From Energy Absorption to Self-Organization : A Physics-Informed Machine Learning Approach to Laser-Matter Interaction

Fayad Ali Banna

N° d’ordre NNT : 2025UJMO0079

THÈSE de DOCTORAT
DE L’UNIVERSITÉ JEAN MONNET SAINT-ÉTIENNE

Membre de l’Université de Lyon

Ecole Doctorale N°488
SIS - Sciences Ingénierie Santé

Spécialité de doctorat : Informatique

Soutenue publiquement le 15 décembre 2025, par :

Fayad ALI BANNA

From Energy Absorption to Self-Organization: A Physics-Informed Machine Learning Approach to Laser-Matter Interaction

---

Devant le jury composé de :

Romain TAVENARD, Professeur des universités, Université Rennes 2	Rapporteur
Olivier PIERRE-LOUIS, Directeur de recherche, Université Claude Bernard Lyon 1	Rapporteur
Christine SOLNON, Professeure des universités, INSA de Lyon	Examinatrice
Rémi EMONET, Professeur des universités, Université Jean Monnet Saint-Etienne	Co-directeur de thèse
Jean-Philippe COLOMBIER, Professeur des universités, Université Jean Monnet Saint-Etienne	Co-directeur de thèse
Eduardo Brandão, Maître de conférences, Université Jean Monnet Saint-Etienne	Invité

Abstract

A repeated irradiation of a solid surface with ultrashort laser triggers self-organization phenomena resulting in a variety of patterns at the nanoscale level. The resulting surfaces present interesting properties with promising applications such as designing hydrophobic surface and reducing bacterial colonization on medical implants. However, the understanding of the laser-matter interaction that leading to self-organization is still an open area to work.

Therefore, this thesis aims at first to advance both the machine learning approaches for discovering the equations governing the self-organization in laser-matter interaction, and second, to advance the understanding the physical dynamics of laser-induced self-organization.

In order to model self-organization in laser-matter interactions we adopt Physics-Informed Machine Learning (PIML) techniques. PIML refers to a subfield of Machine Learning (ML) that combines the accuracy physics-based modelling and the generalization ability of data-driven models to improve the reliability of ML models. Although, existing PIML methods such as Physics-Informed Neural Networks (PINN), Neural Ordinary Differential Equations (NeuralODE) and Fourier Neural Operators (FNO) have successfully solved a variety of problems, they still have limitations. In practice, these methods rely on two ideal conditions that are hard to satisfy in real-world problems, especially in laser-induced self-organization. They require the knowledge of governing equations behind the studied phenomenon or a large amount of data. Although Maxwell's equations describe the interactions during each irradiation, the post-irradiation dynamics leading to the structures formation is still not understood. Consequently, there is no governing equations to explain the self-organization and the available data is limited to surface after the whole structures formation process since we cannot measure self-organization in real-time.

To address the limitation of existing ML methods and advance the understanding of self-organization in laser-matter interaction, this thesis proposes an end-to-end framework that maps the state of an irradiated surface, the number of irradiations and the laser parameters to the resulting self-organized structures. The first component is a fast, differentiable neural solver for modelling the interactions during the pulse characterized by the distribution of laser energy that initiate the self-organization. It emulates Maxwell's equations with high fidelity and drastically reduces computation time. The second component, named Unrolled-SINDy, extends the Sparse Identification of Non-linear Dynamics (SINDy) framework to recover the post-irradiation dynamics from measurements separated from a large temporal gap, where traditional PDE discovery methods fail. Unrelated to the end-to-end framework, a third contribution leverages the solver developed in the first contribution ability to accelerate large scale analysis to propose an explanation for self-organization. This contribution supports the hypothesis that self-organization behaves like a learning process to produce patterns that maximize energy absorption. Moreover, we identify the López-Ruiz–Mancini–Calbet (LMC) complexity measure as a strong predictor of laser energy absorption. This provides a quantitative link between self-organized patterns complexity and their response to laser energy deposit.

Résumé

L'irradiation répétée d'une surface solide avec un laser ultra-court déclenche des phénomènes d'auto-organisation qui donnent lieu à une variété de motifs à l'échelle du nanomètre. Les surfaces ainsi obtenues présentent des propriétés intéressantes qui ouvrent la voie à des applications prometteuses, telles que la conception de surfaces hydrophobes et la réduction de la colonisation bactérienne sur les implants médicaux. Cependant, la compréhension de l'interaction laser-matière qui conduit à l'auto-organisation reste un domaine à explorer.

Par conséquent, cette thèse vise d'abord à faire progresser les approches d'apprentissage automatique pour découvrir les équations régissant l'auto-organisation dans l'interaction laser-matière et ensuite à faire progresser la compréhension de la dynamique physique de l'auto-organisation induite par laser.

Afin de modéliser l'auto-organisation dans les interactions laser-matière, nous adoptons les techniques d'apprentissage automatique informé par la physique (PIML). Le PIML fait référence à un sous-domaine de l'apprentissage automatique (ML) qui combine la précision de la modélisation basée sur la physique et la capacité de généralisation des modèles basés sur les données afin d'améliorer la fiabilité des modèles ML. Bien que les méthodes PIML existantes, telles que les Physics-Informed Neural Networks (PINN), les Neural Ordinary Differential Equations (NeuralODE) et les Fourier Neural Operators (FNO), aient permis de résoudre avec succès divers problèmes, elles présentent encore des limites. Dans la pratique, ces méthodes reposent sur deux conditions idéales difficiles à satisfaire dans les problèmes du monde réel, en particulier dans le cas de l'auto-organisation induite par laser. Elles nécessitent la connaissance des équations régissant le phénomène étudié ou une grande quantité de données. Bien que les équations de Maxwell décrivent les interactions pendant chaque irradiation, la dynamique post-irradiation conduisant à la formation des structures n'est encore pas comprise. Par conséquent, il n'existe pas d'équations régissant l'auto-organisation et les données disponibles se limitent à la surface à la fin du processus de formation des structures car nous ne pouvons pas mesurer l'auto-organisation en temps réel.

