across scientific fields and have already been applied to DA problems for system reconstruction and/or parameter estimation [14, 63, 68, 124].

While most OSP methods may achieve a good reconstruction of the dynamical system in a certain sense, they typically rely on a set of stationary sensors. Being stationary can be viewed as a constraint for sensors, as it restricts their observational location and potentially misses key features of the dynamical system that evolve or move over time. This limitation becomes severe when sensors are sparse, and the underlying dynamical system is chaotic. Therefore, removing this stationary constraint and allowing sensors to move could collect more information about the dynamical system, thereby improving the AOT performance [54].

2.4 Sensor movement

Designing optimal trajectories for moving sensors is a central challenge across many scientific fields, including robotics and estimation [17, 22, 57], as well as in tracking fluid flows [37, 84]. This challenge typically arises from two aspects: the high dimensionality of the system, which complicates its modeling, and the fact that estimation or reconstruction quality depends directly on the sensors' trajectories. Consequently, planning optimal paths requires jointly considering system dynamics and sensor trajectories, resulting in a complex optimization problem that is often difficult to implement in practice. While the former is often specific to the application, the latter enables the manipulation of simple trajectories to improve system reconstruction. These trajectories might include using circular or elliptical motion, motion along level curves [84], clothoids, or spline-based paths [56]. Other approaches construct trajectories to maximize information gain or observability [90, 96], although they often incur high planning costs. However, equipping sensors with motors to follow planned trajectories could increase operational costs, limiting large-scale deployment of sensors. As such, one might want to use inexpensive, low-maintenance sensors with minimal planning and propulsion requirements while deploying many of them to achieve sufficient observation of the dynamical system.

A classic approach with zero "trajectory planning" cost is the Lagrangian sensors.

These sensors move passively with the flow of the dynamical system and collect intuitive measurements along their trajectories. In fluid dynamics, this passive movement corresponds to the advection of the fluid flow, which is governed by

$$\frac{dx}{dt} = u(x(t), t), \tag{2.7}$$

where x denotes the spatial coordinates of the sensor at time t, and u is the flow velocity field. Due to natural motion with the flow, zero planning and low manufacturing and maintenance costs, Lagrangian sensors are widely used in fluid dynamics applications in the form of, for example, ocean buoys and/or weather balloons [31, 38, 66, 105].

Despite the low cost and ease of deployment, the passive nature of Lagrangian sensors imposes several limitations. For instance, their passive trajectories are not necessarily optimal for observing important features of the dynamical system, such as vortex cores [79], that could degrade the performance of the DA. Additionally, passive trajectories of Lagrangian sensors make them sensitive to initial locations, which could prevent consistent and effective DA reconstruction when Lagrangian sensors are initialized at different positions before the reconstruction process.

The sensitivity of Lagrangian sensors to initial conditions is well studied in the context of Lagrangian Coherent Structures (LCS) [60, 61, 111]. LCS appear as ridge-like features in finite-time Lyapunov exponent (FTLE) fields and act as material transport barriers—either attracting or repelling nearby trajectories. For example, two sensors initialized near a repelling LCS will diverge exponentially over time, observing entirely different flow regions. Conversely, sensors distributed near the same attracting LCS may converge and cluster, providing redundant observation over time. This issue alone suggests that DA, including the AOT algorithm that relies on Lagrangian sensors, can suffer from inconsistent performance, particularly when the sensor network is sparse.

Another limitation is that achieving ideal Lagrangian sensors is actually challenging, as real sensors often carry inertia. These inertia sensors may cluster and oversample some areas, while leaving important regions undersampled, thereby hindering effective DA reconstruction [12]. The tendency for such clustering can be characterized by the Stokes number (St).

The Stokes number is a dimensionless parameter that quantifies the sensor's interaction with the flow:

$$St = \frac{t_0 u^*}{l_0},$$

where t_0 represents the sensor's relaxation time (i.e., the timescale over which the sensor adjusts its velocity to match that of the fluid), u^* is a characteristic velocity of the fluid,

and l_0 is a characteristic length scale of the flow. The relaxation time t_0 is defined as:

$$t_0 = \frac{\rho_p d_p^2}{18\nu},$$

where ρ_p is the sensor's density, d_p its diameter, and ν the kinematic viscosity of the fluid. This relaxation time assumes a spherical sensor operating in the Stokes flow regime (i.e., low Reynolds number), where viscous forces dominate over inertia. Higher relaxation times are associated with greater sensor mass and/or diameter, as well as larger Stokes numbers, indicating a greater deviation of the sensor's trajectory from the surrounding fluid (i.e., Lagrangian trajectory).

The Stokes number is a critical parameter for indicating how strongly inertia sensors cluster in turbulent flows, a phenomenon known as particle clustering [12], preferential concentration [106], or segregation [24]. At low Stokes numbers ($St \ll 1$), sensors exhibit little inertia effect and closely follow the surrounding fluid (i.e., Lagrangian sensors with $St = 0$), which could reduce clustering [106].

In contrast, high Stokes numbers ($St \gg 1$) indicate that a sensor's motion is dominated by its inertia, making it less responsive to the dynamics of the carrier fluid. Such sensors with high inertia tend to follow nearly ballistic trajectories that are largely decoupled from the flow streamlines. This pronounced decoupling becomes crucial in practical applications, where sensors have only a finite time to interact with coherent flow features, such as eddies, before those structures are diffused or distorted by surrounding features. However, sensors with high inertia move slowly, thus lagging in interacting with flow structures while they exist. As a result, sensors with large Stokes numbers are less prone to clustering [12, 39, 41].

Clustering is pronounced in turbulent flows when the Stokes number is of order one ($St \sim \mathcal{O}(1)$) [11, 12, 28, 41, 62, 106]. In this scenario, inertia sensors are typically expelled from rotational flow structures (vortices) and cluster in high-strain regions. This can lead to over-sampling strain regions and under-sampling vortices, potentially degrading DA performance.

The passive behavior of Lagrangian sensors and clustering of inertia sensors (i.e., both might not track key features of the flow) highlights the importance of exploring alternative sensor movement strategies to improve the performance of the DA framework, including the AOT algorithm. For instance, the study in [54] proposed *ad hoc* sensor movement strategies that were validated in 2D turbulent Navier-Stokes equations. These strategies include: *bleeps*, where sensors randomly appear at different locations in the domain at each time step; *creeps*, which are stationary (i.e., Eulerian) sensors that randomly shift on the grid (i.e., slow random walk); and *sweeps*, where a smaller sub-grid of sensors

moves following a prescribed pattern, for example, a straight line. The study in [54] demonstrated that the moving sensors could collect more informative data compared to Lagrangian sensors, significantly increasing the convergence rate of the AOT algorithm. These results suggest that Lagrangian sensors are not optimal for AOT and may also limit the performance of other DA methods.

While *bleeps, sweeps,* and *creeps* achieved faster error convergence than Lagrangian sensors, their trajectories were heuristic and *ad hoc*, which were not built on any analytical or theoretical ground. As a result, despite being refreshing and inspiring, this work does not offer a universal framework for determining optimal trajectories for moving sensors.

Although a sensor movement strategy derived from theory was later proposed in [18], it is limited in achieving the fastest error convergence of the AOT algorithm. Specifically, the study introduced a moving observation window with uniform sensor distribution inside it. Then, this window moves with a certain speed towards high-error regions, where the error between the AOT reconstruction and the underlying dynamical system is large. By observing high-error regions, the proposed sensor moving strategy achieved a much faster convergence speed compared to the random movement of observational windows. This suggests that relocating sensors to the high error location could significantly enhance the effectiveness of the AOT algorithm. However, the locations of sensors were restricted to a finite-sized moving observation window, typically much smaller than the full physical domain. As a result, key features of the system outside this window may remain unobserved, preventing the method from achieving the best possible initial error convergence.

2.5 Research objectives

Given the limitations of passive trajectories of Lagrangian sensors and inertia sensors, as well as the lack of a theoretical framework for guiding sensor movement to achieve optimal convergence rate for DA in previous studies, this thesis pursues two primary goals:

- investigate ideal and perturbed trajectories[1] of Lagrangian and inertia sensors under the AOT algorithm for reconstructing dynamical systems in dense and sparse sensor settings;
- proposes a theoretically grounded sensor movement strategy to observe the key features of the dynamical system by maximizing the convergence rate of the AOT algorithm.

[1]Note that the perturbation is added to the Lagrangian and inertia trajectories to prevent potential clustering, while the sensors continue to measure the flow without noise.

The performance of the above sensor movement strategies was evaluated on two dynamical systems: the one-dimensional Kuramoto–Sivashinsky equation (1D KSE) and the two-dimensional Navier–Stokes equations (2D NSE). The 1D KSE, a canonical example of low-dimensional chaotic dynamics, serves as a simplified testbed for developing and validating the proposed sensor movement strategies. In contrast, the 2D NSE, which is more representative of real-world systems such as atmospheric flows, is used as a benchmark to assess the effectiveness of these strategies in a more complex environment.

Chapter 3

Sensor placement and movement for 1D KSE

This chapter presents an optimal sensor movement strategy that guides sensors to observe information-rich regions of the 1D KSE. It also discusses a random perturbation applied to both Lagrangian and sensors with inertia that could prevent clustering and improve AOT performance. After a brief overview of the KSE in Section 3.1, Section 3.2 revisits the error convergence analysis of the AOT algorithm and provides a novel guideline for optimal sensor placement and movement. Next, based on this error convergence analysis, Section 3.3 formulates a minimization problem that maximizes the convergence rate of the AOT algorithm on 1D KSE. Then, Sections 3.4–3.5 discuss how the solution to the minimization problem yields optimal sensor locations. Section 3.6 reviews how physical sensors will be implemented to reach their optimal sensor locations. Lastly, Section 3.7 covers a random perturbation applied to Lagrangian sensors.

3.1 Theoretical setup of Kuramoto–Sivashinsky equation

The KSE is a simple nonlinear PDE that exhibits chaotic behavior. In 1D, it is written as

$$\begin{gathered} u_t + u_{xx} + \nu u_{xxxx} + \frac{1}{2}(u^2)_x = 0 \\ u(x,0) = u_0(x) \end{gathered}, \tag{3.1}$$

where $u \in C([0,T], L^2) \cap L^2([0,T], H^2)$ is a real-valued function with admissible initial condition $u_0 \in L_0^2([0,L])$ [9, 89], evolving on a domain with length L and time T, and ν is kinematic viscosity. This PDE and its chaotic properties were studied by Kuramoto et al., when they discovered the emergence of spatially periodic patterns and temporal instabilities in reaction-diffusion systems [77]. Independent discovery of this PDE was led by Sivashinsky, who described how the KSE captured the nonlinear growth and interaction of small perturbations on a flame front, causing chaotic behavior [113]. This chaotic behavior, which even exists in one dimension, makes the reconstruction of KSE an appealing toy problem for the DA studies, as the corresponding results are often applicable to other chaotic systems, such as fluid flows and weather models [112, 115].

Although the KSE has chaotic behavior, it remains fundamentally dissipative. This dissipative nature exhibits an asymptotic behavior of the KSE and features a finite-dimensional global attractor [8, 34, 101, 108, 121]—a set towards which all solutions of (3.1) evolve regardless of initial conditions. Because this global attractor is finite-dimensional, the long-term behavior of the KSE is determined by a finite number of degrees of freedom, such as finite Fourier modes [49, 50, 70], finite elements [33, 50], finite interpolation polynomials [33, 50], determining volume elements [51, 71], or determining nodes [49, 50]. Observing sufficient degrees of freedom with enough sensors guarantees the convergence of the AOT algorithm to a true KSE solution [89].

However, under a constrained sensor budget, capturing all determining modes might be infeasible; thus, the convergence of error cannot be guaranteed. In this sparse sensor scenario, one can hope to approximate at least some determining modes by strategically moving sensors to observe information-rich regions. This can be achieved through the error convergence analysis of the AOT algorithm, which guides sensor movement to maximize the convergence rate, as discussed in Section 3.2.

3.2 Error convergence analysis for 1D KSE

In this section, the error convergence analysis of the AOT algorithm is reviewed from [89] with $L = 2\pi$, which guides sensor movement to maximize the convergence rate. Applying the AOT algorithm to (3.1) yields

$$\begin{aligned} \hat{u}_t + \hat{u}_{xx} + \nu \hat{u}_{xxxx} + \frac{1}{2}(\hat{u}^2)_x + \kappa I_h(\hat{u} - u) = 0 \\ \hat{u}(x,0) = \hat{u}_0(x) \end{aligned}, \tag{3.2}$$

where κ is the nudging parameter, and I_h is the spatial interpolation with h being the distance between sensors. Defining the error $\epsilon = \hat{u} - u$ and subtracting (3.1) from (3.2) leads to

$$\epsilon_t + \nu\epsilon_{xxxx} + \epsilon_{xx} = -\epsilon\epsilon_x - u\epsilon_x - \epsilon u_x - \kappa I_h(\epsilon). \tag{3.3}$$

Multiplying (3.3) by ϵ, integrating by parts with respect to x using periodic boundary conditions and using the fact that

$$I_h(\epsilon)\epsilon = \epsilon(I_h(\epsilon) - \epsilon) + \epsilon^2$$

results in

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}\int_0^{2\pi} \epsilon^2 \, dx + \nu \int_0^{2\pi} \epsilon_{xx}^2 \, dx = \|\epsilon_x\|_{L^2}^2 + \frac{1}{2}\int_0^{2\pi} \epsilon^2 u_x \, dx - \\ -\kappa\|\epsilon\|_{L^2}^2 + \kappa \int_0^{2\pi} (\epsilon - I_h(\epsilon))\epsilon \, dx. \end{aligned} \tag{3.4}$$

Applying Poincaré, Cauchy-Schwarz, Young, and Agmon $\left(\|u\|_{L^\infty}^2 \leq \|u\|_{L^2} \|u_x\|_{L^2}\right)$ inequalities as in [89], the expression in (3.4) further simplifies to

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}\|\epsilon\|_{L^2}^2 + \nu\|\epsilon_{xx}\|_{L^2}^2 \leq \frac{\nu}{8}\|\epsilon_{xx}\|_{L^2}^2 + \\ + \frac{\nu}{4}\|\epsilon_{xx}\|_{L^2}^2 + \left[\frac{2}{\nu} - \kappa + \|u_x\|_{L^\infty} + \frac{3}{4}\left(\frac{\kappa^4 c_0 h^4}{\nu}\right)^{1/3}\right]\|\epsilon\|_{L^2}^2, \end{aligned} \tag{3.5}$$

where $\|u_x\|_{L^\infty}$ represents the supremum of the velocity gradient over the domain. Grouping the terms with the same error norm ~~and without using assumptions in Eq. (15) in [89]~~, the expression (3.5) results in an inequality in terms of the reconstruction error

$$\frac{1}{2}\frac{d}{dt}\|\epsilon\|_{L^2}^2 \leq -\frac{5\nu}{8}\|\epsilon_{xx}\|_{L^2}^2 + \underbrace{\left(\frac{2}{\nu} - \kappa + \|u_x\|_{L^\infty} + \frac{3}{4}\left(\frac{\kappa^4 c_0 h^4}{\nu}\right)^{1/3}\right)}_{r(h,x)}\|\epsilon\|_{L^2}^2, \tag{3.6}$$

which dictates that the error is mainly controlled by the distance h and the nudging parameter κ, as all other terms are given constants, including the viscosity ν of the flow.

The upper bound of (3.6) resembles the ordinary differential equation (ODE) for $\|\epsilon\|_{L^2}^2$. This ODE shows that the error must decay exponentially if h and κ are properly selected such that the coefficient of the second term on the right-hand side (i.e., $r(h, x)$) in (3.6) is

negative:

$$r(h,x) = \left(\underbrace{\frac{2}{\nu} - \kappa + \|u_x\|_{L^\infty}}_{a(x)} + \underbrace{\frac{3}{4}\left(\frac{\kappa^4 c_0 h^4}{\nu}\right)^{1/3}}_{bh^{\frac{4}{3}}} \right) < 0, \tag{3.7}$$

where $r(h,x)$ can be viewed as a global decay rate of the error.

The value of $r(h,x)$ determines the error convergence behavior of the AOT reconstruction. If $r(h,x) < 0$, the error of the AOT reconstruction is guaranteed to converge exponentially. The more negative r is, the faster the convergence occurs. If $r(h,x) \geq 0$, it is not guaranteed that the AOT reconstruction converges. However, a positive convergence rate ($r(h,x) > 0$) does not necessarily always lead to a diverged AOT algorithm. This is because the error estimate of (3.6) is not sharp, meaning that the error convergence might still occur in practice even for small positive values of r. The smaller the r is (i.e., closer to 0), the more likely the AOT reconstruction converges; and if it converges, the convergence rate is likely higher.

Therefore, regardless of the sign of the error convergence rate, minimizing $r(h,x)$ should lead to better convergence. The task of minimizing the error decay rate $r(h,x)$ provides a guideline for optimal sensor placement and/or movement. This minimization problem is discussed in Section 3.3.

3.3 Minimization problem

The convergence rate defined in (3.7) can be interpreted pointwisely, since the integrals in (3.4) may be evaluated over arbitrarily small subdomains. This interpretation enables switching from global to local convergence rates, where the global gradient norm $\|u_x\|_{L^\infty}$ is replaced by the pointwise magnitude $|u_x|$. This switch also enables the distance term h to be viewed as a spatially varying function $h(x)$, indicating the spatial density of the sensor, which can be used to construct an adaptive grid and thus, guide sensor placement/movement.

Although the local convergence rate in each subdomain impacts the global convergence rate, the latter is ultimately governed by the maximum local rates. To illustrate this governance, consider partitioning the original domain Ω into a set of subdomains. Under this

partitioning, the global error can be expressed as an integral of the local errors multiplied by their convergence rate factors:

$$\|\epsilon(t)\|_\Omega \leq \int_\Omega e^{r(h,x)t} \|\epsilon_0\| \, dx, \tag{3.8}$$

where ϵ_0 is an initial error in subdomains and $e^{r(h,x)t}$ is their exponential rate factor. The global error behavior is expressed as

$$\|\epsilon(t)\|_\Omega = e^{k(t)t} \|\epsilon_0\|_\Omega \,, \tag{3.9}$$

where $k(t)$ is a time-varying scalar. Solving (3.8) for k from (3.9) yields the relationship between the global error convergence rate and local ones:

$$k(t) \leq \frac{1}{t} \log \left(\frac{1}{\|\epsilon_0\|_\Omega} \int_\Omega e^{r(x,t)t} \|\epsilon_0\| \, dx \right).$$

For a finite time t, the scalar $k(t)$ reflects contributions from all subdomains. However, as $t \to \infty$, $k(t)$ approaches $\max r(h,x)$, because the region with the largest local error growth $e^{r(h,x)}$ (i.e., sensitive region) eventually overwhelms the contributions from other areas.