Pour résoudre ce problème, cette thèse propose un modèle end-to-end qui permettrait connaissant l'état initial d'une surface irradiée, le nombre d'irradiations et les paramètres laser de determiner la forme des structures auto-organisées qui en résulteraient. Le premier composant est un solveur neuronal rapide et différentiable permettant de modéliser les interactions pendant l'impulsion caractérisées par la distribution de l'énergie laser qui déclenche l'auto-organisation. Il résoud les équations de Maxwell avec une grande fidélité et réduit considérablement le temps de calcul. Le deuxième composant, appelé Unrolled-SINDy, étend le modèle de découverte d'équations aux dérivées partielles (EDP) SINDy (Sparse Identification of Non-linear Dynamics) pour identifier la dynamique post-irradiation à partir de mesures séparées par un grand écart temporel, là où les méthodes traditionnelles de découverte des EDP échouent. Sans rapport direct avec le modèle end-to-end, une troisième contribution tire parti de la capacité du solveur développé dans la première contribution à accélérer l'analyse à grande échelle de données pour proposer une explication de

l'auto-organisation. Cette contribution soutient l'hypothèse selon laquelle l'auto-organisation se comporte comme un processus d'apprentissage pour produire des modèles qui maximisent l'absorption d'énergie. De plus, nous identifions la mesure de complexité López-Ruiz-Mancini-Calbet (LMC) comme un puissant prédicteur de l'absorption d'énergie laser. Cela fournit un lien quantifiable entre la complexité des modèles auto-organisés et leur réponse au dépôt d'énergie laser.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis revolves around Physics Informed Machine Learning (PIML) [60, 102, 54], a subfield of artificial intelligence that studies the integration of physical principles directly into Machine Learning (ML) models. By embedding physical laws into the learning process, PIML aims at achieving higher accuracy and produces interpretable models consistent with fundamental scientific principles.

PIML methods can generally be grouped into two categories: forward problems and inverse problems. The forward problems refer to the use of machine learning models guided by physical laws to predict the future behavior or state of a physical system based on known conditions [102, 27]. Within this category, PIML has been shown to be particularly effective in three main directions. One involves the construction of surrogate solvers, such as PINNs [102], PPINNs [86], or NeuralODEs [27], which are trained to approximate the input–output behavior of numerical solvers for a specific instance of a PDE or a fixed parameter configuration. Once trained, these surrogates deliver solutions orders of magnitude faster than classical methods, enabling large parameter sweeps, real-time prediction, and integration into optimization or design pipelines. A second important application is the correction of incomplete physical models using approaches like Hidden Physics Models [101] and Hybrid PINNs [81]. In many cases, PDEs capture only part of the dynamics, while other effects are unknown or neglected. These methods learn data-driven corrective terms that augment existing models, striking a balance between physical fidelity and flexibility. The third direction focuses on accelerating large-scale simulations, with models such as the FNO [69] or DeepONet [78]. Unlike surrogate solvers, these neural operators aim to learn a general solution operator capable of predicting PDE solutions across varying initial or boundary conditions, geometries, or parameter configurations, effectively bypassing costly numerical iterations for many instances of a problem. This broader generalization makes them particularly valuable for computationally intensive tasks such as uncertainty quantification, inverse design, or control. In contrast, inverse problems involve inferring the underlying physical laws or system parameters from observed data [20, 109]. Typical examples include discovering governing equations such as PDEs directly from measurements using SINDy [20] or PDE-FIND [109], and estimating unknown model parameters with methods like NeuralODEs [27].

Despite their promise, PIML methods have inherent limitations. They assume two favorable conditions: (i) the underlying physical laws are well known and can be fully written as PDEs, and (ii) large quantities of high-quality training data are available. However, many real-world scientific problems violate both assumptions. In particular, PDEs often capture only part of the dynamics while omitting important physical effects, and the available data is extremely scarce, sometimes limited to a few indirect measurements. These unfavorable conditions remain minimally explored in the PIML literature, yet they represent some of the most pressing challenges in

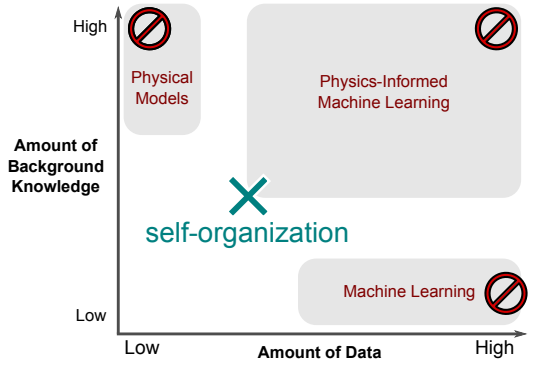


FIGURE 1.1: Illustration of self-organization specific challenge in PIML framework.

physics-based modeling. A particularly relevant and scientifically rich example is the self-organization of matter under ultrashort laser pulses, the target application of this thesis, where no explicit governing equations exist and experimental observations are extremely limited, making data-driven modeling particularly challenging, as illustrated in Fig. 1.1, which positions self-organization within the general PIML framework.

To investigate self-organized pattern formation in a controlled setting, we irradiate multiple spots on a metallic surface with a series of double laser pulses. Each spot receives a different number of pulses, for example, one may be exposed to N double pulses, another to N+1, and so on, with these irradiations widely spaced in time. Once the irradiation experiment is complete, each spot is imaged using Scanning Electron Microscopy (SEM) to reveal the resulting surface structures, providing a visual representation of the scanned surface patterns as shown in Fig. 1.2. This approach is necessary because current electronic imaging technologies cannot capture pattern formation in real time during laser exposure. In summary, we have detailed post-irradiation surface information for each spot, but the step-by-step evolution driven by each double pulse remains unknown. In this process, the first pulse rapidly heats and partially melts the surface, and before this process finishes, the second pulse arrives, repeating the interaction. After each sequence of pulses, the surface is allowed to cool before the next exposure, and Fig. 1.3 illustrates the laser-induced effects on the surface during these pulses. The system dynamics are governed by Maxwell's equations during the pulses, whereas in the intervals between pulses, the dominant processes remain poorly understood.