Therefore, the maximum value of local rates $r(h,x)$ must be minimized. The minimization problem is formulated to find the distance function $h(x)$ that solves

$$\min_h \max_x a(x) + bh(x)^{\frac{4}{3}} \tag{3.10}$$

with a constraint on the number of sensors N:

$$\int_\Omega \frac{1}{h(x)} dx = N, \tag{3.11}$$

where $1/h(x)$ is the density of sensors over the entire domain.

The solution to the above minimization problem is uniquely determined, and no lower minimum can be found. To illustrate this fact, suppose $\mathcal{H}_N$ defines an admissible set of distance functions:

$$\mathcal{H}_N := \left\{ h \in L^\infty(\Omega) \middle| h(x) > 0, \int_\Omega \frac{1}{h(x)} dx = N \right\}.$$

The objective function in (3.10) can be rewritten as

$$\min_{R,h} R,$$

where $R := \sup_x a(x) + bh(x)^{\frac{4}{3}}$ such that

$$a(x) + bh(x)^{\frac{4}{3}} \leq R. \tag{3.12}$$

From (3.12), $h(x)$ can be found as

$$h(x) \leq \left(\frac{R - a(x)}{b} \right)^{\frac{3}{4}}. \tag{3.13}$$

Defining the upper bound of (3.13) as

$$\phi_R(x) = \left(\frac{R - a(x)}{b} \right)^{\frac{3}{4}}$$

and substituting the above function into (3.11) yields

$$G(R) := \int_\Omega \frac{1}{\phi_R(x)} dx = \int_\Omega \left(\frac{b}{R - a(x)} \right)^{\frac{3}{4}} dx,$$

where $G(R)$ is a continuous function for $R \neq a(x)$ and strictly decreasing with R:

$$\lim_{R \to \sup a(x)} G(R) = \infty \tag{3.14}$$

$$\lim_{R \to \infty} G(R) = 0. \tag{3.15}$$

By the intermediate value theorem

$$\exists R^* : G(R^*) = N \Rightarrow \int_\Omega \frac{1}{\phi_{R^*}(x)} dx = N, \tag{3.16}$$

from which the optimal distance function can be found:

$$h^*(x) := \phi_{R^*}(x) = \left(\frac{R^* - a(x)}{b} \right)^{\frac{3}{4}}. \tag{3.17}$$

Given the optimal distance function derived above, we now show by contradiction that it is the unique solution. Suppose that the pair (R^*, h^*) is not the minimum solution to (3.10), implying that

$$\exists h'(x) \in \mathcal{H} : \sup r_{h'}(x) < R^* \Rightarrow r_{h'}(x) \leq R' < R^*. \tag{3.18}$$

However, this leads to a contradiction, as using the definition

$$h'(x) \leq \left(\frac{R' - a(x)}{b}\right)^{\frac{3}{4}} = \phi_{R'}(x),$$

$h'(x)$ would need more sensors than N:

$$R' < R^* \Rightarrow \int_\Omega \phi_{R'}(x)^{-\frac{3}{4}} > \int_\Omega \phi_{R^*}(x)^{-\frac{3}{4}} \Rightarrow G(R') > G(R^*) = N.$$

The contradiction of having multiple minima shown above proves that the error convergence rate R^* is a unique solution; hence, all that is needed to get a global minimum R^* is to find the distance function $h(x)$. However, searching for the optimal distance function in an infinite-dimensional function space is a challenging task. Instead of this search, the distance function $h(x)$ can be attained through the following equivalence claim: minimizing the maximum among the local error rates (3.10) is equivalent to requiring $r = \text{const}\ \forall x \in \Omega$, that is

$$\min_h \max_x a(x) + bh^{\frac{4}{3}} \quad \Leftrightarrow \quad \min_h r(h, x) \text{ where } r(h, x) = \text{const} \quad \forall x \in \Omega. \tag{3.19}$$

Proving this equivalence claim is performed using the Karush–Kuhn–Tucker (KKT) conditions.

3.4 Optimal $h(x)$ and the KKT conditions

The KKT conditions are fundamental in constrained optimization, offering necessary (and in convex settings, sufficient) criteria for identifying optimal solutions [13]. These conditions are typically used to search for a solution to optimization problems with both equality and inequality constraints, which is the case with the optimization problem posed in (3.10). We next show that applying the KKT conditions to (3.10) can prove the equivalence claim presented in (3.19).

Although the KKT conditions are optimal for convex and smooth optimization problems, the minimization problem formulated in (3.10) is neither convex due to the $1/h$ term in inequality constraints nor smooth due to the inner max in the objective function. First, to transform this minimization problem into a convex one, the $h(x)$ function is replaced with the function $p(x)$ defined as

$$p(x) = \begin{cases} \frac{1}{h(x)} & h(x) > 0 \\ \infty & h(x) \leq 0 \end{cases}, \tag{3.20}$$

where $p \mapsto p^{-\frac{4}{3}}$ is a convex function in $\mathbb{R} \to \mathbb{R} \cup \{\infty\}$. Replacing $h(x)$ with $p(x)$ in (3.10) yields

$$\begin{aligned} &\min_{p} \max_{x} a(x) + bp(x)^{-\frac{4}{3}}, \\ &\text{s.t.} \quad \left\{ \int_{\Omega} p(x)\ dx = N \right\}. \end{aligned} \tag{3.21}$$

Introducing a slack variable $R := \sup_{x \in \Omega} a(x) + bp(x)^{-\frac{4}{3}}$, (3.21) transforms to

$$\begin{aligned} &\min_{R,p} R \\ &\text{s.t.} \quad \left\{ a(x) + bp(x)^{-\frac{4}{3}} \leq R, \int_{\Omega} p\ dx = N \right\}. \end{aligned} \tag{3.22}$$

Once the smooth and convex optimization problem in (3.22) is obtained, the KKT conditions can be applied to prove the equivalence claim (3.19). For simplicity, this thesis proves the equivalence claim in a discrete setting, although the proof can be generalized to infinite-dimensional convex optimization over the Hilbert space [10, 20]. The corresponding discretized setting is

$$\begin{aligned} &\min_{R,\mathbf{p}} R \\ &\text{s.t.} \quad \left\{ a_i + b_i p_i^{-\frac{4}{3}} \leq R, \sum_i p_i = N \right\}. \end{aligned} \tag{3.23}$$

To find R from the above formulation, the following KKT conditions are applied: there exist Lagrangian multipliers $\mu \in \mathbb{R}$ and $\lambda \in \mathbb{R}^n$ such that

- Primal feasibility: $\sum_i p_i = N, a_i + bp_i^{-\frac{4}{3}} \leq R \quad \forall i = 1, \ldots, n$;
- Dual feasibility: $\lambda_i \geq 0$;
- Complementarity: $\lambda_i (R - a_i - bp_i^{-\frac{4}{3}}) = 0$,

are satisfied. These conditions define the Lagrangian cost function, which is then minimized by setting its gradient to 0. First, writing the Lagrangian cost function yields

$$\mathcal{L}(p, \lambda, \mu) = R + \sum_i \lambda_i (a(x_i) + bp_i^{-\frac{4}{3}} - R) + \mu \sum_i (p_i - N). \tag{3.24}$$

Next, finding optimal p requires solving the system of equations formed by setting $\nabla_{R,p_i}\mathcal{L} = 0$:

$$\begin{pmatrix} \frac{\partial \mathcal{L}}{\partial R} \\ \frac{\partial \mathcal{L}}{\partial p_1} \\ \vdots \\ \frac{\partial \mathcal{L}}{\partial p_n} \end{pmatrix} = 0 \Rightarrow \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} \lambda_1 + \lambda_2 + \ldots + \lambda_n \\ \frac{4}{3}\lambda_1 b p_1^{-\frac{7}{3}} \\ \vdots \\ \frac{4}{3}\lambda_n b p_n^{-\frac{7}{3}} \end{pmatrix} + \begin{pmatrix} 0 \\ \mu \\ \vdots \\ \mu \end{pmatrix}. \tag{3.25}$$

Note that $\mu = 0$ in (3.25) is not feasible, as it implies $\lambda_i = 0$, contradicting the first component of the gradient equation. Therefore $\mu \neq 0$, implying $\lambda_i \neq 0$, as function $p > 0$. Since $\lambda > 0$, the complementarity condition from the KKT conditions:

$$\lambda_i (R - a_i - b p_i^{-\frac{4}{3}}) = 0$$

simplifies to

$$(R - a_i - b p_i^{\frac{3}{4}}) = 0,$$

leading to $a_i + b p_i^{\frac{3}{4}} = R$, proving the equivalence claim (3.19).

A simple heuristic for the above equivalence proof using the KKT conditions can clarify why it works. Suppose such p^* (and therefore $h^*(x)$) is found such that $R^* = \min_p \max_x r(h^*, x)$, but the convergence rate $R^* \neq \text{const} \quad \forall x \in \Omega$. This non-constant rate implies that there exists $x, \hat{x}$ such that $r(x) > r(\hat{x})$, where $r(x)$ is a local maximum. Since $h^*(x)$ is strictly increasing with r, it can redistribute sensors by removing some from $\hat{x}$ and placing them in the region x. This process reduces the local maximum $r(x)$, contradicting the assumption that $h^*(x)$ was optimal. Therefore, the only situation when all local maxima are minimized is $r = \text{const} \quad \forall x \in \Omega$.

Proving the equivalence claim helps to determine the distance function $h(x)$ from (3.7):

$$h(x) = \left(\frac{R^* - a(x)}{b} \right)^{3/4}, \tag{3.26}$$

where R^* is a minima solving (3.19). The distance function $h(x)$ described above does not directly provide sensor locations; instead, it defines the pairwise distances between sensors. To infer sensor locations from these distances, an algorithm is proposed in Section 3.5. In this thesis, the term 'target sensor locations' refers to sensor positions determined by the distance function $h(x)$, whereas 'directed sensors' are the physical sensors that move toward these target locations.

3.5 Target sensor locations

The thesis computes target sensor locations using perhaps the simplest algorithm (Algorithm 1). This algorithm relies on the distance function $h(x)$ (the reciprocal of which yields the sensor density function $1/h(x)$), where sensors are densely placed when $h(x)$ is small (i.e., the sensor density is high). This is achieved by dividing the discretized domain Ω into intervals until the number of intervals equals the number of available sensors, N. Once the number of intervals equals N, sensors are placed in the middle of each interval. Intuitively, (3.26) suggests that regions of high gradient (i.e, large $a(x)$) must receive more intervals and thus, be observed by more sensors.

To keep track of intervals at each iteration of the algorithm process and to decide which interval needs more refinement, a set of intervals is initialized as $\xi = \{\Omega\}$. Next, the condition $|\xi| < N$ is checked, and if it is false, the algorithm terminates. For instance, setting the total number of sensors to $N = 1$ would yield one sensor located at $(a + b)/2$, where $[a, b]$ are the bounds of the interval (or domain) Ω. Otherwise, the interval Ω is removed from the set and split in the middle, which populates the set with two intervals $\xi = \{\Omega_l, \Omega_r\}$. Then, the algorithm computes the number of sensors needed for each interval based on the integral of the sensor density (i.e., carrying out (3.11) for each interval). Finally, the algorithm selects the next interval with higher density from ξ and repeats the process. For visual demonstration of this algorithm, refer to Figure 3.1.

3.6 Directed sensors

Once target locations are obtained from the Algorithm 1, directed sensors can move towards these locations. This movement is typically constrained by a finite sensor speed to replicate physical scenarios. However, for simplicity, this thesis assumes that directed sensors move to target locations immediately. When the directed sensors reach their target locations, Algorithm 1 is executed again. This approach enables both target sensor locations and the directed sensors to track the key features of the flow as it evolves.

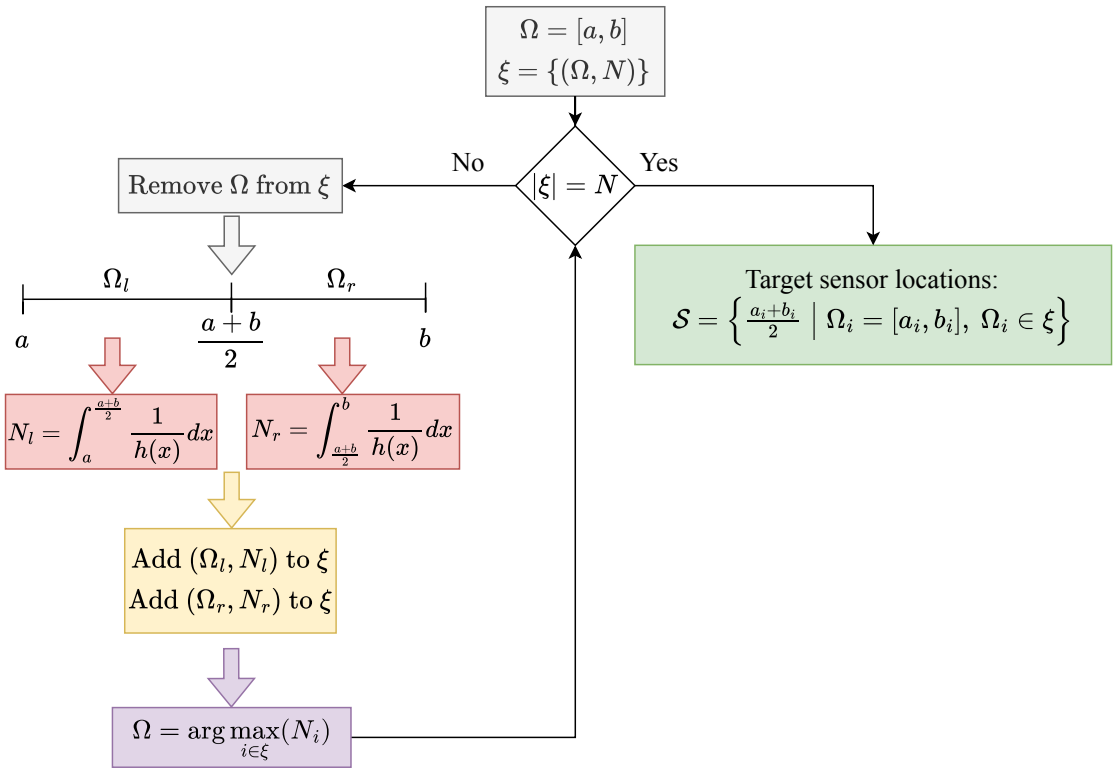


Figure 3.1: A flow chart of finding target sensor locations based on the Algorithm 1.

3.7 Lagrangian and the perturbed Lagrangian sensors

The passive movement trajectory of Lagrangian sensors is not necessarily optimal in all scenarios and could degrade the performance of the DA framework, including the AOT algorithm (Chapter 2). One intuitive counterexample for the optimality of the Lagrangian sensors is the 1D KSE (3.1), as shown in Figure 3.2. From this figure, distributed Lagrangian sensors would cluster at stagnation points. Over time, stagnation points vanish and emerge, eventually causing all Lagrangian sensors to cluster at a single stagnation point and move along with it. This clustering leads to redundant observations at one point and insufficient coverage in other regions of the domain.

To address the issue of the Lagrangian sensors, one hypothesizes that perturbations from the Lagrangian trajectory could prevent the clustering and improve the DA. The

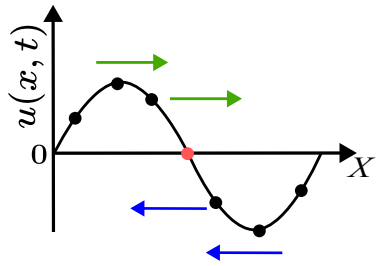


Figure 3.2: Clustering of Lagrangian sensors (black circles) in 1D KSE. Green arrows indicate regions where the flow moves to the right, while blue arrows indicate leftward motion. The red circle marks an example of a stagnation point.

simplest perturbations might be introducing random noise to the ideal Lagrangian sensors:

$$x(t) = u(x(t), t) + c \cdot \delta \cdot u^*, \tag{3.27}$$

where δ represents white noise drawn from a uniform distribution $\delta \sim U(-1, 1)$, c is a scalar coefficient that controls the magnitude of the perturbation, and u^* is the flow's characteristic velocity:

$$u^* = \sqrt{\frac{1}{LT} \int_0^T \int_0^L u(x,t)^2 \, dxdt}. \tag{3.28}$$

where T is the simulated time and L the length of the domain.

The proposed random perturbation may challenge the traditional view of the effectiveness of Lagrangian sensors in DA. If the perturbed sensors result in improved AOT reconstruction, it would suggest that the ideal Lagrangian configuration is far from optimal, as even a simple perturbation leads to better performance. Numerical tests comparing the performance of perturbed and ideal Lagrangian sensors will be presented in Chapter 5.

Chapter 4

Sensor placement and movement for 2D NSE

The current chapter builds upon the concepts of optimal sensor movement from the 1D KSE to the 2D NSE. First, Sections 4.1–4.2 briefly review the theoretical setup of the 2D NSE and the error convergence analysis of the corresponding AOT algorithm. Second, Section 4.3 transforms this convergence analysis into the minimization problem. Next, Section 4.4 provides a solution to this minimization problem, which guides sensor placement/movement to achieve fast convergence for the AOT-based reconstruction. Lastly, Section 4.5 discusses how a random perturbation applied to Lagrangian sensors might improve the performance of the AOT algorithm, and perhaps other DA methods.

4.1 Theoretical setup of Navier-Stokes equations

The Navier-Stokes equations (NSE) form the mathematical foundation for describing the motion of viscous fluid substances. These equations underpin many applications in fluid dynamics, including oceanography [58], biomedical flows [48, 107], and atmospheric modeling [118]. In addition to their practical applications, the NSE are often useful for studying the nature of turbulence and the distribution of energy across scales.

For instance, the 2D NSE exhibits an inverse energy cascade, in which energy transfers from fine-scale structures, such as small eddies, to larger scales. In contrast to 2D NSE, the 3D NSE display a forward cascade, where large eddies break down into smaller ones until the energy is dissipated. Despite this difference in energy flow, the long-term dynamics

in both 2D and 3D NSE are governed by the large scales [65], which are often the most relevant for a successful AOT algorithm [7].

Because of its simpler structure, lower computational cost, and the presence of large-scale dynamics that dominate long-term behavior, the thesis focuses on the 2D NSE as an illustrative example for studying optimal sensor placement/movement using the AOT algorithm [7]. In 2D, incompressible divergence-free NSE are written as

$$\begin{aligned} \mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u} - \nu \nabla^2 \mathbf{u} &= -\nabla p + \mathbf{f} \\ \nabla \cdot \mathbf{u} &= 0 \end{aligned}, \tag{4.1}$$

where $\mathbf{u}$ is the velocity in x and y directions, $\mathbf{f}$ the body force term, p the pressure, and ν the kinematic viscosity. Here, the viscosity term makes the NSE a dissipative system.

The NSE belong to the class of dissipative dynamical systems, which are known to exhibit asymptotic behavior toward a global attractor over time [33, 35, 49, 52, 59, 101, 108, 110, 121]. Importantly, this global attractor for the NSE is typically finite-dimensional [8, 34], implying that a finite number of degrees of freedom govern their long-term behavior. As a result, reconstructing the NSE using DA methods, including the AOT algorithm, becomes trivial when a sufficient number of these degrees of freedom are observed.

While observing a sufficient number of degrees of freedom ensures the NSE reconstruction under the AOT algorithm [6, 83], a limited number of sensors prevents this guarantee. With this limitation, one must strategically move sensors to approximate at least some degrees of freedom by observing information-rich regions of the NSE. This sensor movement is achieved by the error convergence analysis of the AOT algorithm, as discussed in Section 4.2.

4.2 Error convergence analysis for 2D NSE

The following error convergence analysis of the AOT algorithm is a recap of the results introduced in [83], which is extended to study optimal sensor placement and movement. First, note $\mathbf{u}$ is a classical solution to (4.1) assuming admissible initial conditions $\mathbf{u}_0 \in H^2 \cap H_0^1$ [83]. Then, applying the AOT algorithm to (4.1) yields

$$\begin{aligned} \hat{\mathbf{u}}_t + \hat{\mathbf{u}} \cdot \nabla \hat{\mathbf{u}} - \nu \nabla^2 \hat{\mathbf{u}} &= -\nabla p + \mathbf{f} + \kappa I_h(\mathbf{u} - \hat{\mathbf{u}}) \\ \nabla \cdot \hat{\mathbf{u}} &= 0 \end{aligned}, \tag{4.2}$$

where $\hat{\mathbf{u}}$ is also a classical solution to (4.2) given admissible initial conditions $\hat{\mathbf{u}}_0 \in H^2 \cap H_0^1$, κ is the nudging parameter and I_h is the linear interpolation with distance between sensors

h. Next, the NSE (4.1) and the AOT reconstruction (4.2) are simplified to eliminate the pressure term and the divergence-free condition. This elimination is done by mapping these PDEs into a divergence-free space. This map is called the Leray Projector, defined as

$$P_\sigma : L_0^2(\Omega) \to H,$$

where $L_0^2(\Omega)$ consists of square-integrable function with zero mean, and H is a Hilbert space.

Besides mapping the NSE into the divergence-free space, the Leray Projector is also useful for replacing the nonlinear terms in (4.1) and (4.2). This replacement is done by defining the Stokes operator

$$A\mathbf{u} = -P_\sigma(\Delta \mathbf{u})$$

and the bilinear form

$$B(\mathbf{u}, \hat{\mathbf{u}}) = P_\sigma((\mathbf{u} \cdot \nabla)\hat{\mathbf{u}}).$$

Finally, applying the Leray Projector to (4.1) and (4.2) yields

$$\begin{aligned} \text{NSE: } & \mathbf{u}_t + B(\mathbf{u}, \mathbf{u}) + \nu A(\mathbf{u}) = \mathbf{f} \\ \text{AOT: } & \hat{\mathbf{u}}_t + B(\hat{\mathbf{u}}, \hat{\mathbf{u}}) + \nu A(\hat{\mathbf{u}}) = \mathbf{f} + P_\sigma \kappa I_h(\mathbf{u} - \hat{\mathbf{u}}) \end{aligned}$$

with the assumption that the forcing term $\mathbf{f}$ is divergence-free, e.g. $P_\sigma(\mathbf{f}) = \mathbf{f}$ and non-divergence-free component has been absorbed into the pressure gradient.

After mapping (4.1) and (4.2) into the divergence-free space, the error convergence analysis of the AOT algorithm is briefly reviewed. Writing the error term as $\epsilon = \mathbf{u} - \hat{\mathbf{u}}$ and subtracting (4.2) from (4.1) yields the error equation:

$$\epsilon_t + B(\mathbf{u}, \mathbf{u}) - B(\hat{\mathbf{u}}, \hat{\mathbf{u}}) + \nu A\epsilon = -\kappa I_h \epsilon. \tag{4.3}$$

Note that the difference between bilinear terms can be written as

$$B(\mathbf{u}, \mathbf{u}) - B(\hat{\mathbf{u}}, \hat{\mathbf{u}}) = B(\mathbf{u}, \epsilon) + B(\epsilon, \mathbf{u}) - B(\epsilon, \epsilon),$$

transforming the error equation in (4.3) into

$$\epsilon_t + B(\mathbf{u}, \epsilon) + B(\epsilon, \mathbf{u}) - B(\epsilon, \epsilon) + \nu A\epsilon = -\kappa I_h \epsilon. \tag{4.4}$$

Next, taking the inner product in (4.4) with ϵ gives

$$\frac{1}{2}\frac{d}{dt}\|\epsilon\|_{L^2}^2 + \langle B(\mathbf{u}, \epsilon), \epsilon\rangle + \langle B(\epsilon, \mathbf{u}), \epsilon\rangle - \langle B(\epsilon, \epsilon), \epsilon\rangle + \nu \left\| A^{\frac{1}{2}}\epsilon \right\|_{L^2}^2 = -\kappa \langle I_h(\epsilon), \epsilon\rangle . \tag{4.5}$$

Then, using the orthogonality property of the bilinear form (Lemma 2.2 in [83]):

$$\langle B(\mathbf{u}, \hat{\mathbf{u}}), \hat{\mathbf{u}} \rangle = 0 \quad \forall \mathbf{u}, \hat{\mathbf{u}}, \epsilon \in H^1,$$

where H^1 is seminorm, two of the bilinear forms vanish, thus the error equation in (4.5) simplifies to

$$\frac{1}{2}\frac{d}{dt}\|\epsilon\|_{L^2}^2 + \langle B(\epsilon, \mathbf{u}), \epsilon \rangle + \nu \left\| A^{\frac{1}{2}} \epsilon \right\|_{L^2}^2 = -\kappa \langle I_h(\epsilon), \epsilon \rangle . \tag{4.6}$$

Taking the estimates for the nonlinear term as in [89] (i.e., having pointwise control over $\nabla \mathbf{u}$) and using periodic boundary conditions, the expression (4.6) results in

$$\begin{aligned}
\frac{1}{2}\frac{d}{dt}\|\epsilon\|_{L^2}^2 + \nu \|\epsilon\|_{H^1}^2 &= -\langle B(\epsilon, \mathbf{u}), \epsilon \rangle - \kappa \langle I_h(\epsilon), \epsilon \rangle \\
&\leq \|\nabla \mathbf{u}\|_{L^\infty}^2 \|\epsilon\|_{L^2}^2 + \kappa \|\epsilon - I_h(\epsilon)\|_{L^2} \|\epsilon\|_{L^2} - \kappa \|\epsilon\|_{L^2}^2 \\
&\leq \|\nabla \mathbf{u}\|_{L^\infty}^2 \|\epsilon\|_{L^2}^2 + \frac{\kappa}{2} \|\epsilon - I_h(\epsilon)\|_{L^2}^2 + \frac{\kappa}{2} \|\epsilon\|_{L^2}^2 - \kappa \|\epsilon\|_{L^2}^2 \\
&\leq \|\nabla \mathbf{u}\|_{L^\infty}^2 \|\epsilon\|_{L^2}^2 + \frac{\kappa}{2} c_0 h^2 \|\epsilon\|_{H^1}^2 + \frac{\kappa}{2} \|\epsilon\|_{L^2}^2 - \kappa \|\epsilon\|_{L^2}^2 \\
&\leq \left(\frac{\kappa}{2} c_0 h^2\right) \|\epsilon\|_{H^1}^2 + \left(\|\nabla \mathbf{u}\|_{L^\infty}^2 - \frac{\kappa}{2}\right) \|\epsilon\|_{L^2}^2 ,
\end{aligned}$$

Grouping terms with similar norms yields an inequality in terms of the reconstruction error

$$\frac{1}{2}\frac{d}{dt}\|\epsilon\|_{L^2}^2 \leq \left(\frac{\kappa}{2} c_0 h^2 - \nu\right) \|\epsilon\|_{H^1}^2 + \left(\|\nabla \mathbf{u}\|_{L^\infty}^2 - \frac{\kappa}{2}\right) \|\epsilon\|_{L^2}^2 . \tag{4.7}$$

Using the definition of H^1 norm, the above expression results in

$$\frac{1}{2}\frac{d}{dt}\|\epsilon\|_{L^2}^2 \leq \left(\frac{\kappa}{2} c_0 h^2 - \nu\right) (\|\epsilon\|_{L^2}^2 + \|\nabla \epsilon\|_{L^2}^2) + \left(\|\nabla \mathbf{u}\|_{L^\infty}^2 - \frac{\kappa}{2}\right) \|\epsilon\|_{L^2}^2 .$$

Lastly, matching terms with the same error norms in the above gives

$$\frac{1}{2}\frac{d}{dt}\|\epsilon\|_{L^2}^2 \leq \underbrace{\left(\frac{\kappa}{2} c_0 h^2 - \nu + \|\nabla \mathbf{u}\|_{L^\infty}^2 - \frac{\kappa}{2}\right)}_{r(h,x,y)} \|\epsilon\|_{L^2}^2 + \underbrace{\left(\frac{\kappa}{2} c_0 h^2 - \nu\right)}_{M(h,x,y)} \|\nabla \epsilon\|_{L^2}^2 . \tag{4.8}$$

The upper bound of (4.8) resembles the classical first-order ODE for the error $\|\epsilon\|_{L^2}^2$. This ODE suggests that an exponential error convergence to zero is achieved when two coefficients on the right-hand-side are negative:

$$r(h, x, y) = \left(\underbrace{\frac{\kappa}{2} c_0 h^2}_{bh^2} \underbrace{-\nu + \|\nabla \mathbf{u}\|_{L^\infty}^2 - \frac{\kappa}{2}}_{a(x,y)} \right) < 0, \tag{4.9}$$

where $r(h, x, y)$ can be viewed as a global convergence rate of the error, and

$$M(h, x, y) = \frac{\kappa}{2} c_0 h^2 - \nu < 0,$$

where $M(h, x, y)$ denotes the constant to which the error converges. Having $M(h, x, y) < 0$ guarantees that the error decays to zero. However, even if $M(h, x, y)$ is slightly positive, convergence to zero might still occur, since the derived analysis is not sharp. Therefore, the focus shifts toward achieving a faster convergence rate.

Similar to the argument in Chapter 3, the smaller $r(h, x, y)$ is, the faster the AOT algorithm would converge. Therefore, to get the fastest error convergence, the error convergence rate $r(h, x, y)$ must be minimized. This leads to the minimization problem for the error convergence rate $r(h, x, y)$, which guides sensor placement/movement discussed in Section 4.3.

4.3 Minimization problem

As in the analysis presented in Chapter 3, the convergence rate given in (4.9) allows a pointwise interpretation. This enables us to work with local convergence rates, where the global gradient norm is replaced with a local magnitude $\|\nabla \mathbf{u}\|_{L^\infty}^2 = \max(|\nabla u(x, y)|^2, |\nabla v(x, y)|^2)$. The sensor distance h is similarly treated as a spatially varying function $h(x, y)$, which can guide sensor placement/movement. These two factors, the magnitude of the local gradient and the sensor distance, govern the local convergence rate and, in turn, influence the global convergence behavior.

While each local convergence rate contributes to the global convergence behavior, the latter is ultimately dictated by the maximum of the local rates (similar to the 1D KSE case as shown in Chapter 3). This is because even small errors in the region with a high and positive local convergence rate $r(h(x, y), x, y)$ (i.e., sensitive region) may grow exponentially, dominating and degrading the global convergence. Therefore, the minimization problem is formulated as finding such a distance function $h(x, y)$ to observe this sensitive region:

$$\min_h \max_{x,y} a(x, y) + bh(x, y)^2. \tag{4.10}$$

Accounting for the distance function constraint on the number of available sensors N, the

minimization problem in (4.10) transforms into a constrained minimization:

$$\begin{aligned} &\min_{h} \max_{x,y} a(x,y) + bh(x,y)^2 \\ &\text{s.t.} \quad \left\{ \int_{\Omega_x} \int_{\Omega_y} \frac{1}{h(x,y)} dydx = N \right\}. \end{aligned} \tag{4.11}$$

Using the same chain of logic as in Chapter 3, the above minimization problem is replaced with its equivalent:

$$\begin{aligned} &\min_{h} r(h(x,y),x,y) \\ &\text{s.t.} \quad \left\{ \int_{\Omega_x} \int_{\Omega_y} \frac{1}{h(x,y)} dydx = N, r(h(x,y),x,y) = \text{const } \forall x,y \in \Omega \right\} \end{aligned} \tag{4.12}$$

Proving that the above minimization problem is equivalent to (4.11) requires applying KKT conditions. This proof is done in discrete settings, similar to Chapter 3, but it can be generalized to infinite-dimensional convex optimization over Hilbert space [10, 20].

4.4 Optimal $h(x,y)$ and the KKT conditions

The proof based on the KKT conditions closely follows the one presented in Chapter 3, so minor details are omitted here for brevity. First, transforming the minimization problem (4.10) into a convex and smooth one yields

$$\begin{aligned} &\min_{R,p} R, \\ &\text{s.t.} \quad \left\{ a_{ij} + bp_{ij}^{-2} \leq R, \sum_i \sum_j p_{ij} = N \right\}. \end{aligned} \tag{4.13}$$

Once a convex and smooth minimization problem is formulated, the KKT conditions are applied to prove the equivalence (4.12). First, discretizing the minimization problem and applying the KKT conditions yields the Lagrangian cost function:

$$\mathcal{L}(p,\lambda,\mu) = R + \sum_i \sum_j \lambda_{ij}(a_{ij} + bp_{ij}^{-2} - R) + \mu \sum_i \sum_j p_{ij}. \tag{4.14}$$

Next, finding the optimal p requires solving the system of equations formed by $\nabla_{R,p_i}\mathcal{L} = 0$:

$$\begin{pmatrix} \frac{\partial \mathcal{L}}{\partial R} \\ \frac{\partial \mathcal{L}}{\partial p_{11}} \\ \vdots \\ \frac{\partial \mathcal{L}}{\partial p_{nm}} \end{pmatrix} = 0 \Rightarrow \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} \lambda_{11} + \lambda_{12} + \ldots + \lambda_{nm} \\ 2\lambda_{11} b p_{11} \\ \vdots \\ 2\lambda_{nm} b p_{nm} \end{pmatrix} + \begin{pmatrix} 0 \\ \mu \\ \vdots \\ \mu \end{pmatrix}. \tag{4.15}$$

Note that if $\mu = 0$, then from the second to the last equation $\lambda_{nm} = 0$, contradicting the first component of the gradient equation. Therefore, $\mu \neq 0$, also implying that $\lambda_{nm} \neq 0$, since $p > 0$. This observation simplifies the complementarity condition

$$\lambda_{ij}(R - a_{ij} - b p_{ij}^{-2}) = 0,$$

yielding

$$\begin{aligned} \lambda_{ij}(R - a_{ij} - b p_{ij}^{-2}) &= 0 \\ (R - a_{ij} - b p_{ij}^{-2}) &= 0 \\ R &= a_{ij} + b p_{ij}^{-2}, \end{aligned} \tag{4.16}$$

which proves the equivalence claim (4.12).

After proving the equivalence claim, the distance function $h(x, y)$ can be easily found from (4.9):

$$\begin{aligned} \frac{\kappa c_0 h^2}{2} - \nu + \nabla u_{\max}^2 - \frac{\kappa}{2} &= R^* \\ h(x, y) &= \sqrt{\frac{2R^* + \kappa - 2\nabla u_{\max}^2 + 2\nu}{\kappa c_0}}, \end{aligned} \tag{4.17}$$

where R^* is a minima solving (4.13) and $\nabla u_{\max} = \max(|\nabla u(x, y)|^2, |\nabla v(x, y)|^2)$. The distance function defined above does not explicitly indicate locations where sensors should be placed. Therefore, similarly to Chapter 3, determining sensor locations is achieved through Algorithm 1, extending from a 1D to a 2D scenario.

4.5 Lagrangian and the perturbed Lagrangian sensors

While Lagrangian sensors are generally less prone to clustering in 2D turbulent flows than that observed in the 1D KSE (Chapter 3), they still move passively with the flow, which may

not be optimal for DA. For example, Lagrangian sensors initialized near attracting LCS, which are typically associated with high-strain regions [100], may remain trapped in these areas for extended periods. This tendency to persist in high-strain regions could reduce the effectiveness of DA by causing oversampling in those regions while undersampling vortical (i.e., low-strain, high-vorticity) areas.

To help Lagrangian sensors escape from high-strain regions, one might hypothesize that introducing a perturbation to their trajectories could prevent them from clustering (Figure 4.1).

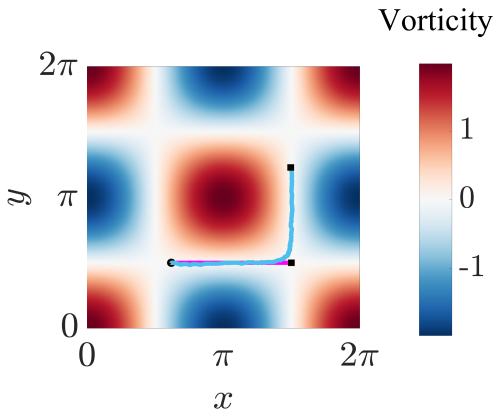


Figure 4.1: Lagrangian (purple) and perturbed Lagrangian (blue, $c = 0.5$) sensor trajectories. The black circle and black square indicate the start and end locations, respectively.

A simple form of such a perturbation is to add random noise to the ideal Lagrangian sensors:

$$\begin{aligned} \frac{dx}{dt} &= u(x(t), y(t), t) + c \cdot \alpha \cdot u^* \\ \frac{dy}{dt} &= v(x(t), y(t), t) + c \cdot \beta \cdot v^*, \end{aligned} \tag{4.18}$$

where u, v are velocity components in x and y directions respectively, α and β are noises drawn from uniform distribution $\alpha, \beta \sim U(-1, 1)$, c is the noise magnitude parameter and u^*, v^* are the flow's characteristic velocities in both directions:

$$u^* = \sqrt{\frac{1}{L_x L_y T} \int_0^T \int_{-L_y/2}^{L_y/2} \int_{-L_x/2}^{L_x/2} u(x, y, t)^2 \, dx \, dy \, dt} \tag{4.19}$$

$$v^* = \sqrt{\frac{1}{L_x L_y T} \int_0^T \int_{-L_y/2}^{L_y/2} \int_{-L_x/2}^{L_x/2} v(x, y, t)^2 \, dx \, dy \, dt}, \tag{4.20}$$

where L_x, L_y are lengths of domains Ω_x and Ω_y, respectively.

Similar to Chapter 3, the perturbation introduced above could challenge the performance of ideal Lagrangian sensors in reconstructing dynamical systems. If the perturbed Lagrangian sensors lead to improved AOT reconstruction, this would suggest that the ideal Lagrangian sensors may not be optimal for DA for 2D NSE, as even a simple perturbation leads to better performance. A detailed comparison between perturbed and ideal Lagrangian sensors will be presented in Chapter 5.