To address the complex and largely unexplored dynamics of laser-induced self-organization, we propose an end-to-end neural model capable of predicting the surface morphology resulting from sequences of ultrashort double laser pulses. By "end-to-end," we mean a single framework that directly maps the pulse parameters and the initial material state to the final surface structure, capturing both the ultrafast pulse interactions and the slower surface dynamics. Through this framework, the contributions of this thesis are two fold. The first one is a fast, differentiable neural solver for laser energy absorption, which efficiently simulates Maxwell's equations and provides accurate predictions of the material's response during each ultrafast pulse. This component is essential not only for computational efficiency, but also to model the interaction of the second pulse with a surface reshaped by the first pulse, whose state is unknown. The second contribution takes the form of a physics learning module designed to discover the post-irradiation dynamics from distant observations

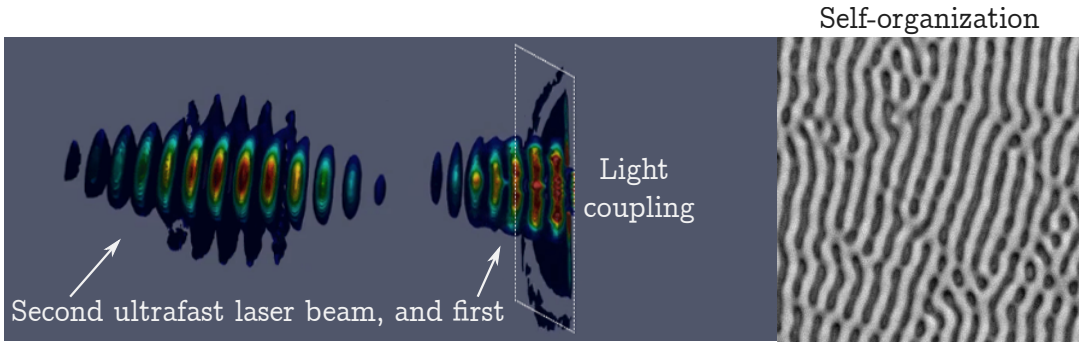


FIGURE 1.2: Self-organization process triggered by ultrafast laser-matter interaction. On the right: Emerging pattern after a series of double laser pulses.

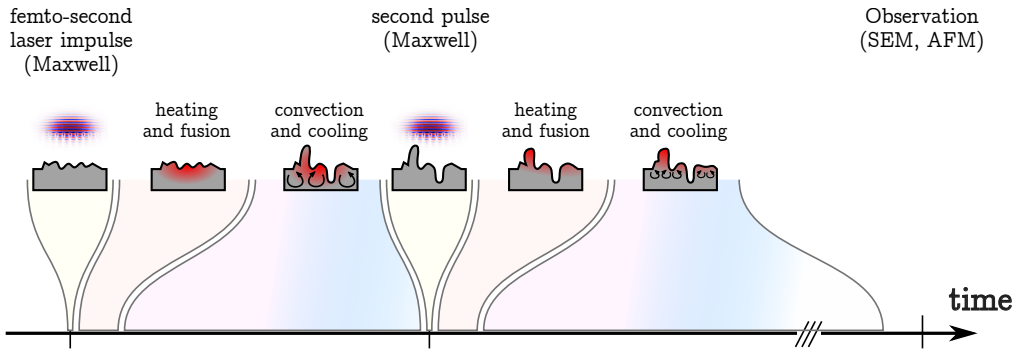


FIGURE 1.3: Illustration the intermediate surface dynamics including material heating, melting, convection, and laser-material interaction during a single double-pulse event.

in time. This module reconstructs the hidden processes that govern surface evolution between double pulse sequences, addressing the lack of explicit governing equations and the limited availability of experimental data. Finally, the third contribution, while separate from the end-to-end module, uses the fast, differentiable, and generalizable surrogate solver for laser energy absorption developed in the first contribution to demonstrate that self-organized surface patterns resulting from laser irradiation act as a form of structural memory leading to the emergence of structures that maximize energy absorption.

The following chapters are organized to provide a comprehensive presentation of the problem, the methodology, and the contributions of this work:

- **Chapter 2** provides a detailed review of the state of the art in Physics-Informed Machine Learning (PIML). It examines strategies for integrating physical knowledge into machine learning models, explores the various types of PIML frameworks, and discusses their strengths and limitations. This chapter also explains the phenomenon of self-organization in laser–matter interactions, outlining the underlying physical mechanisms and presenting key experimental approaches used to generate and study laser-induced self-organization.

- **Chapter 3** summarizes the development of a fast, differentiable neural solver for laser energy absorption, detailing its design, training, and evaluation. The chapter includes a comparison of commonly used PIML architectures for predicting

energy absorption, highlighting that a U-Net–based [104] model consistently outperforms alternative approaches. Beyond its role as a surrogate model, the solver is also employed for a preliminary investigation of the underlying physics of laser-induced self-organization.

- **Chapter 4** proposes an explanation for the formation of self-organized surface patterns, suggesting that these patterns emerge via a learning-like process. Using a series of carefully designed experiments, we demonstrate that the resulting structures correspond to the most stable configurations for maximizing laser energy absorption. This chapter also emphasizes the necessity of a fast surrogate solver, as it enables the processing of energy absorption across thousands of surface configurations. Additionally, we show that the statistical complexity measure known as LMC (López-Ruiz, Mancini, and Calbet) [77] serves as a highly predictive indicator of energy absorption, providing a valuable tool for analyzing pattern formation.

- **Chapter 5** introduces Unrolled-SINDy, a module based on the SINDy [20] framework for learning post-irradiation dynamics from observations recorded with large time gaps, highlighting its potential to enhance PDE discovery. The chapter compares the performance of closed-form solutions using Euler and RK4 integration methods, as well as gradient descent strategy. Furthermore, we demonstrate that the unrolling principle can be extended to iNeural-SINDy [48], an existing PDE discovery method that is robust to noise, showcasing its broader applicability.

- **Chapter 6** concludes the thesis by summarizing the main findings and discussing directions for future research.

The work achieved through this PhD thesis led to the following papers and presentations :

1. International conferences

    - Banna, F. A., Colombier, J. P., Emonet, R., & Sebban, M. (2024, October). Physics-informed Machine Learning for Better Understanding Laser-Matter Interaction. In 2024 IEEE 36th International Conference on Tools with Artificial Intelligence (ICTAI) (pp. 199-205). IEEE. The results presented in this paper are developed in the **chapter 3**.