Chapter 5

Results and Discussion

5.1 1D Kuramoto–Sivashinsky equation

5.1.1 Numerical setup

The numerical experiments for validating sensor movement strategies were obtained by simulating the 1D KSE (3.1) using a pseudo-spectral code, employing a dealiasing rule of 2/3 [98]. The number of mesh nodes to simulate the KSE was chosen as $N_x = 1024$ to resolve the energy spectrum up to machine precision before the dealiasing cutoff (2/3 of N_x). This cutoff prevents aliasing by eliminating higher-frequency modes that would otherwise be folded back into the solution, introducing numerical errors. A timestep was used as $\Delta t = 0.01$ and the initial condition was set as $u_0 = \cos\left(\frac{x}{c}\right) \cdot \left(1 + \sin\left(\frac{x}{c}\right)\right)$, where $c = 16$. For sufficiently chaotic dynamics, the domain length and the viscosity were selected as $L = [0, 32\pi]$ and $\nu = 1$ accordingly. The nudging parameter was set to $\kappa = 100$. Lastly, the KSE characteristic velocity was calculated from (3.28) with $T = 1000$, resulting in $u^* = 1.3$.

A piecewise linear interpolation was used for the interpolation operator I_h defined in (2.5). To maintain periodicity during interpolation, the velocity observations and the original domain were extended by two endpoints $[x_{N_x} - L, x_1 + L]$, then interpolated over this extended domain and truncated back to the original domain. Lastly, the error between the AOT reconstruction and the KSE true solution was calculated using the Frobenius norm $\|\hat{u} - u\|_F$.

Once the experimental setup is complete, the focus shifts to validating the sensor movement strategies for AOT reconstruction of the KSE. The following subsections (Subsec-

tions 5.1.2–5.1.5) will describe the results of the numerical experiments on Lagrangian, inertia, perturbed Lagrangian, perturbed inertia, and directed sensors with $N = 1024$, $N = 103$, $N = 52$, and $N = 33$.

5.1.2 Lagrangian and inertia sensors

Lagrangian and inertia sensors with the low Stokes number[1] failed to achieve a successful AOT reconstruction of the KSE in both dense and sparse sensor scenarios. This failure in reconstruction happens due to clustering near stagnation points (Figure 5.1 (a), see also Figure 5.2 for Lagrangian trajectories that merge over time)

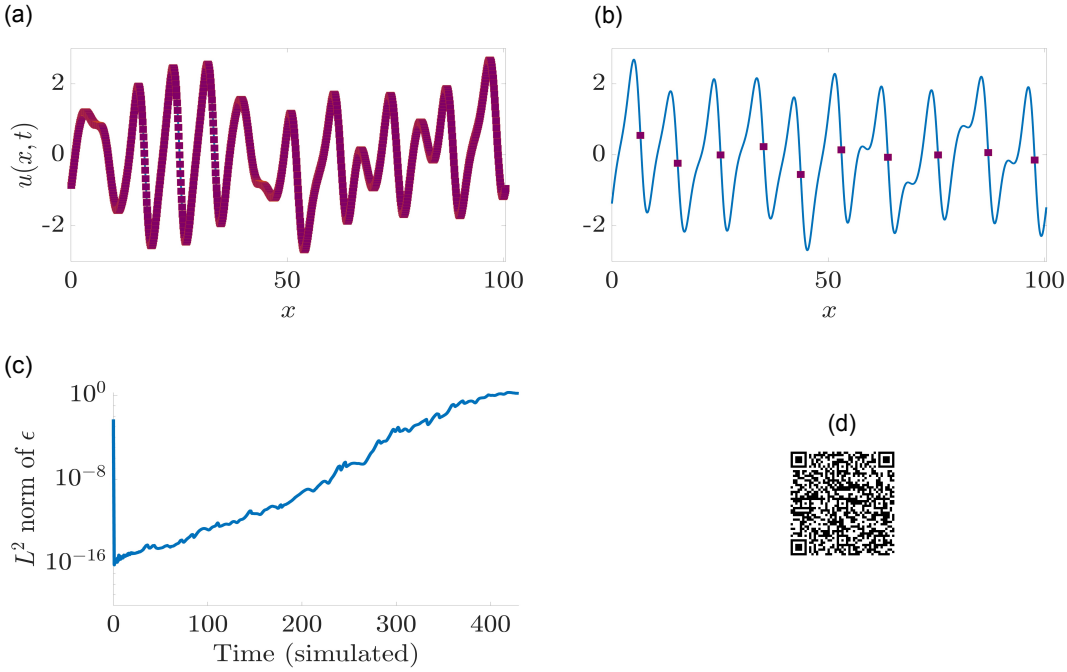


Figure 5.1: Error convergence with $N = 1024$ of uniformly initialized Lagrangian sensors. Panels (a) and (b) show the location of Lagrangian sensors at $t = 0$ and at $t = 20$, respectively, after clustering near stagnation points. Panel (c) illustrates the error behavior over time. For a visual demonstration of Lagrangian clustering in the steady KSE flow, access the QR code in (d).

[1]The terms "inertia sensors with low Stokes number" and "light sensors" are used interchangeably, as are "inertia sensors with high Stokes number" and "heavy sensors".

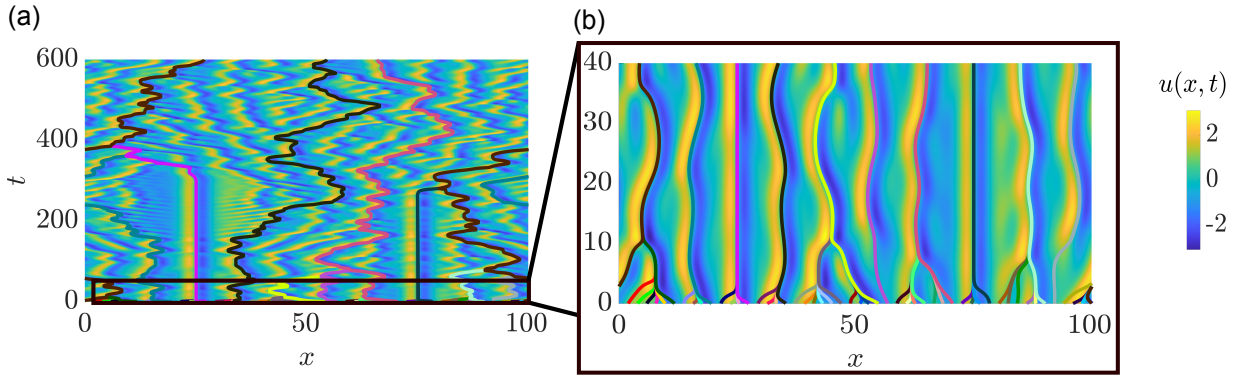


Figure 5.2: The flow field of the 1D KSE overlaid with the trajectories of $N = 52$ uniformly initialized Lagrangian sensors (a) and zoomed-in view (b).

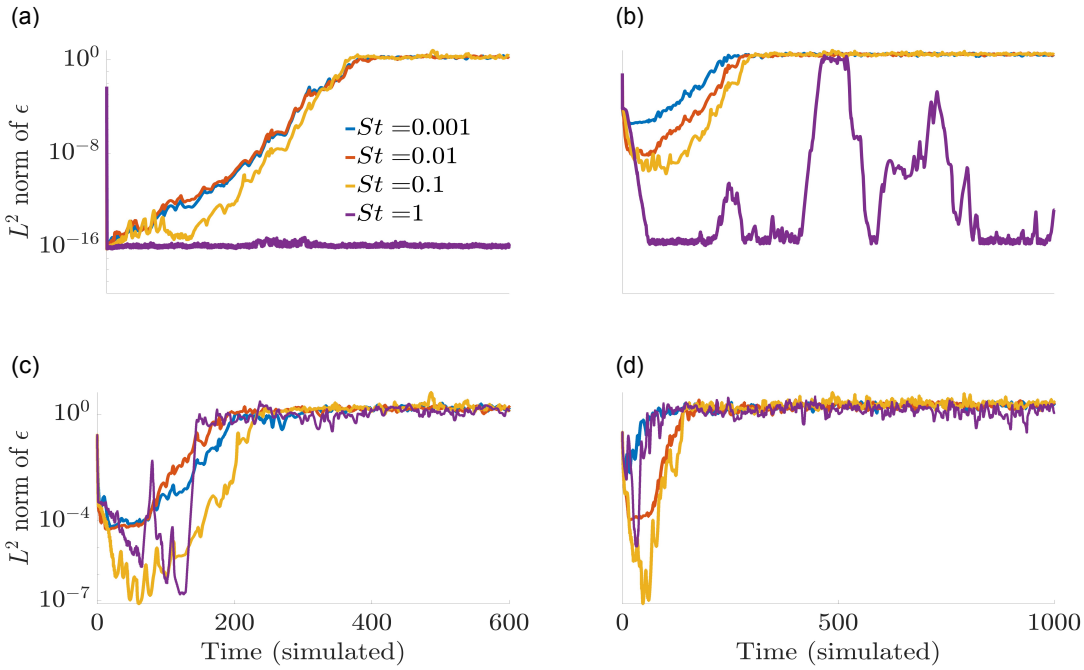


Figure 5.3: Error convergence with uniformly initialized inertia sensors. Panels (a–d) correspond to $N = 1024$, $N = 103$, $N = 52$, and $N = 33$, respectively. Lines with different colors are associated with various Stokes numbers, as shown in the legend in (a).

Observing the KSE flow with more than one sensor only at the stagnation points results in redundant measurements and fails to capture critical flow features present outside these

points. This implies that Lagrangian and light sensors are not necessarily optimal for a successful AOT reconstruction, and perhaps other DA methods.

While Lagrangian and light sensors caused the error divergence due to their clustering, heavy sensors (i.e., $St = 1$) do not exhibit similar clustering behavior; hence, the reconstruction of the KSE could be improved. The heavier the inertia sensors, the more time it takes for them to cluster near stagnation points (Figure 5.1 (d)). However, this clustering may never occur if stagnation points disappear before heavy sensors can reach them. Therefore, these sensors may achieve convergence of the error to machine precision with a large number of sensors (i.e., $N = 1024$, Figure 5.3 (a)).

Although heavy sensors achieved a successful AOT reconstruction with a large number of sensors, they failed to yield the same level of convergence with sparse sensors. For instance, the convergence with $N = 103$ is not monotonic (Figure 5.3 (b)), because clustering occurs for most of the available sensors. This may not provide the AOT algorithm with sufficient flow information, degrading the reconstruction of the true KSE solution (Figure 5.4). Furthermore, with $N = 52$ and $N = 33$, there were not enough sensors outside the clusters to measure the flow; thus, the error diverged (Figure 5.3 (c–d)).

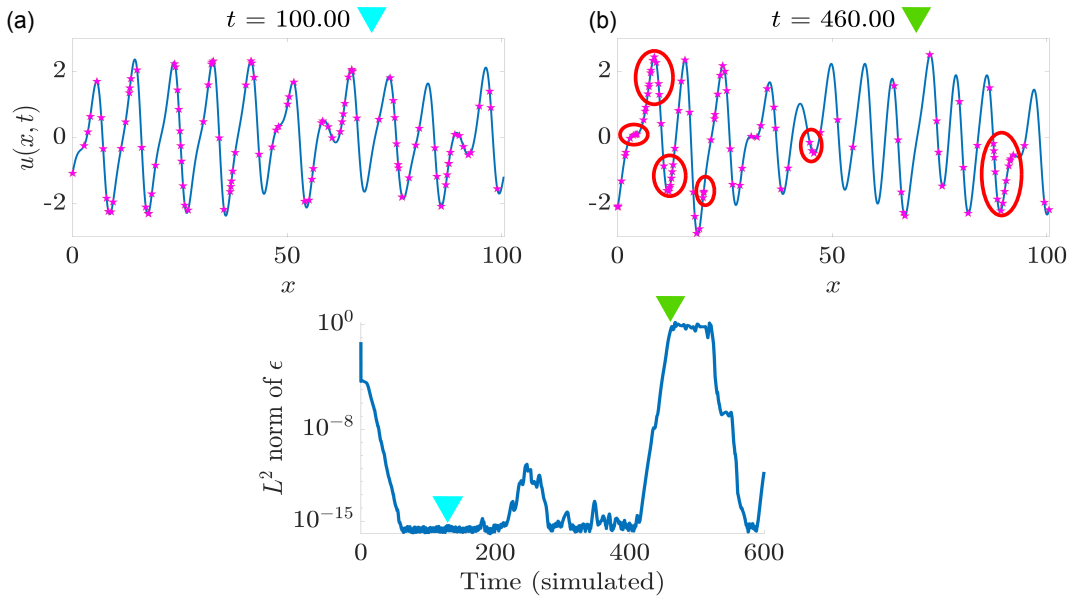


Figure 5.4: Error convergence with $N = 103$ of uniformly initialized inertia sensors with $St = 1$. Panels (a) and (b) show the locations of the inertia sensors at different times, denoted by colored triangles. The panel in the lower row illustrates the error behavior over time.

5.1.3 Perturbed Lagrangian and perturbed inertia sensors

To prevent clustering for Lagrangian and heavy sensors, a random perturbation is introduced in (3.27). Given the stochastic nature of the perturbation, fifty numerical experiments were conducted to assess the average convergence behavior of the error. These numerical experiments demonstrated that perturbation with a large amplitude prevents clustering, thus improving the AOT reconstruction for both Lagrangian sensors (see Figure 5.5 with the trajectories of perturbed Lagrangian sensors in Figure 5.7) and heavy sensors (Figure 5.8). Specifically, a perturbation amplitude of $c = 25$ was the smallest value that ensured error convergence for dense sensors (Figure 5.5 (a)), but it was not sufficient for sparse sensors (Figure 5.5 (b–d)). For the latter, a larger amplitude of $c = 40$ was required to achieve similar error convergence, although the convergence rate varied depending on N.

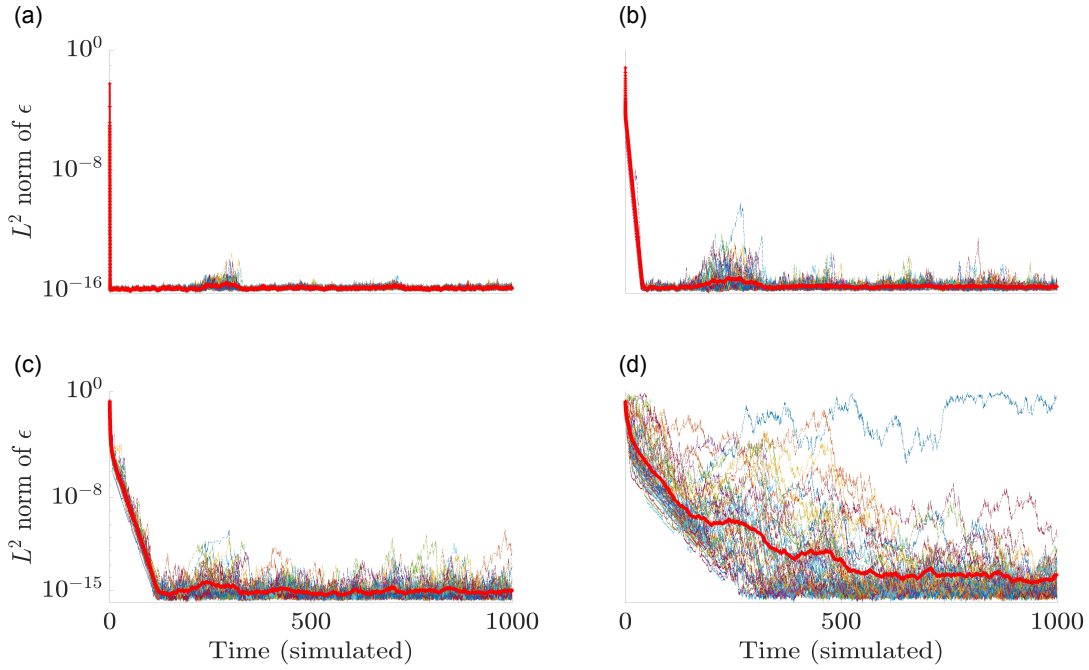


Figure 5.5: Average error convergence (red) with uniformly initialized perturbed Lagrangian sensors over fifty numerical experiments. Panels (a–d) correspond to ($N = 1024, c = 25$), ($N = 103, c = 40$), ($N = 52, c = 40$), and ($N = 33, c = 40$), respectively.

The above numerical experiments required high perturbations, which are not necessar-

ily practical; however, they demonstrated that ideal Lagrangian and inertia sensors are not necessarily optimal for an effective AOT algorithm, and a simple random perturbation of Lagrangian trajectories significantly improved the AOT reconstruction. Therefore, designing a better sensor movement strategy than random perturbation could yield even better performance.

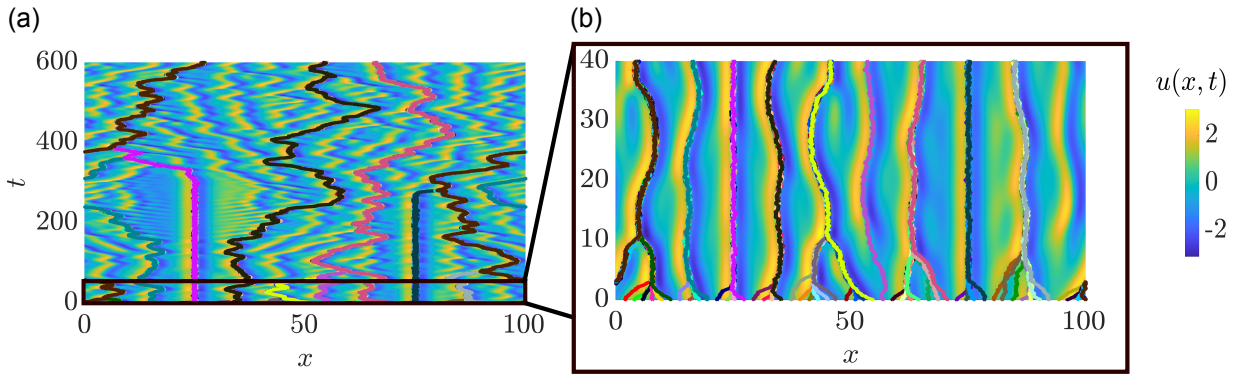


Figure 5.6: The flow field of the 1D KSE overlaid with the trajectories of $N = 52$ uniformly initialized perturbed Lagrangian sensors with $c = 5$ (a) and zoomed-in view (b).