2. National conferences

    - Banna, F. A., Rudenko A., Emonet, R., Sebban, M., Colombier, J. P.. Predicting laser energy absorption on nanostructured surfaces with deep learning. Machine Learning in Photonics, Apr 2024, Strasbourg, France. SPIE, pp.74, ⟨10.1117/12.3022317⟩. ⟨hal-04888157⟩
    - Banna, F. A., Colombier, J. P., Emonet, R., Sebban, M.. Apprentissage automatique informé par la physique : vers une meilleure compréhension de l'interaction laser-matière. CAp–RFIAP 2024, Jul 2024, Lille, France.

3. Under review

- Banna, F. A., Brandao, E., Nakhoul, A., Emonet, E., Sebban, & M., Colombier, J. P. (2025).Photonic Self-Learning in Ultrafast Laser-Induced Complexity. Ultrafast Science. **Chapter 4** develops the results described in this paper.
- Banna, F. A., Brandao, E., Caradot, A., Emonet, E., Sebban, M., & Colombier, J. P.. Unrolled-SINDy: A Stable Explicit Method for Non linear PDE Discovery from Sparsely Sampled Data. AISTATS 2026, International Conference on Artificial Intelligence and Statistics. Finally, the findings of this paper are detailed in **chapter 5**.

4. Other presentations

   - Banna, F. A., Brandao, E., Nakhoul, A., Emonet, E., Sebban, & M., Colombier, J. P..Predicting nanostructures formation with deep learning for physics. EUR Manutech Sleight SSE 11, Jan 2024, Saint-Etienne, France.
   - Banna, F. A., Brandao, E., Nakhoul, A., Emonet, E., Sebban, & M., Colombier, J. P..Learning Mechanism Behind Self-organization Induced By Ultrafast Laser. EUR Manutech Sleight SSE 13, Jan 2025, Saint-Etienne, France.
   - Banna, F. A., Brandao, E., Nakhoul, A., Emonet, E., Sebban, & M., Colombier, J. P. Multi-step SINDy pour la découverte d'EDP à partir d'observations très espacées dans le temps. GDR IASIS, Jun 2025, Paris, France.

Note: to improve readability of the manuscript, various language tools were used to assist the writing.

Chapter 2

Physics-Informed Machine Learning and Self-organization

The study of self-organization in laser-matter interaction is a challenging task. Before addressing these challenges, the purpose of this chapter is to provide a proper introduction to the two main concepts of this work: Physics-Informed Machine Learning and self-organization in laser-matter interaction.

2.1 Introduction

This chapter provides a detailed explanation of Physics-Informed Machine Learning (PIML) and self-organization in laser–matter interactions. It is organized around the following two parts.

The first part presents PIML and its motivation. We then focus on the different strategies for incorporating physical knowledge such as governing equations, symmetries, and conservation laws at different stages of machine learning models implementation through the data, the architecture, the loss formulation, the optimization process and the inference. It also discusses the main challenges of PIML, including the balance between data fidelity and physical consistency.

The second part focuses on self-organization, a process through which ordered patterns emerge spontaneously in a constrained system. We introduce self-organization in general and then consider its specific manifestation in laser–matter interaction. We discuss the theoretical backgrounds of self-organization, the mechanisms thought to govern pattern formation, and its practical applications in surface functionalization. Finally, we highlight the main challenges in modelling self-organization.

2.2 Physics-Informed Machine Learning (PIML)

Physics-Informed Machine Learning (PIML) [60, 102, 54] refers to a class of models that integrate physical laws and domain knowledge directly into machine learning architectures to improve accuracy, interpretability, and generalization. Instead of relying purely on data, PIML leverages governing equations, symmetries, and conservation laws as constraints or inductive biases during training. This hybrid approach allows models to remain consistent with known physics while still benefiting from the flexibility and scalability of modern machine learning techniques.

PIML emerged as a response to the limitations of traditional black-box models of ML, which require large datasets and lack physical interpretability [108], as well as traditional numerical methods, which can be computationally expensive and lack generalization ability [99]. These limitations are particularly problematic in scientific and engineering domains where accuracy, robustness, interpretability, and generalization

are critical [137]. PIML addresses these challenges by integrating physical principles such as conservation laws, symmetries [9], and governing equations [102] directly into the learning process. This integration guides the model toward physically plausible and reliable predictions even when data are scarce or noisy [102, 27]. At the same time, it enables the discovery of unknown governing equations underlying a given phenomenon [20, 109]. Compared to traditional numerical and purely physics-based methods, PIML offers greater flexibility and generalization capabilities along with reduced computational cost, while maintaining physical consistency.

2.2.1 Physical priors

In machine learning priors are any assumptions or constraint imposed on a model before training in order to guide it toward a specific type of solutions. PIML framework borrow this concept through physical priors. Physical priors are a type of prior derived from established physical knowledge such as governing equations, symmetry constraints, and intuitive physical reasoning. They guide the learning process and ensure that the model predictions are reliable by adhering to known physical laws. Depending on the type of knowledge incorporated, physical priors can range from strong constraints embedded directly in the model architecture to weaker forms incorporated as penalties in the loss function [102, 27].

Physical priors can be categorized into three complementary types. The first and strongest form involves governing equations, which provide precise mathematical descriptions of a system's behavior. The second type comprises symmetry constraints, which encode invariances under specific transformations and reduce the effective search space of solutions. The third and weakest form consists of intuitive physical knowledge, which represents empirical understanding that ensures physically plausible outcomes even in the absence of exact mathematical laws. Each of these types of physical priors contributes to the robustness, interpretability, and physical fidelity of PIML models in different ways and they are often combined to achieve optimal learning performance.

(i) Governing equations. Governing equations are precise mathematical descriptions that capture the fundamental behavior of physical systems through relationships between variables and their derivatives. These equations are typically expressed as ordinary differential equations (ODE) or partial differential equations (PDE) that represent conservation laws, transport phenomena, or field dynamics. For instance, the Navier Stokes equations describe fluid motion by relating velocity, pressure, and forces acting within the fluid, while Maxwell's equations govern electromagnetic fields. These governing equations serve as a rigorous foundation for modeling complex systems in science and engineering. In PIML, models such as Physics Informed Neural Networks (PINN) [102] incorporate governing equations directly into the training process by including the residuals of these equations as penalty terms in the loss function. This means the network not only learns from data but also minimizes violations of physical laws, enabling it to generate physically consistent and generalizable predictions even when data is scarce, noisy, or incomplete. Because governing equations represent exact mathematical formulations of fundamental physical laws, they constitute the strongest form of physical prior [60]. When strictly enforced, they provide a hard constraint that guides the model toward physically faithful solutions, ensuring high reliability and interpretability.

(ii) Symmetry constraint. Symmetry constraints are a weaker form of physical prior compared to governing equations. Symmetry constraints reflect inherent invariances in physical systems under certain transformations, such as rotations, translations, reflections, or permutations. These symmetries arise naturally from physical principles and often imply conservation laws or simplifications in system behavior [22]. By incorporating these constraints, machine learning models can reduce the search space of solutions and improve robustness and interpretability. For example, in molecular modeling, rotational and translational symmetries mean that the physical properties of a molecule should not depend on its orientation or position in space [111, 49]. Equivariant Neural Networks [29] are specifically designed to respect such symmetry constraints by ensuring that their outputs transform predictably when the inputs undergo these transformations. A well-known class of these models is Group Equivariant Convolutional Networks (G-CNN), which extend traditional convolutional neural networks [91] (CNN) by building equivariance to transformations from specific symmetry groups directly into their convolution operations. This leads to improved learning efficiency and better generalization across physically equivalent configurations, making them powerful tools in areas like materials science, particle physics, and computer vision involving structured data.

(iii) Intuitive physical knowledge. Among the different forms of physical priors, intuitive physical constraints are generally considered the weakest, as they rely on approximate or common-sense knowledge rather than exact mathematical formulations. Intuitive physical constraints encompass empirical, heuristic, or common-sense knowledge about the physical world that may not be fully described by explicit governing equations but remain essential for producing physically plausible results [36]. This includes both formal empirical rules and the type of intuitive physics that humans naturally use to reason about and predict physical behavior, such as knowing that objects fall to the ground when dropped or that objects continue to exist even when they are no longer visible. Such constraints can involve the positivity of quantities like density or concentration, boundedness of parameters, monotonic relationships, or approximate conservation principles valid within certain regimes. For example, in mechanical systems, the conservation of energy can be embedded as an intuitive constraint to ensure that the total energy remains constant over time. Hamiltonian Neural Networks [52] (HNN) implement this idea by designing architectures that intrinsically conserve energy throughout their dynamics, enabling physically meaningful predictions even with limited or noisy data. By integrating intuitive constraints, machine learning models gain improved stability, realism, and generalization, especially in complex real-world systems where exact equations are unknown or incomplete [140, 96].

2.2.2 Integration of physical knowledge

While governing equations, symmetry constraints, and intuitive physical knowledge can guide PIML models toward reliable solutions, an important challenge persist: how to integrate these physical priors into a model?Generally, developing a machine learning model involves several components: the dataset, the model architecture, the loss function, the optimization strategy, and the inference process [50]. Physical constraints can be incorporated at any of these levels to influence how the model learns and generalizes. Figure 2.1 provides a schematic overview of how physical priors can be embedded into a machine learning model.

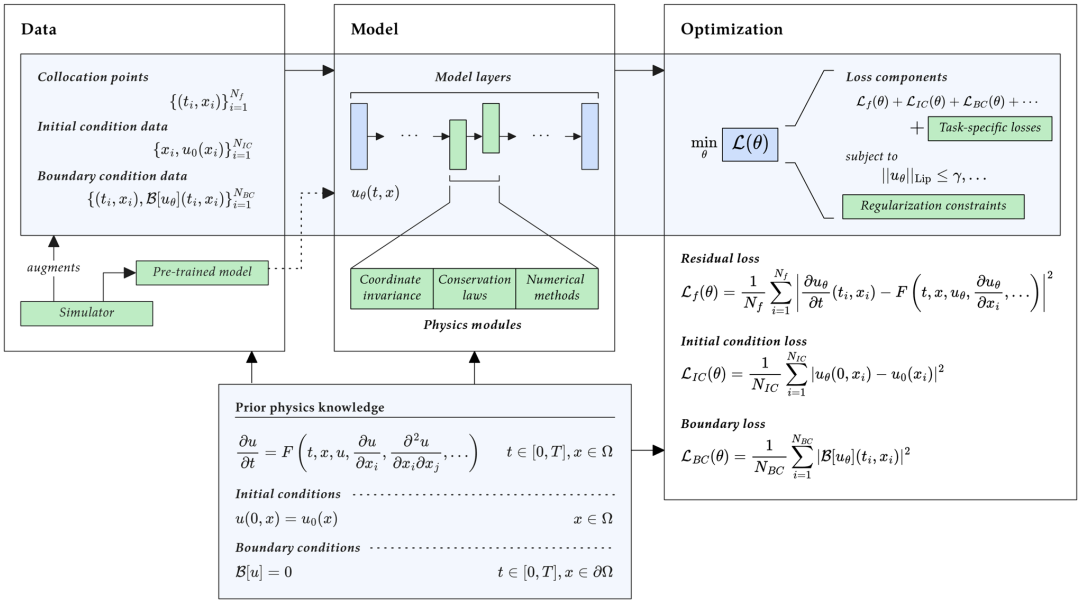


FIGURE 2.1: Schematic representation of how physical knowledge can be incorporated at different stages of a machine learning workflow. Adapted from [85].

(i) Incorporation at the data level. Physical priors can be introduced through the dataset itself. This can be achieved by augmenting the data with synthetic samples that are consistent with known physical laws [137]. By exposing the model to data that inherently respects these constraints, it learns representations that remain physically consistent with the dynamics behind the data. The data-driven integration is particularly useful when experimental data are available.