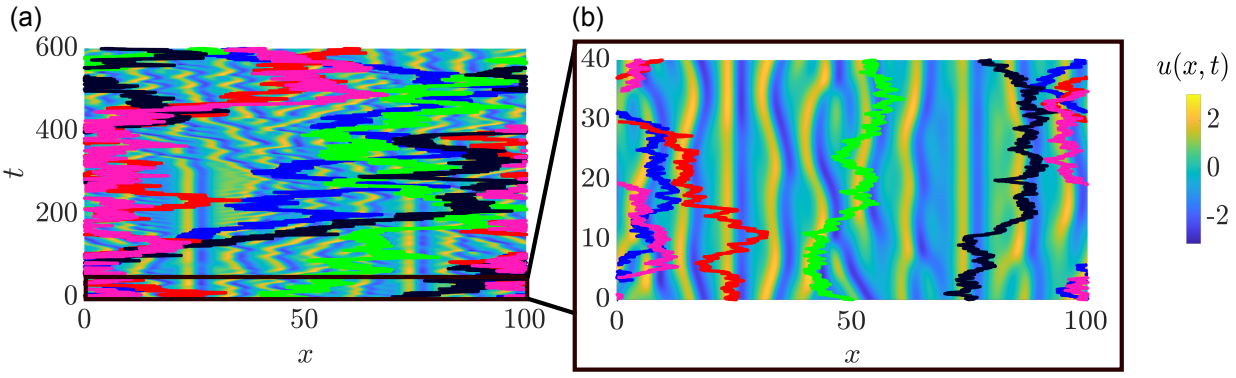


Figure 5.7: The flow field of the 1D KSE overlaid with the trajectories of $N = 5$ uniformly initialized perturbed Lagrangian sensors with $c = 40$ (a) and zoomed-in view (b).

5.1.4 Directed sensors

Directed sensors move immediately toward target locations determined by the algorithm introduced in Chapter 3. Numerical experiments demonstrated that directed sensors

achieved convergence of the error to machine precision in both dense and sparse sensor scenarios (Figure 5.9).

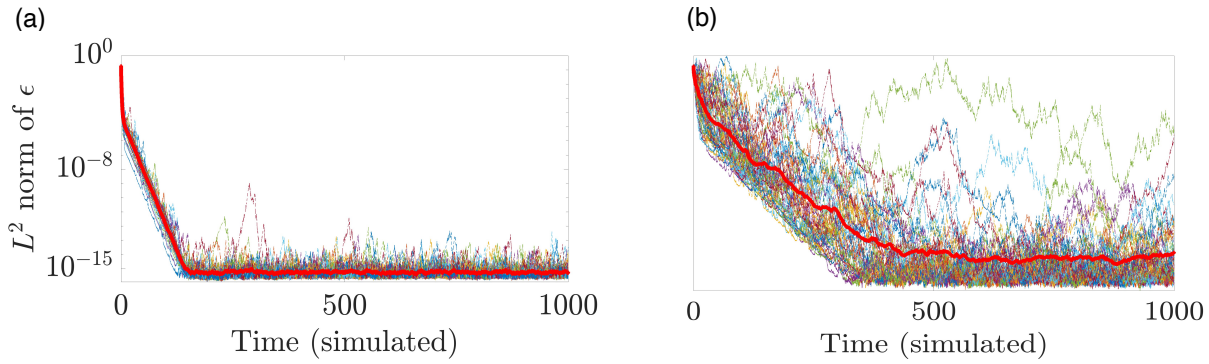


Figure 5.8: Average error convergence (red) with uniformly initialized perturbed inertia sensors ($St = 1$) over fifty numerical experiments. Panels (a–b) correspond to $N = 52$, and $N = 33$, respectively, with $c = 40$.

Additionally, the actual convergence rate was compared with the designed rate, R^*, established in (3.7), showing that the former is always higher than the latter, as the derived analysis is not sharp (Figure 5.10). This implies that, in practice, the actual error may converge even when the designed one diverges (Figure 5.10 (b)).

The performance of the AOT algorithm in the above experiments also depends on the speed of the directed sensors. Although they move to target locations immediately, their average speed was calculated to verify if these sensors are practical. This speed was calculated as

$$\bar{v}_p = \frac{1}{T}\frac{1}{N}\sum_{t_i=1}^{T}\sum_{p=1}^{N}|v_p|, \tag{5.1}$$

where $v_p = (|x_t - x_{t-1}|)/\Delta t$ represents the change in an individual sensor's position between times $t-1$ and t. For instance, plugging $T = 1000$ and $N = 33$ in the above equation yielded $\bar{v}_p \approx 0.66$, which is almost two times lower than the characteristic velocity $u^* = 1.3$. This result suggests that directed sensors exhibit minimal movement, as the distances to their dynamically changing target locations are typically small (Figure 5.11). Besides "infinite" speed, we tested the case with sensor speed limited to $v_p = u^*$, which also yielded error convergence for both dense and sparse sensors. Therefore, directed sensors are feasible from a practical perspective.

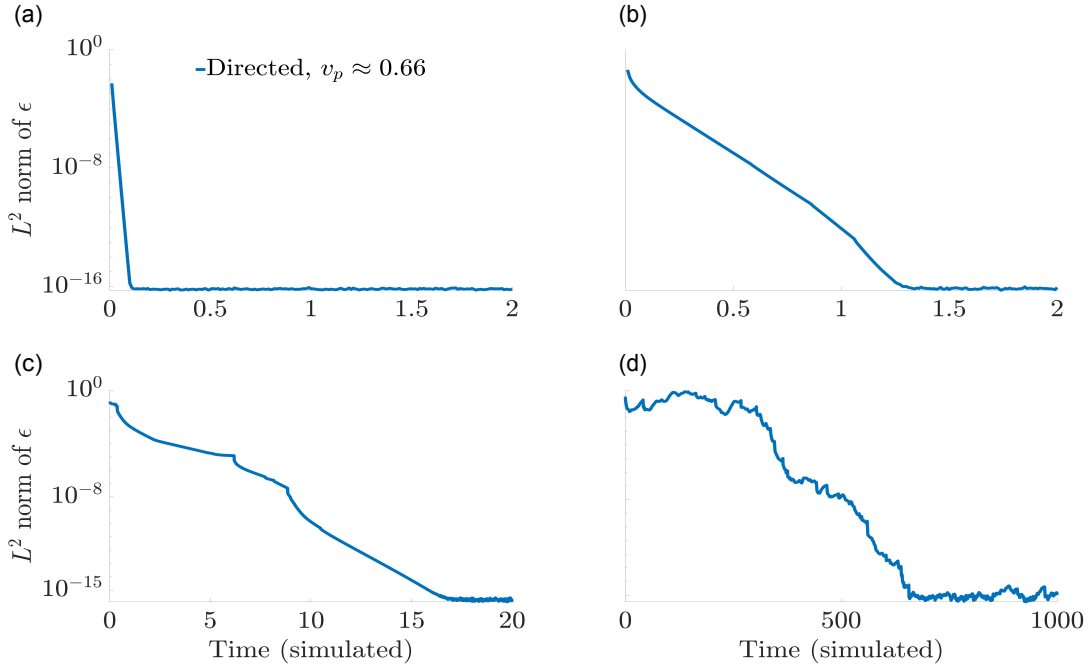


Figure 5.9: Error convergence of directed sensors (red, $v_p \approx 0.66$) with $N = 1024$ in (a), $N = 103$ in (b), $N = 52$ in (c), and $N = 33$ in (d). In both dense and sparse sensor scenarios, directed sensors achieved convergence of error to machine precision.

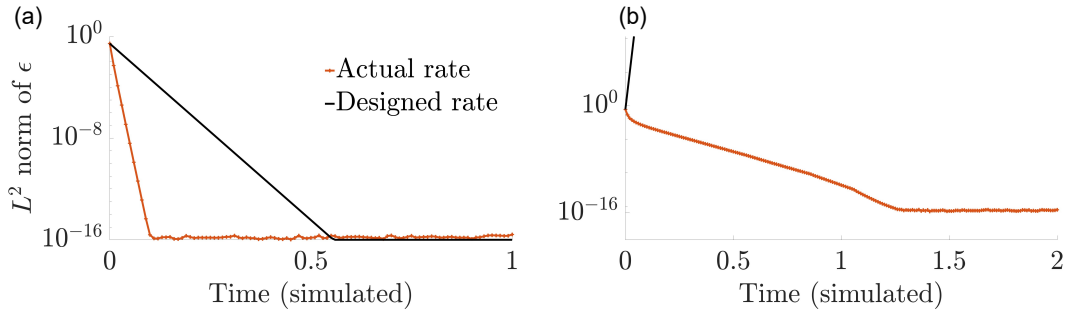


Figure 5.10: Error convergence rate comparison between the directed sensors (red, $v_p \approx 0.66$) and the designed rate (black) in (3.7) with $N = 1024$ in (a), and $N = 103$ in (b).

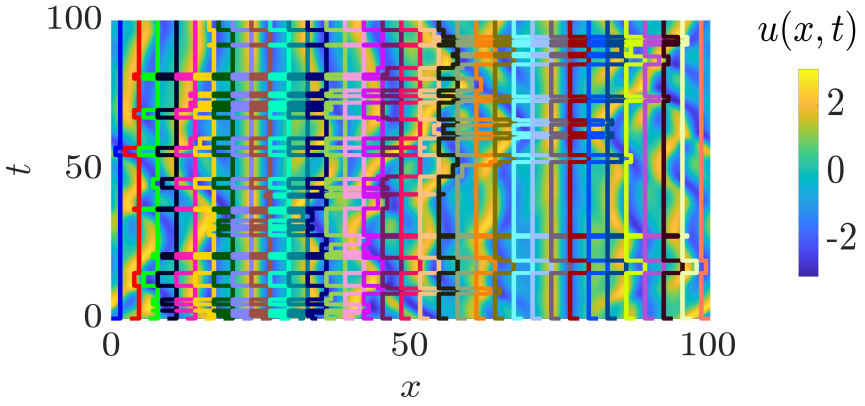


Figure 5.11: The flow field of the 1D KSE overlaid with the trajectories of $N = 33$ directed sensors.

5.1.5 Error convergence of sensor movement strategies

After reviewing each sensor movement strategy separately, this subsection will compare them in the same plot (Figure 5.12). First, using both Lagrangian and heavy sensors led to divergence of the AOT reconstruction from the true KSE solution due to clustering. To mitigate this clustering, a random perturbation with a large noise amplitude was applied, which improved the convergence of the AOT algorithm. These results suggest that ideal Lagrangian and inertia sensors are not necessarily optimal, as even a simple random perturbation achieved error convergence. Second, the optimal directed sensors achieved error convergence to machine precision in both dense and sparse sensor scenarios. Moreover, these sensors achieved the fastest convergence rate at $N = 1024$ among other sensor movement strategies (Figure 5.10 (a)). Lastly, in our tests, limiting the speed of these sensors to $v_p = u^*$ also achieved a successful AOT convergence (not shown). This demonstrates that moving sensors to locations that maximize the convergence rate of the AOT algorithm can lead to faster and more effective reconstruction.

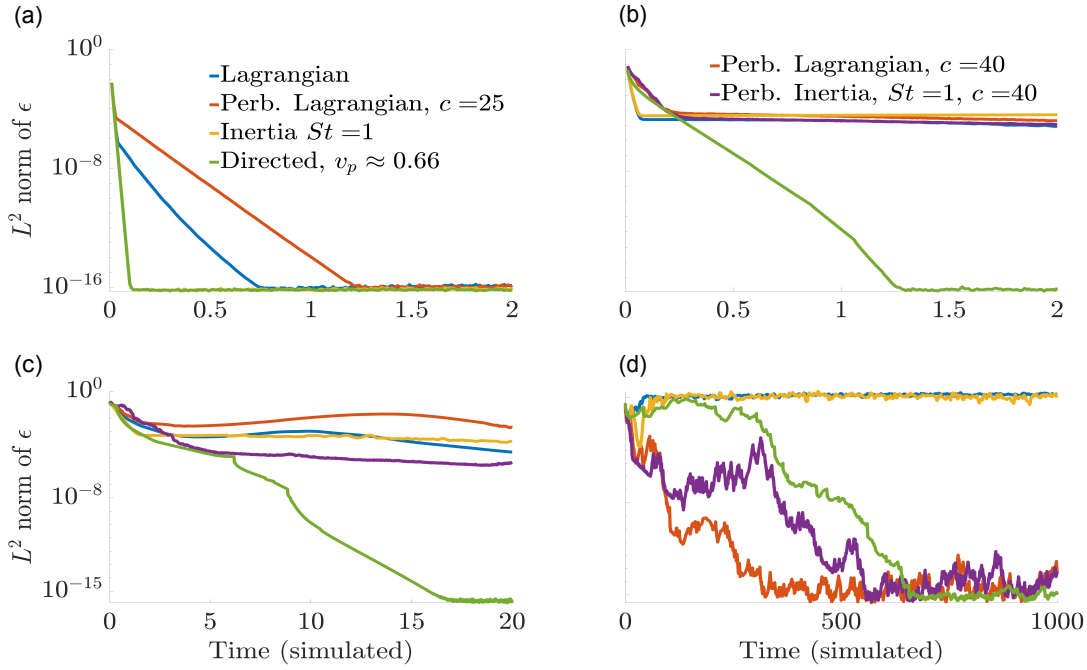


Figure 5.12: Initial stage of the error convergence of sensor movement strategies with $N = 1024$ in (a), $N = 103$ in (b), $N = 52$ in (c), and later stage with $N = 33$ in (d). Lines with different colors are associated with various sensor movement strategies, as shown in the legend in (a–b).

5.2 2D Navier-Stokes equations

5.2.1 Numerical Setup

The numerical results for testing sensor movement strategies were obtained from solving a vorticity-stream formulation of the 2D NSE in (4.1) on a periodic domain $\Omega = [-\pi, \pi]^2$ using a dealiased pseudo-spectral code. Similar to [18, 54], the viscosity was set $\nu = 10^{-4}$ to obtain a forcing $\mathbf{f}$ in Fourier space with the Grashof number

$$G = \frac{\|\mathbf{f}\|_{L^2}}{\nu^2 \lambda_1} = 10^6,$$

where λ_1 is the first eigenvalue of the Laplace operator. The Reynolds number is

$$Re = \frac{u^* L}{\nu} \approx 6000,$$

where L is the characteristic length scale of the domain. The spatial resolution was set to $N_x = N_y = 1024$ to resolve the energy spectrum for initial conditions to machine precision before the dealiasing cutoff (2/3 of N_x). The nudging parameter was set to $\kappa = 10$. The initial conditions were obtained by evolving zero initial data until $T = 25000$. Then, we used this initial data to simulate numerical experiments for a duration of $T = 1000$. Finally, the characteristic velocity was calculated from (4.19) as $u^* \approx 0.096$.

A piecewise linear interpolation I_h from (4.2) in the physical space was used to make the interpolation process more computationally feasible. To ensure a proper interpolation in the periodic domain, velocity observations were extended to $[-3\pi, 3\pi]^2$, then interpolated over this extended domain, and lastly truncated back to the original domain $\Omega = [-\pi, \pi]^2$.

Once the experiment setup is done, the following subsections (Subsections 5.2.2–5.2.5) will review the performance of the AOT algorithm using sensor movement strategies, such as Lagrangian and inertia (both perturbed and unperturbed), as well as directed sensors with $N = 2500$, $N = 900$, $N = 576$, and $N = 196$.

5.2.2 Lagrangian and inertia sensors

Using Lagrangian and light sensors achieved error convergence to machine precision with $N = 2500$, $N = 900$, and $N = 576$ (Figure 5.13 (a–c)), though the error diverged with $N = 196$ (Figure 5.13 (d)). This error convergence occurs partially because these sensors typically do not cluster in 2D turbulent flows. Therefore, when deployed in large numbers, Lagrangian and light sensors may provide sufficient flow information for an effective AOT algorithm. However, having sparse sensors (e.g., $N = 196$) may fail to capture the NSE flow with adequate spatial resolution, causing the error of the AOT reconstruction to blow up.

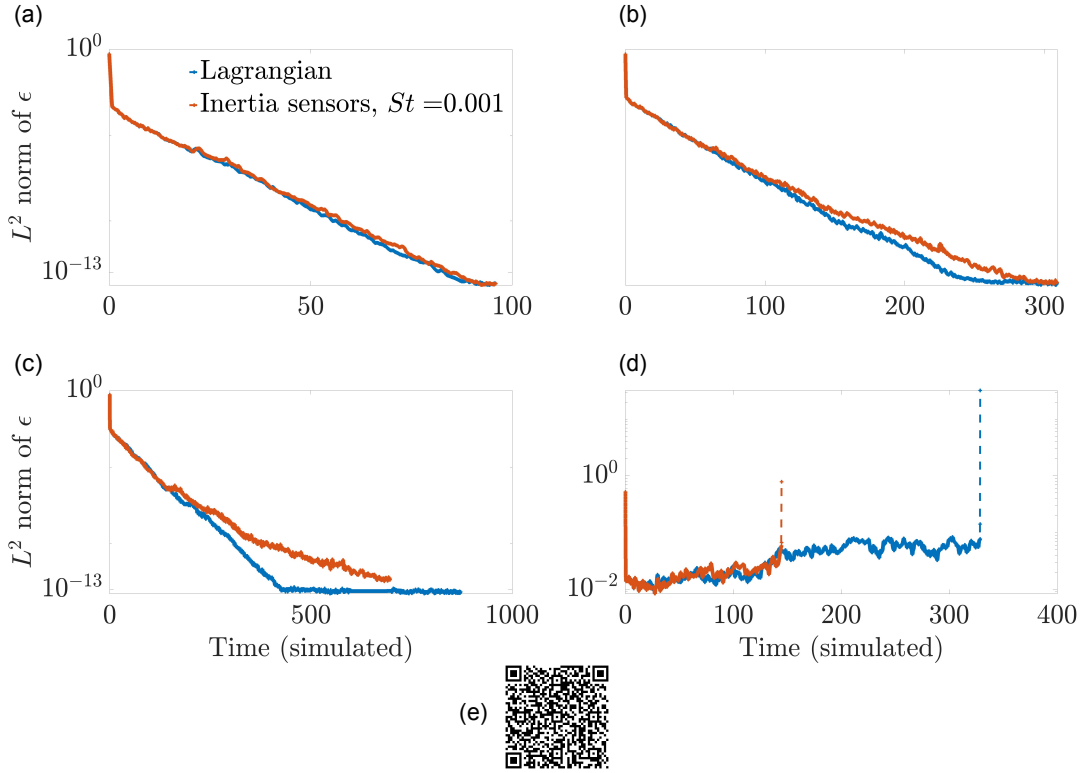


Figure 5.13: Error convergence of Lagrangian (blue) and inertia sensors (red) with $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), and $N = 196$ in (d). To see the motion of Lagrangian sensors in turbulent flow, access the QR code in (e).

While having a large number of Lagrangian and light sensors achieved a successful AOT reconstruction, using inertia sensors with higher St demonstrated limited performance. For instance, inertia sensors with $St = 0.01$ and $St = 0.1$ led to diverged DA, even for dense sensors (Figure 5.14). This divergence is caused by sensor clustering, which leads to redundant observations in high-strain regions and leaves high-vorticity areas unobserved. In contrast to inertia sensors with $St = 0.01$ and $St = 0.1$, inertia sensors with $St = 1$ tend to follow ballistic trajectories; thus, they exhibit less clustering and might achieve more effective AOT reconstruction.

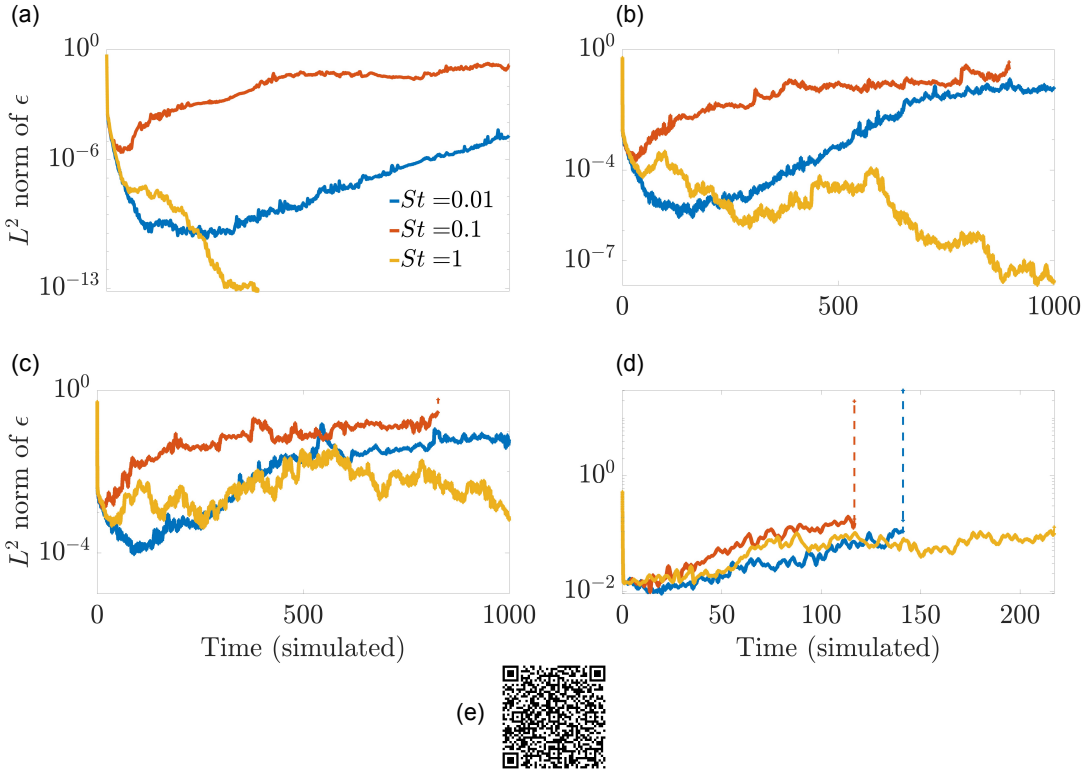


Figure 5.14: Error convergence for inertia sensors with $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c) and $N = 196$ in (d). Lines with different colors are associated with various St, as shown in the legend in (a). To see the clustering of inertia sensors with $St = 0.1$, access the QR code in (e).

5.2.3 Perturbed Lagrangian and perturbed inertia sensors

Small perturbations applied to Lagrangian and inertia sensors had little impact on their trajectories (Figure 5.15) and did not significantly enhance AOT reconstruction. In contrast, larger perturbations led to significant improvements in terms of error convergence, showing a positive correlation with the rate at which the error converges. For example, perturbed Lagrangian sensors with the amplitude of $c = 4$ achieved faster error convergence than ideal Lagrangian sensors (Figure 5.17, see also Figure 5.18 for averaged convergence of the error). A similar positive effect of perturbation can be observed with inertia sensors with $St = 0.1$, where $c = 32$ disrupted clustering (Figure 5.16), and resulted in improved AOT reconstruction (Figure 5.19).

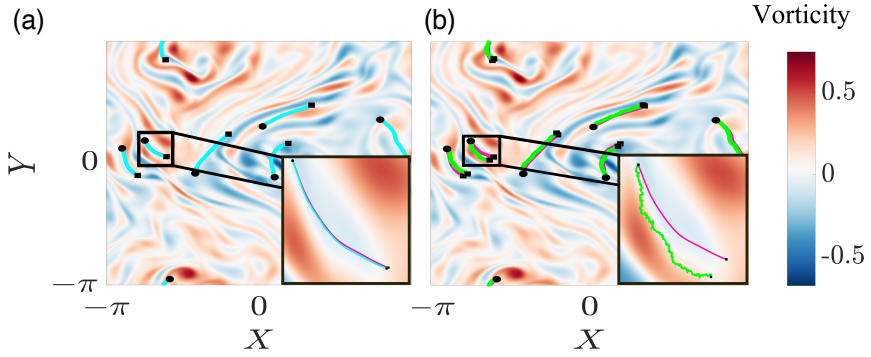


Figure 5.15: Lagrangian (magenta) and perturbed Lagrangian sensors with $c = 0.25$ (cyan) in (a), and $c = 4$ (green) in (b). The black circle and square indicate the starting and ending locations, respectively.

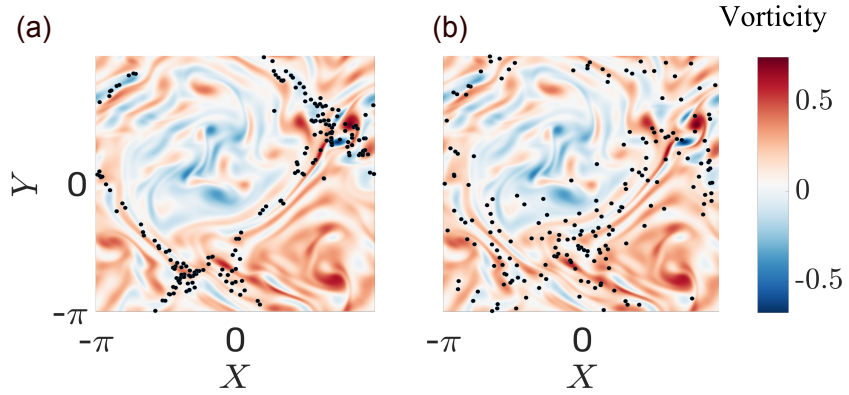


Figure 5.16: Inertia sensors (marked with black dots) with $St = 0.1$ and $N = 196$. Panel (a) illustrates perturbed inertia sensors with noise amplitude $c = 2$, while panel (b) demonstrates a case of larger noise amplitude, $c = 32$.

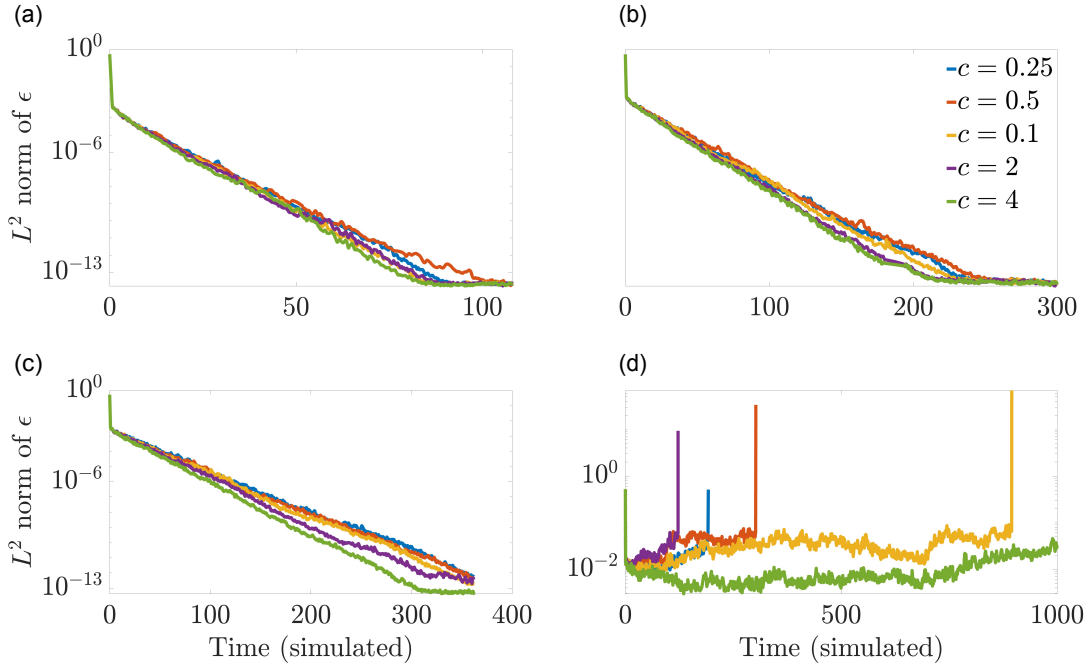


Figure 5.17: Error convergence with perturbed Lagrangian sensors and $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), and $N = 196$ in (d). Lines with different colors are associated with various amplitudes, as shown in the legend in (b).

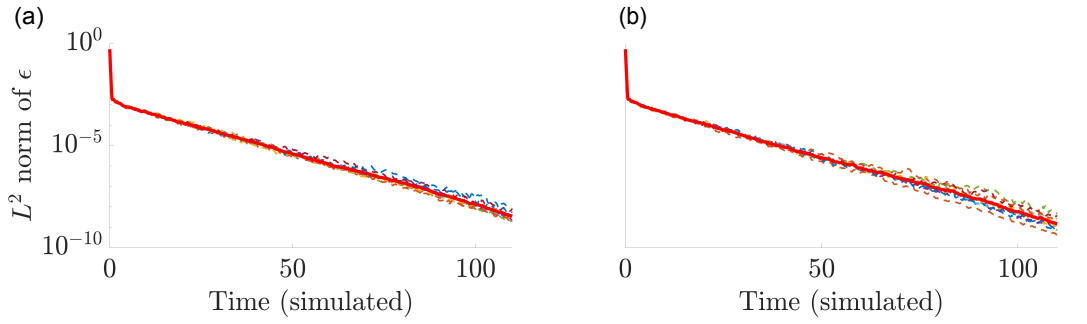


Figure 5.18: Average error convergence with perturbed Lagrangian sensors (red) over the ten numerical experiments with $N = 900$ and $c = 2$ in (a), and $c = 4$ in (b).

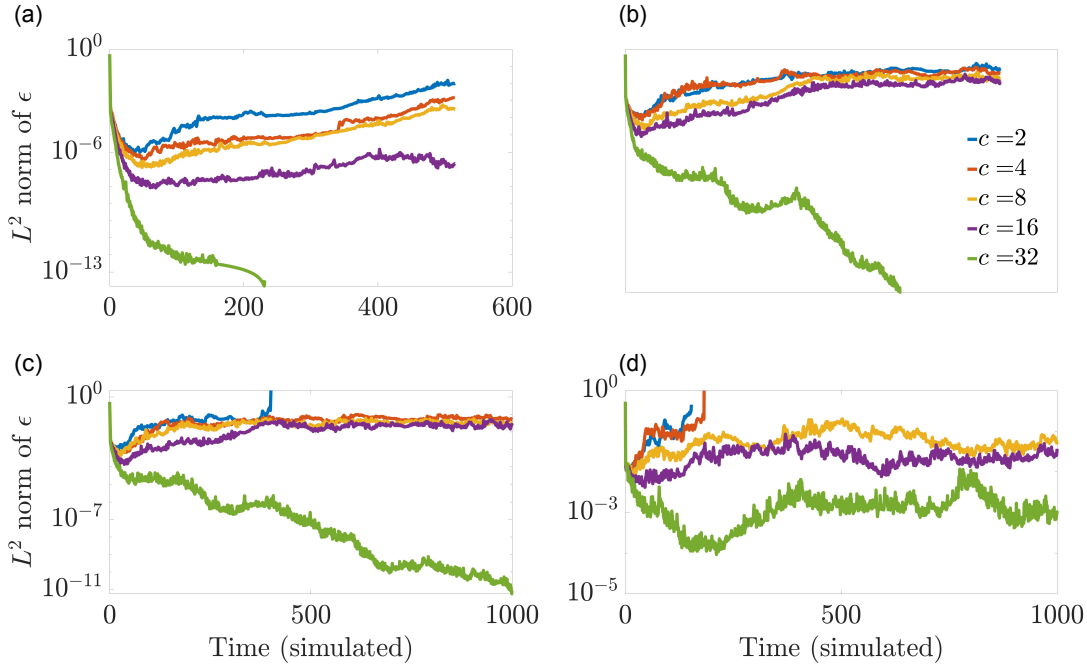


Figure 5.19: Error convergence with perturbed inertia sensors ($St = 0.1$) and $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), and $N = 196$ in (d). Lines with different colors are associated with various amplitudes, as shown in the legend in (b).

The improved AOT performance suggests that ideal Lagrangian and inertia sensors are not optimal, as a simple random perturbation could achieve faster error convergence. This result suggests that employing a better sensor movement strategy, such as directed sensors, could further enhance the AOT performance.

5.2.4 Directed sensors

Similar numerical tests for validating directed sensors were extended from 1D KSE to 2D NSE. These tests showed that directed sensors achieve error convergence to machine precision for $N = 2500$, $N = 900$, and $N = 576$, while for $N = 196$ the error only converged to 10^{-2} (Figure 5.20).

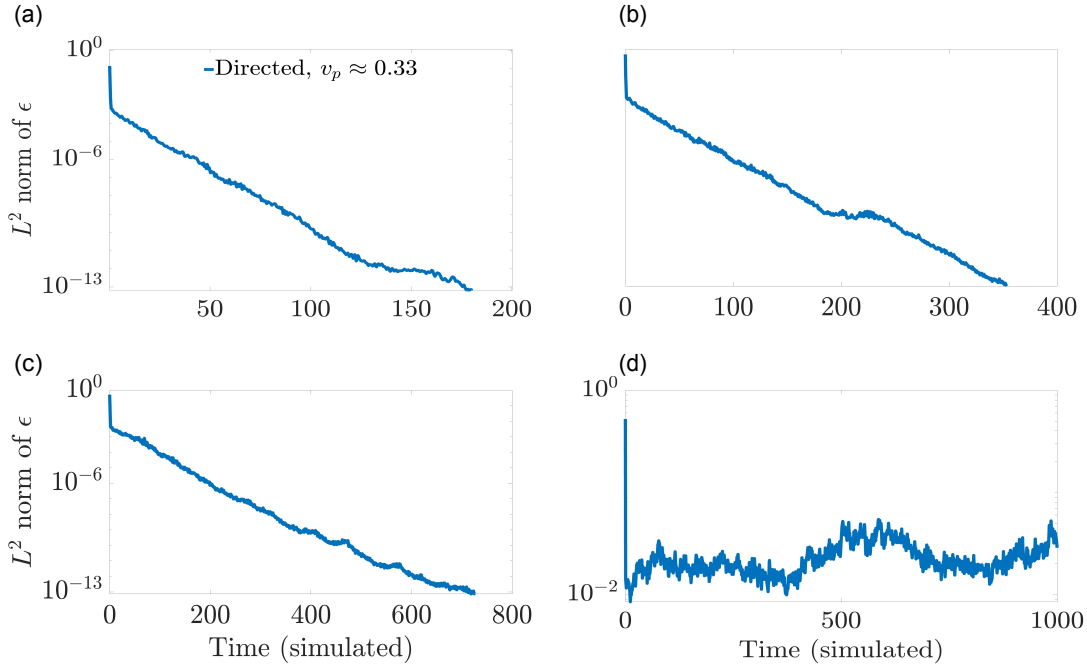


Figure 5.20: Error convergence with directed sensors, moving towards target locations, determined by the Algorithm 1 with $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), $N = 196$ in (d).

Next, a comparison between the actual error convergence rate and the designed rate, R^*, reveals that the former exceeded R^* in both dense and sparse sensor scenarios during the initial phase of convergence (Figure 5.21).

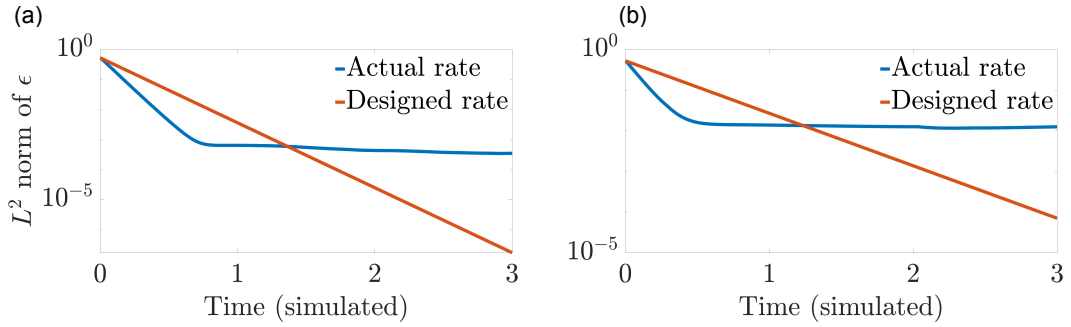


Figure 5.21: Error convergence with Algorithm 1 and designed error convergence rate, R^*, with $N = 2500$ in (a), and $N = 196$ in (b). The blue and red lines illustrate the actual and designed convergence rates, respectively.

These results suggest that the performance of directed sensors aligns with theoretical predictions and enhances the performance of the DA; however, the theory does not necessarily yield optimal global convergence behavior.

The error convergence in the above experiments also depends on the speed of the directed sensors, which determine how fast these reach target locations. For simplicity, in the current research, directed sensors move towards target locations immediately. However, similar to the KSE experiments (see Section 5.1.4), the estimated average velocity of the directed sensors was obtained from (5.1) as $\bar{v}_p = 0.3296$. This value corresponds to a speed approximately three times greater than the characteristic velocity u^*. In addition to "infinite" speed, we limited the sensor speed to $v_p = u^*$ and also achieved error convergence for both dense and sparse sensors. Therefore, directed sensors could be practically implemented in real-world applications.