(ii) Incorporation in the model architecture. Within the model architecture, physical priors can be embedded directly by designing network structures that enforce physical behavior by construction [52, 102, 27]. Examples include neural networks that internally solve differential equations, architectures that preserve symmetries, or layers that ensure conservation of quantities such as energy or mass. Periodic or boundary conditions can also be integrated explicitly when the problem demands it [91]. By embedding these constraints structurally, the model is guaranteed to remain within the space of physically valid solutions throughout training.

(iii) Incorporation through the objective function. Physical knowledge can also be introduced through the loss function. In this approach, additional regularization terms are added to penalize violations of known physical laws [102]. The corresponding weights determine the balance between data fitting and physical consistency. This method provides flexibility by allowing partial adherence to physical constraints, which can be beneficial when the data contain noise or when the physical model is only approximately known.

(iv) Incorporation within the optimization process. During optimization, physical knowledge can be integrated by using algorithms that explicitly enforce physical constraints during training [57]. Instead of minimizing prediction error alone, these optimizers account for constraints such as conservation laws, stability requirements, or boundary constraints at each update step. This ensures that learning proceeds

within a physically meaningful solution in order to improve both convergence and interpretability.

(v) Incorporation at the inference stage. Finally, physical knowledge can also guide the inference stage. Post-processing techniques may be applied to adjust model predictions so that they comply with physical laws. This can include enforcing conservation principles, ensuring symmetry or boundary conditions, or removing physically impossible artifacts through smoothing or projection methods [3]. Incorporating physics at this stage provides an additional safeguard, especially in complex systems where exact compliance is difficult to achieve during training.

2.2.3 Tasks and applications of PIML

The integration of physical priors in machine learning also depends on the type of task being addressed. Broadly, PIML tasks can generally be grouped into two categories: neural simulation and inverse problems. Neural simulation or forward problem refers to the use of machine learning models guided by physical laws to predict the future behavior or state of a physical system based on known conditions [102, 27]. This includes tasks such as solving partial differential equations (PDE), estimating molecular properties, or forecasting surface patterns. On the other hand, inverse problems involve inferring the underlying physical laws or system parameters from observed data [20, 109]. Typical examples include discovering governing equations like PDE directly from measurements or optimizing control strategies for systems governed by physical dynamics.

Within the neural simulation task, PIML are particularly effective in three directions. First, it can be used to construct surrogate solvers, neural models trained to approximate the input–output behavior of numerical solvers while ignoring their main drawbacks[86, 95] (see Chapter 3). These surrogates deliver solutions faster than classical methods, allowing real-time prediction and integration into complex pipelines. Second, PIML can be applied to correct incomplete physical models [101]. In many cases, PDEs capture only part of the dynamics, while other effects are unknown or neglected. PIML methods can learn data-driven corrective terms that augment these models and balance between physical fidelity and flexibility. Third, PIML is increasingly used for accelerating large-scale simulations [69, 78] (see Chapter 4). Neural operators can directly approximate the solution operator of a PDE under varying initial or boundary conditions and bypass expensive numerical simulations. This proves particularly valuable in computationally intensive tasks such as uncertainty quantification, inverse design or control.

Inverse problems, in contrast, are addressed in Chapter 5, where the goal is to infer the underlying physical laws or parameters from observed data in order to discover governing equations.

2.2.4 PIML challenges

Despite these advances, PIML methods face several important limitations that hinder their broader applicability. One key challenge lies in scalability. PINN and related approaches often require significant computational resources due to repeated evaluations of PDE residuals and the use of automatic differentiation, making them difficult to apply to high-dimensional systems or large-scale simulations [115]. Moreover, training can be unstable, as balancing data-driven losses with physics-based constraints frequently leads to stiff optimization spaces and sensitivity to hyperparameter

choices, especially for problems involving sharp gradients, long-time integration, or chaotic dynamics [145, 68]. The effectiveness of PIML also depends strongly on the quality and completeness of physical models. When governing equations are approximate or contain unknown terms, predictions may degrade unless augmented with sufficient data [102]. In contrast, operator-learning approaches such as DeepONet [79] or Fourier Neural Operators [69] (FNO) alleviate some of these computational constraints but remain highly data-intensive, often requiring large and carefully curated datasets to generalize well [78]. Finally, PIML methods remain vulnerable to noise and sparsity in experimental data, which limits their direct application in real-world scientific and engineering problems where measurements are expensive or incomplete.

Given these limitations and the growing success of PIML across various scientific and engineering domains, we explore its potential to advance the understanding of laser–matter interaction. In particular, we focus on the phenomenon of self-organization, which plays a central role in the formation of complex surface structures under laser irradiation. The next section is dedicated to introducing this concept and discussing its relevance within the context of laser–matter interaction.

2.3 Self-organization

2.3.1 Definition of self-organization

Self-organization [1, 89] is the spontaneous emergence of ordered patterns, structures or behaviors in a system that initially lacks any apparent organization. Unlike processes driven by external control or templating, self-organization arises purely from internal interactions among the elements of the system. Through nonlinear feedback mechanisms, small local fluctuations are amplified and stabilized, giving rise to coherent global order. Nonlinear feedback means that the response of the system is not proportional to the input but instead, small perturbations can lead to disproportionately large effects. This relation between amplification and regulation leads to the system evolution toward an organized state without any external predefined plan.

This phenomenon is a hallmark of complex systems, where macroscopic order emerges from microscopic dynamics without centralized direction. It manifests across a wide range of natural and engineered systems, bridging physics, chemistry, biology, and even social sciences. In physical and chemical contexts, it can be seen in convection rolls, sand ripples, and oscillating chemical reactions [31, 135]. In biological systems, it governs the emergence of spatial organization in tissues, the synchronization of neuronal activity, and the formation of animal coat patterns [65]. Self-organization also appears in purely physical systems such as snowflake formation [4], where local molecular interactions give rise to globally ordered crystalline symmetry. Figure 2.2 provides an overview of self-organization patterns in nature.