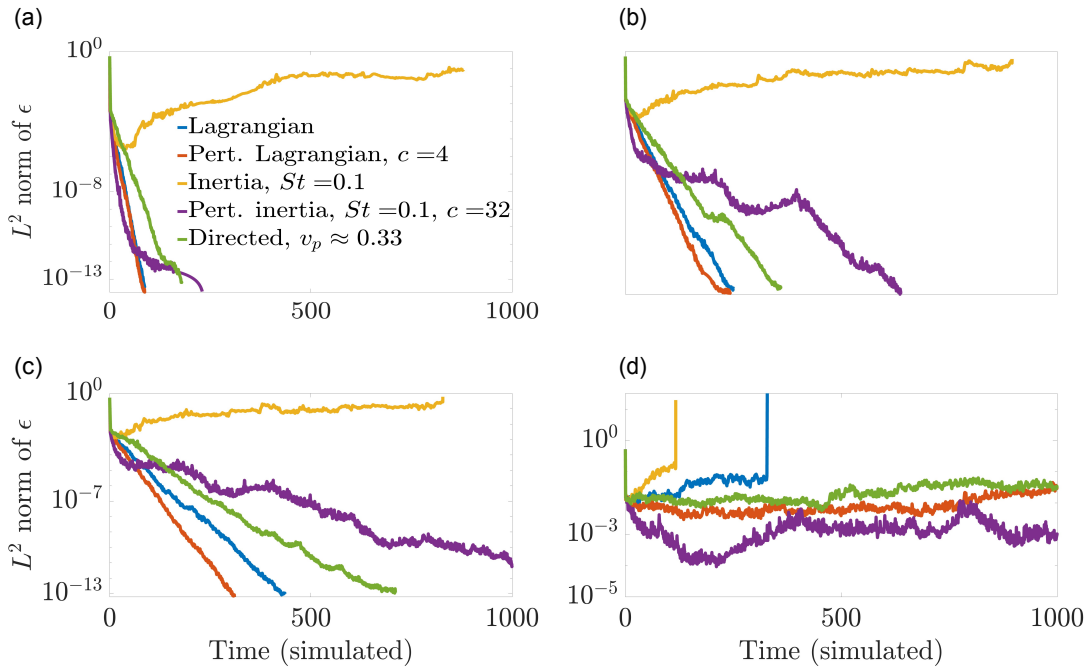


Figure 5.22: Error convergence with Lagrangian, perturbed Lagrangian ($c = 4$), inertia ($St = 0.1$), perturbed inertia ($St = 0.1, c = 32$) and directed sensors with $N = 2500$ in (a), $N = 900$ in (b), $N = 576$ in (c), $N = 196$ in (d). Lines with different colors are associated with various sensor movement strategies, as shown in (a).

5.2.5 Error convergence of sensor movement strategies

After reviewing each sensor movement strategy separately, this subsection will compare them in the same plot (Figure 5.22). First, using inertia sensors with $St = 0.1$ led to divergence due to strong clustering, for both dense and sparse sensors. Perturbing these inertia sensors with high-amplitude noise could prevent such clustering, thus achieving converged AOT reconstruction. However, the needed amplitude of the perturbation was about thirty-two times larger than the characteristic velocity, which is not necessarily practical. Second, Lagrangian sensors achieved effective AOT reconstruction with a large number of sensors, but led to divergence of error with sparse sensors. Applying perturbation to Lagrangian sensors not only increased the convergence speed for dense sensors but also prevented divergence for sparse sensors. This implies that Lagrangian sensors are not necessarily the optimal movement strategy for sampling the flow, as a simple random perturbation improved the AOT convergence. Thirdly, directed sensors achieve effective AOT reconstruction with both dense and sparse sensors. Although the convergence rate for these sensors with $N = 196$ is not higher than that of highly perturbed Lagrangian and inertia sensors, the latter required impractical perturbation to achieve effective AOT reconstruction. Lastly, limiting the speed of directed sensors to $v_p = u^*$ also achieved a successful AOT convergence (not shown). These results suggest that directing sensors toward locations that maximize the AOT convergence rate can enhance reconstruction for sparse sensors.

Chapter 6

Conclusion

6.1 Conclusions

This thesis examines sensor movement strategies for effective reconstruction of dynamical systems using the AOT nudging algorithm. Examples of such strategies include Lagrangian sensors and inertia sensors with and without random perturbation, as well as a proposed optimal sensor movement strategy (i.e., directed sensors guided by maximizing the convergence rate of the AOT algorithm). These strategies are validated on two canonical dynamical systems that model fluid flows: the 1D KSE and the 2D NSE.

Although Lagrangian sensors are often used as the most intuitive way to sample the flow, they have severe drawbacks due to their passive trajectories, which may prevent the successful convergence of the AOT algorithm and other DA methods. First, Lagrangian sensors follow the pathlines of the flow, but not necessarily observing key features. Second, they are sensitive to initial locations, which might lead to inconsistent DA performance. Third, achieving ideal Lagrangian sensors is challenging due to inertial effects, which often lead to clustering in turbulent flows, with some regions of the flow being oversampled while others remain undersampled.

This thesis proposes two sensor movement strategies that improve the AOT reconstruction. The first strategy involves adding a random perturbation to Lagrangian and inertial trajectories. The second strategy directs sensors towards information-rich regions, which is achieved by maximizing the AOT convergence rate, potentially resulting in a fast and effective reconstruction of a dynamical system.

Numerical experiments showed that Lagrangian sensors and inertia sensors with $St \sim \mathcal{O}(1)$ tended to cluster near stagnation points in the 1D KSE and high-strain regions in

the 2D NSE, respectively, thus degrading the AOT performance. To mitigate this clustering, a random perturbation with large amplitude was applied, which disrupted clustering and improved the AOT performance. This improvement suggests that ideal Lagrangian and inertia sensors are not necessarily optimal for effective AOT. In the 1D KSE, the "optimal" directed sensors achieved the fastest error convergence among Lagrangian, inertia, perturbed Lagrangian, and perturbed inertia sensors, demonstrating that directing sensors to locations that maximize the convergence rate can significantly improve the DA reconstruction.

In the 2D NSE, the results showed that directed sensors achieved converged AOT reconstruction for both dense and sparse scenarios. In contrast, Lagrangian and inertia sensors led to error divergence for sparse sensors. Furthermore, constraining the speed of directed sensors to the characteristic velocity also resulted in converged AOT reconstruction, making them practical.

6.2 Future Work

While directed sensors achieved the effective AOT reconstruction, several limitations in their performance and movement design remain. For instance, in 2D NSE, directed sensors converged the error of the AOT reconstruction at a slower rate than Lagrangian sensors for dense sensors. This could be due to two reasons: (i) the current error convergence estimate of the AOT algorithm is not sharp; (ii) the Algorithm 1 is rather naive and might not be ideal for determining optimal target sensor locations.

To address (i), one could improve the analysis by considering a nonlinear AOT algorithm, as proposed in [26, 80], which has been shown to achieve faster convergence rates compared to classical linear AOT. Subsequently, (ii) can be solved by designing a better algorithm for determining target sensor locations. This might include leveraging ideas of adaptive mesh refinement techniques, which also increase element density in regions with high solution gradients. Additionally, ideas from gradient-based error indicators or weighted centroidal Voronoi tessellations (CVT) might also improve target sensor locations.

Other limitations of proposed directed sensors include that they are typically flow-dependent. To address this limitation, the effectiveness of these sensors can be tested in other important nonlinear flows, such as 3D NSE or Rayleigh–Bénard convection. We will leave the investigation of this effectiveness to future work.

References

[1] M. U. Altaf, E. S. Titi, T. Gebrael, O. M. Knio, L. Zhao, M. F. McCabe, and I. Hoteit. Downscaling the 2D Bénard convection equations using continuous data assimilation. *Computational Geosciences*, 21(3):393–410, June 2017.

[2] M. U. Altaf, E. S. Titi, O. M. Knio, L. Zhao, M. F. McCabe, and I. Hoteit. Downscaling the 2D Benard Convection Equations Using Continuous Data Assimilation, December 2015. arXiv:1512.04671 [math].

[3] Jeffrey L. Anderson. An Ensemble Adjustment Kalman Filter for Data Assimilation. *Monthly Weather Review*, 129(12):2884–2903, December 2001.

[4] Richard A. Anthes. Data Assimilation and Initialization of Hurricane Prediction Models. *Journal of the Atmospheric Sciences*, 31(3):702–719, April 1974.

[5] Mark Asch, Marc Bocquet, and Maëlle Nodet. *Data Assimilation: Methods, Algorithms, and Applications*. Society for Industrial and Applied Mathematics, Philadelphia, PA, December 2016.

[6] A. Azouani and E. S. Titi. Feedback control of nonlinear dissipative systems by finite determining parameters—a reaction-diffusion paradigm. *Evolution Equations and Control Theory*, 3(4):579–594, 2014.

[7] Abderrahim Azouani, Eric Olson, and Edriss S. Titi. Continuous Data Assimilation Using General Interpolant Observables. *Journal of Nonlinear Science*, 24(2):277–304, April 2014. arXiv:1304.0997 [nlin, physics:physics].

[8] A. V. Babin and M. I. Vishik. Attractors of partial differential evolution equations in an unbounded domain. *Proceedings of the Royal Society of Edinburgh: Section A Mathematics*, 116(3-4):221–243, 1990.

[9] John C. Baez, Steve Huntsman, and Cheyne Weis. The Kuramoto-Sivashinsky Equation, October 2022. arXiv:2210.01711 [math].

[10] Heinz H. Bauschke and Patrick L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. CMS Books in Mathematics. Springer International Publishing, Cham, 2017.

[11] J. Bec, A. Celani, M. Cencini, and S. Musacchio. Clustering and collisions of heavy particles in random smooth flows. *Physics of Fluids*, 17(7):073301, July 2005.

[12] Jérémie Bec. Fractal clustering of inertial particles in random flows. *Physics of Fluids*, 15(11):L81–L84, November 2003.

[13] Amir Beck. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with Python and MATLAB, Second Edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, January 2023.

[14] L. Mark Berliner, Zhan-Qian Lu, and Chris Snyder. Statistical Design for Adaptive Weather Observations. *Journal of the Atmospheric Sciences*, 56(15):2536–2552, August 1999.

[15] Stefano Berti, Francisco Alves Dos Santos, Guglielmo Lacorata, and Angelo Vulpiani. Lagrangian drifter dispersion in the southwestern Atlantic Ocean. *Journal of Physical Oceanography*, 41(9):1659–1672, September 2011. arXiv:1110.3590 [nlin].

[16] Hakima Bessaih, Eric Olson, and E. S. Titi. Continuous Data Assimilation with Stochastically Noisy Data. *Nonlinearity*, 28(3):729–753, March 2015. arXiv:1406.1533 [math].

[17] Luca Biferale, Fabio Bonaccorso, Michele Buzzicotti, Patricio Clark Di Leoni, and Kristian Gustavsson. Zermelo's problem: Optimal point-to-point navigation in 2D turbulent flows using Reinforcement Learning. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 29(10):103138, October 2019. arXiv:1907.08591 [nlin].

[18] Animikh Biswas, Zachary Bradshaw, Michael Jolly, University of Maryland Baltimore County, USA, University of Arkansas, USA, and Indiana University, USA. Convergence of a mobile data assimilation scheme for the 2D Navier-Stokes equations. *Discrete and Continuous Dynamical Systems*, 43(11):4042–4068, 2023.

[19] Marc Bocquet, Carlos A. Pires, and Lin Wu. Beyond Gaussian Statistical Modeling in Geophysical Data Assimilation. *Monthly Weather Review*, 138(8):2997–3023, August 2010. Publisher: American Meteorological Society.

[20] Jonathan Borwein and Adrian Lewis. *Convex Analysis and Nonlinear Optimization*. CMS Books in Mathematics. Springer New York, New York, NY, 2006.

[21] Burak Boyacıoğlu, Mahnoush Babaei, Amanuel H. Mamo, Sarah Bergbreiter, Thomas L. Daniel, and Kristi A. Morgansen. Sensor Placement for Flapping Wing Model Using Stochastic Observability Gramians, July 2024. arXiv:2310.00127 [eess].

[22] Michele Buzzicotti, Luca Biferale, Fabio Bonaccorso, Patricio Clark di Leoni, and Kristian Gustavsson. Optimal control of point-to-point navigation in turbulent time-dependent flows using Reinforcement Learning, February 2021. arXiv:2103.00329 [eess].

[23] Ulrich Callies, Markus Kreus, Wilhelm Petersen, and Yoana G. Voynova. On Using Lagrangian Drift Simulations to Aid Interpretation of in situ Monitoring Data. *Frontiers in Marine Science*, 8, July 2021. Publisher: Frontiers Media SA.

[24] Enrico Calzavarini, Massimo Cencini, Detlef Lohse, and Federico Toschi. Quantifying Turbulence-Induced Segregation of Inertial Particles. *Physical Review Letters*, 101(8):084504, August 2008.

[25] Elizabeth Carlson, Joshua Hudson, and Adam Larios. Parameter Recovery and Sensitivity Analysis for the 2D Navier-Stokes Equations Via Continuous Data Assimilation, December 2018. arXiv:1812.07646 [math].

[26] Elizabeth Carlson, Adam Larios, and Edriss S. Titi. Super-Exponential Convergence Rate of a Nonlinear Continuous Data Assimilation Algorithm: The 2D Navier–Stokes Equation Paradigm. *Journal of Nonlinear Science*, 34(2):37, April 2024.

[27] Elizabeth Carlson, Luke P. Van Roekel, Humberto C Godinez, Mark R. Petersen, and Adam Larios. Exploring a New Computationally Efficient Data Assimilation Algorithm For Ocean Models, June 2021.

[28] Reiyu Chein and J. N. Chung. Simulation of particle dispersion in a two-dimensional mixing layer. *AIChE Journal*, 34(6):946–954, June 1988.

[29] Kevin K. Chen and Clarence W. Rowley. H_2 optimal actuator and sensor placement in the linearised complex Ginzburg–Landau system. *Journal of Fluid Mechanics*, 681:241–260, August 2011. Publisher: Cambridge University Press (CUP).

[30] Nan Chen, Shubin Fu, and Georgy E. Manucharyan. An efficient and statistically accurate Lagrangian data assimilation algorithm with applications to discrete element sea ice models. *Journal of Computational Physics*, 455:111000, April 2022.

[31] Yan Chen and Dean S. Oliver. Information content in Lagrangian sensor measurements for reservoir characterization. *Computers & Geosciences*, 55:96–109, June 2013.

[32] Yat Tin Chow, Wing Tat Leung, and Ali Pakzad. Continuous data assimilation for two-phase flow: Analysis and simulations. *Journal of Computational Physics*, 466:111395, October 2022.

[33] Bernardo Cockburn, Don Jones, and Edriss Titi. Estimating the number of asymptotic degrees of freedom for nonlinear dissipative systems. *Mathematics of Computation*, 66(219):1073–1087, 1997.

[34] P. Constantin, C. Foias, B. Nicolaenko, and R. Teman. *Integral Manifolds and Inertial Manifolds for Dissipative Partial Differential Equations*, volume 70 of *Applied Mathematical Sciences*. Springer New York, New York, NY, 1989.

[35] R. Dascaliuc, C. Gname, and M. S. Jolly. Relations Between Energy and Enstrophy on the Global Attractor of the 2-D Navier-Stokes Equations. *Journal of Dynamics and Differential Equations*, 17(4):643–736, October 2005.

[36] Srinivas Desamsetti, Hari Prasad Dasari, Sabique Langodan, Edriss S. Titi, Omar Knio, and Ibrahim Hoteit. Efficient dynamical downscaling of general circulation models using continuous data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 145(724):3175–3194, October 2019.

[37] Levi DeVries, Sharanya J. Majumdar, and Derek A. Paley. Observability-based Optimization of Coordinated Sampling Trajectories for Recursive Estimation of a Strong, Spatially Varying Flowfield. *Journal of Intelligent & Robotic Systems*, 70(1-4):527–544, April 2013.

[38] Lisa Deyle, Thomas H. Badewien, Oliver Wurl, and Jens Meyerjürgens. Lagrangian surface drifter observations in the North Sea: an overview of high-resolution tidal dynamics and surface currents. *Earth System Science Data*, 16(4):2099–2112, April 2024.

[39] J.K. Eaton and J.R. Fessler. Preferential concentration of particles by turbulence. *International Journal of Multiphase Flow*, 20:169–209, August 1994. Publisher: Elsevier BV.

[40] 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, June 2020. arXiv:1902.07358 [physics].

[41] Mahdi Esmaily-Moghadam and Ali Mani. Analysis of the clustering of inertial particles in turbulent flows. *Physical Review Fluids*, 1(8):084202, December 2016.

[42] Geir Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5):10143–10162, May 1994.

[43] Geir Evensen. Advanced Data Assimilation for Strongly Nonlinear Dynamics. *Monthly Weather Review*, 125(6):1342–1354, June 1997. Publisher: American Meteorological Society.

[44] Geir Evensen. *Data Assimilation: The Ensemble Kalman Filter*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[45] Jianing Fang, Chuheng Hu, Nour Smaoui, Doug Carlson, Jayant Gupchup, Razvan Musaloiu-E, Chieh-Jan Mike Liang, Marcus Chang, Omprakash Gnawali, Tamas Budavari, Andreas Terzis, Katalin Szlavecz, and Alexander S. Szalay. Wireless sensor network for in situ soil moisture monitoring, February 2021. arXiv:2102.10260 [eess].

[46] Aseel Farhat, Adam Larios, Vincent R. Martinez, and Jared P. Whitehead. Identifying the body force from partial observations of a 2D incompressible velocity field, February 2024. arXiv:2302.04701 [physics].

[47] Aseel Farhat, Evelyn Lunasin, and Edriss S. Titi. Abridged continuous data assimilation for the 2D Navier-Stokes equations utilizing measurements of only one component of the velocity field. *Journal of Mathematical Fluid Mechanics*, 18(1):1–23, March 2016. arXiv:1504.05978 [math].

[48] Ana Fernández Tena and Pere Casan Clarà. Use of Computational Fluid Dynamics in Respiratory Medicine. *Archivos de Bronconeumología (English Edition)*, 51(6):293–298, June 2015.

[49] C. Foias, O. Manley, R. Rosa, and R. Temam. *Navier-Stokes Equations and Turbulence*. Cambridge University Press, 1 edition, August 2001.

[50] C. Foias, O.P. Manley, R. Temam, and Y.M. Treve. Asymptotic analysis of the navier-stokes equations. *Physica D: Nonlinear Phenomena*, 9(1-2):157–188, October 1983.

[51] C Foias and E S Titi. Determining nodes, finite difference schemes and inertial manifolds. *Nonlinearity*, 4(1):135–153, February 1991.

[52] Ciprian Foias, Michael S. Jolly, Rostyslav Kravchenko, and Edriss S. Titi. A unified approach to determining forms for the 2D Navier-Stokes equations - the general interpolants case. *Russian Mathematical Surveys*, 69(2):359–381, April 2014. arXiv:1309.0247 [math].

[53] Ciprian Foias, Cecilia F. Mondaini, and Edriss S. Titi. A discrete data assimilation scheme for the solutions of the 2D Navier-Stokes equations and their statistics, May 2016. arXiv:1602.05995 [math].

[54] Trenton Franz, Adam Larios, and Collin Victor. The Bleeps, the Sweeps, and the Creeps: Convergence Rates for Dynamic Observer Patterns via Data Assimilation for the 2D Navier-Stokes Equations. *Computer Methods in Applied Mechanics and Engineering*, 392:114673, March 2022. arXiv:2110.10362 [cs, math].

[55] Matthew Gardner, Adam Larios, Leo G. Rebholz, Duygu Vargun, and Camille Zerfas. Continuous data assimilation applied to a velocity-vorticity formulation of the 2D Navier-Stokes equations. *Electronic Research Archive*, 29(3):2223–2247, 2021.

[56] Wanli Gu, Shuo Cai, Yunfeng Hu, Hui Zhang, and Hong Chen. Trajectory planning and tracking control of a ground mobile robot:A reconstruction approach towards space vehicle. *ISA Transactions*, 87:116–128, April 2019.