2.3.2 Self-organization in laser-matter interaction

Theoritical fundations of self-organization

The concept of self-organization has profoundly influenced our understanding of complex systems across scientific disciplines. Ilya Prigogine, in thermodynamics, introduced the notion of dissipative structures [97, 89], which are ordered states maintained through continuous energy dissipation. He demonstrated that irreversible

(A) (B) (C)

FIGURE 2.2: Examples of self-organized patterns in nature: (a) zebra stripes, (b) sea shell patterns, and (c) sand dune ripples.

processes occurring far from thermodynamic equilibrium can spontaneously generate and sustain order, challenging the traditional view that disorder is the inevitable outcome of such processes. Small fluctuations within an open system can be amplified through energy flows to produce coherent macroscopic structures. It means that order can emerge naturally from the intrinsic dynamics of a system.

In addition, Alan Turing, in the context of morphogenesis [130], provided a rigorous mathematical framework to explain how spatial patterns can spontaneously emerge. Through his reaction–diffusion theory, Turing showed that the interplay of local activation and long-range inhibition can transform initially uniform conditions into stable, complex patterns without any external guidance. While Turing did not explicitly use the term "self-organization" his model demonstrated the fundamental principle that order can arise purely from internal system interactions.

Together, the contributions of Prigogine and Turing provide a definition of self-organization as a phenomenon through which complex, coherent structures arise from microscopic dynamics, independent of external control. Understanding the mechanisms of self-organization provides a powerful lens for interpreting a vast number of phenomena, from biological morphogenesis to the emergent patterns observed in laser–matter interactions.

Mechanism of self-organization in laser-matter interaction

In this work, we focus on the phenomenon of self-organization as it occurs in laser–matter interactions. Specifically, we study how ordered surface structures spontaneously emerge when a metallic surface is irradiated with linearly polarized ultrashort laser pulses [39, 103].

From the perspective of laser–matter interaction, self-organization typically occurs on surfaces that are open systems, meaning they exchange energy with their surroundings through the continuous input of laser light. The system is driven far from thermodynamic equilibrium, because the laser deposits energy at a rate that prevents the surface from relaxing back to its initial state. Under these conditions, dissipative processes such as melting, ablation, heat conduction, and material redeposition redistribute the absorbed energy [89], enabling the formation of stable patterns. Small fluctuations in the surface, such as tiny variations in roughness or local absorption, are amplified by nonlinear feedback mechanisms, where local increases in absorption lead to more material modification, which further enhances absorption in the same region. This interplay between energy input, dissipation, and feedback transforms initially disordered surfaces into self-organized structures such as ripples, grooves, or laser-induced periodic surface structures. In this way, disorder and fluctuations

act as the seeds of order, and coherent patterns emerge spontaneously without any external template or guidance.

At first glance, one might argue that the laser imposes external control since its parameters, such as fluence, polarization, pulse number or wavelength, influence the outcome. However, the distinction lies in the difference between providing energy and directly determining the formation of the structures. In laser–matter interaction, the laser acts as an external energy source but not as a structuring agent. A structuring agent is an influence that explicitly dictates the spatial arrangement of a system's components like a lithography mask that directly defines where material is deposited or removed. In contrast, the laser delivers energy to the surface without specifying any particular spatial pattern. Therefore, while laser parameters set the conditions that allow self-organization to occur, the pattern formation itself is self-organized. The structures are not directly imposed by the laser but emerge spontaneously through the nonlinear coupling of light, material properties and evolving surface morphology during consecutive laser irradiation.

Applications of self-organized structures

The study of self-organized patterns on laser-irradiated surfaces is motivated by theoretical interest and intellectual curiosity. These structures allow to study how complex and ordered structures can emerge spontaneously from initially disordered surfaces under well-controlled energy input. Understanding the mechanisms behind self-organization in laser–matter interaction offers insights into fundamental questions about pattern formation, nonlinear feedback and the interplay between energy deposition and material response at the micro- and nanoscale.

Beyond their conceptual significance, self-organized structures also present opportunities for practical applications. Laser texturing can modify surface wettability and morphology, a property that has been explored to reduce bacterial colonization [110] on dental and orthopedic implants. In photonics, laser-induced periodic surface structures on metallic substrates such as stainless steel can act as diffraction gratings, manipulating light efficiently [37, 139]. By carefully tuning the laser parameters, multiple orientations and overlapping structures can be generated, producing surfaces that display different structural colors or patterns depending on the viewing angle or illumination direction. This tunable optical response can be harnessed for advanced applications such as structural coloration [132], optical encryption, and anti-counterfeiting technologies.

2.3.3 Challenges in modeling self-organization

Self-organization, especially in the context of laser–matter interaction, remains an open and challenging topic. There is currently no exact partial differential equation (PDE) that fully describes the formation of these surface structures. Researchers have proposed phenomenological models, such as the Swift-Hohenberg equation to capture the mechanism of pattern formation including characteristic wavelengths and orientation. Other approaches involve coupled simulations of the electromagnetic field and surface evolution which remain computationally expensive.

In practice, the concept of self-organization is hard to define, so identifying it in real systems often involves some degree of subjectivity. There is no universally accepted criterion that clearly distinguishes a self-organizing process from an externally controlled one. In general, a system is considered self-organizing when its structure emerges spontaneously from internal interactions rather than being directly

imposed by external control. However, this distinction can be subtle because external parameters such as laser fluence, polarization or pulse number define the conditions that allow internal feedback mechanisms to operate. It is difficult to separate the contribution of external energy input from that of internal organization. For this reason, discussions continue about what qualifies as authentic self-organization, especially in laser–matter interaction where both external driving and internal feedback are essential.

In addition, obtaining meaningful data is difficult. The dynamics evolve at extremely small spatial and temporal scales. The characteristic pattern sizes are often smaller than one-tenth of the laser wavelength, placing them well below the optical diffraction limit. Meanwhile, the relevant timescales (from 10^{-13} s to 10^{-12} s) are far shorter than the response time of current electronic acquisition devices ($\sim 10^{-9}$ s). As a result, direct observation of the underlying dynamics is impossible. Instead, the only accessible data comes from measurements taken after the laser–matter interaction by scanning electron microscopy (SEM) images of the surface. The resulting datasets are not only small but also represent the cumulative effect of many hidden processes making them extremely difficult to interpret.