[57] Peter Gunnarson, Ioannis Mandralis, Guido Novati, Petros Koumoutsakos, and John O. Dabiri. Learning efficient navigation in vortical flow fields. *Nature Communications*, 12(1):7143, December 2021.

[58] D.B. Haidvogel, H. Arango, W.P. Budgell, B.D. Cornuelle, E. Curchitser, E. Di Lorenzo, K. Fennel, W.R. Geyer, A.J. Hermann, L. Lanerolle, J. Levin, J.C. McWilliams, A.J. Miller, A.M. Moore, T.M. Powell, A.F. Shchepetkin, C.R. Sherwood, R.P. Signell, J.C. Warner, and J. Wilkin. Ocean forecasting in terrain-following coordinates: Formulation and skill assessment of the Regional Ocean Modeling System. *Journal of Computational Physics*, 227(7):3595–3624, March 2008.

[59] Jack K. Hale. *Asymptotic Behavior of Dissipative Systems*. Number v.25 in Mathematical Surveys and Monographs. American Mathematical Society, Providence, 2014.

[60] G. Haller. Lagrangian coherent structures from approximate velocity data. *Physics of Fluids*, 14(6):1851–1861, June 2002.

[61] George Haller. Lagrangian Coherent Structures. *Annual Review of Fluid Mechanics*, 47(1):137–162, January 2015.

[62] Nils E.L. Haugen, Axel Brandenburg, Christer Sandin, and Lars Mattsson. Spectral characterisation of inertial particle clustering in turbulence. *Journal of Fluid Mechanics*, 934:A37, March 2022.

[63] Kevin D. Heaney, Pierre F. J. Lermusiaux, Timothy F. Duda, and Patrick J. Haley. Validation of genetic algorithm-based optimal sampling for ocean data assimilation. *Ocean Dynamics*, 66(10):1209–1229, October 2016.

[64] Sigrid Passano Hellan, Christopher G. Lucas, and Nigel H. Goddard. Optimising Placement of Pollution Sensors in Windy Environments, August 2022. arXiv:2012.10770 [cs].

[65] William D. Henshaw, Heinz-Otto Kreiss, and Jacob Yström. Numerical Experiments on the Interaction Between the Large- and Small-Scale Motions of the Navier-Stokes Equations. *Multiscale Modeling & Simulation*, 1(1):119–149, January 2003.

[66] Sebastian Hofmann, Christian Weiland, Jürgen Fitschen, Alexandra Von Kameke, Marko Hoffmann, and Michael Schlüter. Lagrangian sensors in a stirred tank reactor: Comparing trajectories from 4D-Particle Tracking Velocimetry and Lattice-Boltzmann simulations. *Chemical Engineering Journal*, 449:137549, December 2022.

[67] Andrew H. Jazwinski. *Stochastic processes and filtering theory*. Number v. 64 in Mathematics in science and engineering. Academic Press, New York, 1970.

[68] Bo Jin, Simon J. Illingworth, and Richard D. Sandberg. Optimal sensor and actuator placement for feedback control of vortex shedding. *Journal of Fluid Mechanics*, 932:A2, February 2022.

[69] Michael S. Jolly, Vincent R. Martinez, Eric J. Olson, and Edriss S. Titi. Continuous data assimilation with blurred-in-time measurements of the surface quasi-geostrophic equation, September 2018. arXiv:1809.00106 [math].

[70] Don Jones and Edriss Titi. Upper Bounds on the Number of Determining Modes, Nodes, and Volume Elements for the Navier-Stokes Equations. *Indiana University Mathematics Journal*, 42(3):875, 1993.

[71] Don A. Jones and Edriss S. Titi. Determining finite volume elements for the 2D Navier-Stokes equations. *Physica D: Nonlinear Phenomena*, 60(1-4):165–174, November 1992.

[72] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, March 1960.

[73] Wei Kang and Liang Xu. Optimal placement of mobile sensors for data assimilations. *Tellus A: Dynamic Meteorology and Oceanography*, 64(1):17133, December 2012.

[74] Niharika Karnik, Mohammad G. Abdo, Carlos E. Estrada Perez, Jun Soo Yoo, Joshua J. Cogliati, Richard S. Skifton, Pattrick Calderoni, Steven L. Brunton, and Krithika Manohar. Constrained optimization of sensor placement for nuclear digital twins, February 2024. arXiv:2306.13637 [cs, math].

[75] Eric J Kostelich, Yang Kuang, Joshua M McDaniel, Nina Z Moore, Nikolay L Martirosyan, and Mark C Preul. Accurate state estimation from uncertain data and models: an application of data assimilation to mathematical models of human brain tumors. *Biology Direct*, 6(1):64, 2011.

[76] Andreas Krause, Carlos Guestrin, Anupam Gupta, and Jon Kleinberg. Near-optimal sensor placements: maximizing information while minimizing communication cost. In *Proceedings of the fifth international conference on Information processing in sensor networks - IPSN '06*, page 2, Nashville, Tennessee, USA, 2006. ACM Press.

[77] Y. Kuramoto and T. Tsuzuki. On the Formation of Dissipative Structures in Reaction-Diffusion Systems: Reductive Perturbation Approach. *Progress of Theoretical Physics*, 54(3):687–699, September 1975.

[78] Hans R. Künsch. Particle filters. *Bernoulli*, 19(4), September 2013. arXiv:1309.7807 [math].

[79] A. Laforgia, X. Leoncini, L. Kuznetsov, and G.M. Zaslavsky. Passive tracer dynamics in 4 point-vortex flow. *The European Physical Journal B*, 20(3):427–440, April 2001.

[80] Adam Larios and Yuan Pei. Nonlinear Continuous Data Assimilation, March 2017. arXiv:1703.03546 [math].

[81] Adam Larios, Yuan Pei, and Collin Victor. The second-best way to do sparse-in-time continuous data assimilation: Improving convergence rates for the 2D and 3D Navier-Stokes equations, March 2023. arXiv:2303.03495 [math].

[82] Adam Larios and Collin Victor. Continuous Data Assimilation with a Moving Cluster of Data Points for a Reaction Diffusion Equation: A Computational Study, December 2018. arXiv:1812.01686 [math].

[83] Adam Larios and Collin Victor. Continuous Data Assimilation for the 3D and Higher-Dimensional Navier–Stokes equations with Higher-Order Fractional Diffusion, June 2023. arXiv:2307.00096 [math].

[84] Naomi Ehrich Leonard, Derek A. Paley, Francois Lekien, Rodolphe Sepulchre, David M. Fratantoni, and Russ E. Davis. Collective Motion, Sensor Networks, and Ocean Sampling. *Proceedings of the IEEE*, 95(1):48–74, January 2007.

[85] Bangjun Li, Haoran Liu, and Ruzhu Wang. Efficient Sensor Placement for Signal Reconstruction Based on Recursive Methods. *IEEE Transactions on Signal Processing*, 69:1885–1898, 2021.

[86] Xu Liu, Wen Yao, Wei Peng, Zhuojia Fu, Zixue Xiang, and Xiaoqian Chen. A physics-driven sensor placement optimization methodology for temperature field reconstruction. *Applied Thermal Engineering*, 257:124476, December 2024.

[87] A. C. Lorenc. Analysis methods for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 112(474):1177–1194, October 1986.

[88] David G. Luenberger. Observing the State of a Linear System. *IEEE Transactions on Military Electronics*, 8(2):74–80, 1964. Publisher: Institute of Electrical and Electronics Engineers (IEEE).

[89] Evelyn Lunasin and Edriss S. Titi. Finite determining parameters feedback control for distributed nonlinear dissipative systems - a computational study, 2015. Version Number: 2.

[90] Kevin M. Lynch, Ira B. Schwartz, Peng Yang, and Randy A. Freeman. Decentralized Environmental Modeling by Mobile Sensor Networks. *IEEE Transactions on Robotics*, 24(3):710–724, June 2008.

[91] Krithika Manohar, Bingni W. Brunton, J. Nathan Kutz, and Steven L. Brunton. Data-Driven Sparse Sensor Placement for Reconstruction. *IEEE Control Systems*, 38(3):63–86, June 2018. arXiv:1701.07569 [cs, math].

[92] Krithika Manohar, J. Nathan Kutz, and Steven L. Brunton. Optimal Sensor and Actuator Selection Using Balanced Model Reduction. *IEEE Transactions on Automatic Control*, 67(4):2108–2115, April 2022.

[93] Neda Mardani, Mohammadreza Khanarmuei, Kabir Suara, Richard Brown, Adrian McCallum, and Roy C. Sidle. Lagrangian Data Assimilation for Improving Model Estimates of Velocity Fields and Residual Currents in a Tidal Estuary. *Applied Sciences*, 11(22):11006, November 2021.

[94] Mateusz Matuszak, Johannes Röhrs, Pål Erik Isachsen, and Martina Idžanović. Persistence and Robustness of Lagrangian Coherent Structures, April 2024.

[95] Lewis McCaffrey and Alexander L. Koeberle. An Economical Open-Source Lagrangian Drifter Design to Measure Deep Currents in Lakes. *Water Resources Research*, 60(10), October 2024. Publisher: American Geophysical Union (AGU).

[96] Jiazhong Mei, Steven L. Brunton, and J. Nathan Kutz. Mobile Sensor Path Planning for Kalman Filter Spatiotemporal Estimation. *Sensors*, 24(12):3727, June 2024.

[97] Cecilia F. Mondaini, Edriss S. Titi, Animikh Biswas, and Ciprian Foias. Downscaling data assimilation algorithm with applications to statistical solutions of the Navier–Stokes equations. *Annales de l'Institut Henri Poincaré C, Analyse non linéaire*, 36(2):295–326, April 2019.

[98] Ning Ning and Collin Victor. An Assessment of Ensemble Kalman Filter and Azouani-Olson-Titi Algorithms for Data Assimilation: A Comparative Study, July 2024. arXiv:2407.17424 [math].

[99] Eric Olson and Edriss S. Titi. Determining modes and Grashof number in 2D turbulence: a numerical case study. *Theoretical and Computational Fluid Dynamics*, 22(5):327–339, August 2008.

[100] Nicholas T. Ouellette, Charlie A. R. Hogg, and Yang Liao. Correlating Lagrangian structures with forcing in two-dimensional flow. *Physics of Fluids*, 28(1):015105, January 2016.

[101] Ciprian Foias Peter Constantin. Navier-Stokes equations: Peter Constantin and Ciprian Foias: Chicago Lectures in Mathematics, The University of Chicago Press, Chicago, London, 1988, ix + 190pp. *Acta Applicandae Mathematicae*, 28(3):295–296, September 1992.

[102] S. V. Prants. Dynamical systems theory approach in oceanography: a review on achievements, limitations, verification and validation of Lagrangian methods. *Frontiers in Marine Science*, 12, July 2025. Publisher: Frontiers Media SA.

[103] Zhaoxia Pu and Eugenia Kalnay. Numerical Weather Prediction Basics: Models, Numerical Methods, and Data Assimilation. In Qingyun Duan, Florian Pappenberger, Jutta Thielen, Andy Wood, Hannah L. Cloke, and John C. Schaake, editors, *Handbook of Hydrometeorological Ensemble Forecasting*, pages 1–31. Springer Berlin Heidelberg, Berlin, Heidelberg, 2018.

[104] Juri Ranieri, Amina Chebira, and Martin Vetterli. Near-Optimal Sensor Placement for Linear Inverse Problems. *IEEE Transactions on Signal Processing*, 62(5):1135–1146, March 2014. arXiv:1305.6292 [cs, math].

[105] Ryan Rautenbach, Sebastian Hofmann, Lukas Buntkiel, Jan Schäfer, Sebastian Felix Reinecke, Marko Hoffmann, Uwe Hampel, and Michael Schlüter. Dynamics of Lagrangian Sensor Particles: The Effect of Non-Homogeneous Mass Distribution. *Processes*, 12(8):1617, August 2024.

[106] Walter C. Reade and Lance R. Collins. Effect of preferential concentration on turbulent collision rates. *Physics of Fluids*, 12(10):2530–2540, October 2000.

[107] Luke Reid. An Introduction to Biomedical Computational Fluid Dynamics. In Paul M. Rea, editor, *Biomedical Visualisation*, volume 1334, pages 205–222. Springer International Publishing, Cham, 2021. Series Title: Advances in Experimental Medicine and Biology.

[108] Jc Robinson and C Pierre. Infinite-Dimensional Dynamical Systems: An Introduction to Dissipative Parabolic PDEs and the Theory of Global Attractors. Cambridge Texts in Applied Mathematics. *Applied Mechanics Reviews*, 56(4):B54–B55, July 2003.

[109] Yuji Saito, Taku Nonomura, Keigo Yamada, Kumi Nakai, Takayuki Nagata, Keisuke Asai, Yasuo Sasaki, and Daisuke Tsubakino. Determinant-Based Fast Greedy Sensor Selection Algorithm. *IEEE Access*, 9:68535–68551, 2021.

[110] George R. Sell and Yuncheng You. *Dynamics of Evolutionary Equations*, volume 143 of *Applied Mathematical Sciences*. Springer New York, New York, NY, 2002.

[111] Shawn C. Shadden, Francois Lekien, and Jerrold E. Marsden. Definition and properties of Lagrangian coherent structures from finite-time Lyapunov exponents in two-dimensional aperiodic flows. *Physica D: Nonlinear Phenomena*, 212(3-4):271–304, December 2005.

[112] Bo-Wen Shen, Roger A. Pielke, Xubin Zeng, Jong-Jin Baik, Sara Faghih-Naini, Jialin Cui, and Robert Atlas. Is Weather Chaotic?: Coexistence of Chaos and Order within a Generalized Lorenz Model. *Bulletin of the American Meteorological Society*, 102(1):E148–E158, January 2021.

[113] G.I. Sivashinsky. Nonlinear analysis of hydrodynamic instability in laminar flames—I. Derivation of basic equations. *Acta Astronautica*, 4(11-12):1177–1206, November 1977.

[114] Astrid Marie Skålvik, Camilla Saetre, Kjell-Eivind Frøysa, Ranveig N. Bjørk, and Anders Tengberg. Challenges, limitations, and measurement strategies to ensure data quality in deep-sea sensors. *Frontiers in Marine Science*, 10, April 2023. Publisher: Frontiers Media SA.

[115] Julia Slingo and Tim Palmer. Uncertainty in weather and climate prediction. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 369(1956):4751–4767, December 2011.

[116] Pieter B. Smit, Isabel A. Houghton, Kalina Jordanova, Thomas Portwood, Evan Shapiro, David Clark, Michael Sosa, and Tim T. Janssen. Assimilation of distributed ocean wave sensors, March 2020. arXiv:2003.04435 [physics].

[117] Harold W. Sorenson and Institute of Electrical and Electronics Engineers, editors. *Kalman filtering, theory and application.* IEEE Press selected reprint series. IEEE Press, New York, 1985.

[118] M. S. Sreeraj, S. L. Sruthi, and S. G. Sumod. Modeling in Fluid Mechanics: Atmospheric Modeling Using Navier Stokes Equation. In *New Advances in Materials Technologies*, pages 59–86. Apple Academic Press, New York, 1 edition, February 2024.

[119] Tyler H. Summers and John Lygeros. Optimal Sensor and Actuator Placement in Complex Dynamical Networks, March 2014. arXiv:1306.2491 [math].

[120] Olivier Talagrand and Philippe Courtier. Variational Assimilation of Meteorological Observations With the Adjoint Vorticity Equation. I: Theory. *Quarterly Journal of the Royal Meteorological Society*, 113(478):1311–1328, October 1987.

[121] Roger Temam. *Infinite-Dimensional Dynamical Systems in Mechanics and Physics*, volume 68 of *Applied Mathematical Sciences*. Springer New York, New York, NY, 1997.

[122] Edriss S. Titi and Collin Victor. On the inadequacy of nudging data assimilation algorithms for non-dissipative systems: A computational examination of the Korteweg de-Vries and Lorenz equations, November 2024. arXiv:2411.08273 [math].

[123] Vasileios Tzoumas, Ali Jadbabaie, and George J. Pappas. Sensor Placement for Optimal Kalman Filtering: Fundamental Limits, Submodularity, and Algorithms, May 2020. arXiv:1509.08146 [math].

[124] F.E. Uilhoorn. An algorithm for finding optimal sensor placement and model uncertainty in data assimilation applied to Riemann problems. *Applied Mathematical Modelling*, 103:649–673, March 2022.

[125] P. T. M. Vermeulen and A. W. Heemink. Model-Reduced Variational Data Assimilation. *Monthly Weather Review*, 134(10):2888–2899, October 2006.

[126] Bowen Wang, Zhibin Sun, Xinyue Jiang, Jun Zeng, and Runqing Liu. Kalman Filter and Its Application in Data Assimilation. *Atmosphere*, 14(8):1319, August 2023.

[127] Chen Wang, Victoria Huang, Gang Chen, Hui Ma, Bryce Chen, and Jochen Schmidt. Learning to Optimise Climate Sensor Placement using a Transformer, March 2024. arXiv:2310.12387 [cs].

[128] Jan Williams, Olivia Zahn, and J. Nathan Kutz. Data-driven sensor placement with shallow decoder networks, February 2022. arXiv:2202.05330 [math].

[129] Qingfang Wu, Andrew Tinka, Kevin Weekly, Jonathan Beard, and Alexandre M. Bayen. Variational L agrangian data assimilation in open channel networks. *Water Resources Research*, 51(4):1916–1938, April 2015.

[130] Shu-Chih Yang, Eugenia Kalnay, and Brian Hunt. Handling Nonlinearity in an Ensemble Kalman Filter: Experiments with the Three-Variable Lorenz Model. *Monthly Weather Review*, 140(8):2628–2646, August 2012. Publisher: American Meteorological Society.

[131] B. Yildirim, C. Chryssostomidis, and G.E. Karniadakis. Efficient sensor placement for ocean measurements using low-dimensional concepts. *Ocean Modelling*, 27(3-4):160–173, January 2009.

APPENDICES

Appendix A

Determining target sensor locations

Algorithm 1 Target sensor locations

Input: Distance function: h, Mesh: x, Domain: Ω, Number of sensors: N
Output: Target sensor locations, S

Initialize $\xi = \{(\Omega, N)\}$ ▷ Define set for storing intervals
while $|\xi| < N$ **do**

    Pop Ω from ξ and split it at the midpoint into two subintervals Ω_l, and Ω_r.
    Calculate N_l, N_r for Ω_l, and Ω_r, respectively, using the determine_density function
    Add Ω_l, Ω_r to ξ with their corresponding densities N_l, N_r
    Select $\Omega = \arg\max_{i \in \xi}(N_i)$

end while

$S \leftarrow \frac{a_i + b_i}{2}$, where $[a, b] = \{x \in \Omega_i : a \leq x \leq b\} \quad \forall \Omega_i \in \xi$
return S

function DETERMINE_DENSITY(h, a, b, x)
    return $\int_a^b \frac{1}{h(x)}\, dx$
end function