To move from qualitative interpretation toward quantitative characterization, several approaches have been developed to measure the degree of self-organization. One promising direction relies on information theory and complexity analysis. From this perspective, self-organization corresponds to an increase in correlations within the system or a decrease in overall entropy, meaning that the system becomes more structured and less random over time. Entropy-based quantities such as Shannon entropy, transfer entropy, or statistical complexity can be used to quantify the amount of order that has emerged compared to the initial state. In physical systems, a reduction in configurational entropy together with the appearance of stable spatial or temporal structures is a clear indicator of self-organization. Complexity measures also capture the balance between order and disorder: too little interaction leads to randomness, while too much constraint leads to trivial uniformity. The emergence of intermediate, structured complexity, as observed for example in laser-induced periodic surface structures, represents the signature of self-organization where local fluctuations and global order coexist through nonlinear feedback.

Despite these limitations, the agreement between theoretical models, simulations, and experimental observations supports the view that the formation of laser-induced surface patterns is a manifestation of self-organization. These structures emerge from the intrinsic coupling between energy deposition, material dissipative properties, surface fluctuations and nonlinear feedback mechanisms. Rather than being influenced by external forces, the resulting order arises spontaneously through the system's internal dynamics, reflecting the self-organizing nature of the interaction between light and matter.

2.4 Conclusion

This chapter explored the concepts of Physics-Informed Machine Learning (PIML) and self-organization. We first presented the principles of PIML, explained the use of physical priors such as governing equations, symmetries and intuitive physical constraints and discussed the different strategies for embedding them at multiple stages of model development. Main tasks such as neural simulation and inverse problems were also outlined, along with the challenges associated with noisy, incomplete data and balancing physical fidelity with data-driven flexibility.

We then examined self-organization as a central mechanism leading the spontaneous formation of ordered structures in laser-irradiated surfaces. The chapter covered the theoretical foundations, the mechanisms driving pattern formation, practical applications, and the difficulties in collecting experimental data and modeling these systems. By highlighting the interplay between energy input, dissipation, fluctuations and nonlinear feedback, we illustrated why self-organization remains both a fascinating and challenging phenomenon to study.

Chapter 3

Neural Surrogate model for Maxwell's Equations

Understanding and predicting complex physical phenomena often requires solving partial differential equations, which can be prohibitively expensive for nonlinear, multi-physics, or nanoscale systems. This is especially true for the self-organization of matter under ultrashort laser pulses, where energy deposition and surface pattern formation are governed by Maxwell's equations in inhomogeneous, lossy media. In this chapter, we introduce a fast, differentiable neural solver for energy absorption that serves as an efficient surrogate for Maxwell's equations. This approach enables rapid exploration of the interplay between evolving surface morphologies and light-matter interaction and provides a foundation for inverse design, end-to-end learning of surface dynamics, and large-scale parametric studies.

3.1 Introduction

In the real world, physical phenomena are often governed by complex systems described by partial differential equations (PDEs). These equations encapsulate fundamental physical laws, ranging from fluid dynamics to electromagnetism, and constitute the cornerstone of modern science and engineering. Solving PDEs allows researchers to predict, optimize, and control natural processes with applications across physics, chemistry, biology, and materials science.

Despite their central role, efficient numerical solution of PDE remains challenging. Classical numerical methods, such as finite differences, finite elements, or spectral schemes, typically require substantial computational resources to resolve fine-scale dynamics. When PDEs are nonlinear, involve multiple coupled fields or must be solved over high-dimensional parameter spaces, the computational cost can become prohibitive. This is particularly problematic in settings where repeated simulations are required, such as design optimization, inverse problems, or data-driven discovery. Moreover, many scientific applications increasingly demand differentiability of the solver so that predictions can be integrated with optimization and learning pipelines, something that traditional numerical solvers are not naturally designed for.

To address this issue, Physics-informed machine learning (PIML) [54, 60] has recently emerged as a compelling strategy for crafting efficient surrogate solvers, augmenting physical laws with data-driven models or more generally constraining physics-aware models [75]. Among the various purposes of PIML methods, a particularly active direction is the development of neural architectures that can efficiently approximate the solution of complex PDEs, effectively serving as rapid solvers for computationally demanding problems. These solvers are sought to be mesh-free and adaptable to varying initial conditions and irregular boundary conditions. Rapid

solvers are particularly valuable because they allow for large-scale parameter sweeps, real-time predictions, and integration into design or optimization pipelines, which would be infeasible with traditional methods. This trajectory has led to many algorithmic contributions that have allowed to build suitable physics-regularized loss functions, guide the initialization of deep neural networks with consistent parameters or design new hybrid neural architectures at the same time data-driven and physics-driven. This new line of research led to a novel generation of deep-learning frameworks aiming to predict either the solution of a PDE or the dynamics itself (*e.g.*, PINN [102], FNO [69], PINO [70], UNO [100], PDE-Net [76], to cite a few).

PIML methods have great potential but often rely on ideal conditions that are rarely met in practice, such as complete knowledge of governing PDEs and abundant high-quality data. In systems like laser-induced self-organization, these assumptions fail because the dynamics are partially understood, experimental observations are sparse and indirect, and the processes occur over very different sizes and times with complex interactions that are highly sensitive to small changes. This combination of incomplete physics and limited data makes modeling self-organization particularly challenging.

Self-organization refers to the spontaneous emergence of ordered patterns from initially disordered states. It is a fundamental principle of complexity science and occurs across a wide range of natural and engineered systems, from biological organisms to geophysical formations. In the context of surface engineering, self-organization arises when a femtosecond or picosecond laser pulse interacts with a solid surface, a process sometimes referred to as light coupling. In this interaction, the laser deposits energy into the material, inducing localized heating, partial melting, and transient flow of the molten layer. Remarkably, instead of producing random surface modifications, these processes frequently result in highly regular, periodic micro- and nanostructures, collectively known as laser-induced surface pattern.

Modeling self-organization is particularly challenging because it involves the coupled interaction of multiple physical processes. Electromagnetic effects, thermodynamics, and fluid mechanics simultaneously govern light absorption, heating and melting, and molten material flow, yet no single set of partial differential equations captures the full dynamics. Data collection adds another layer of difficulty: the relevant spatial scales are often below one-tenth of the laser wavelength, and timescales are far shorter than the response time of current electronic acquisition devices. Consequently, direct observation of the dynamics is impossible, and the only available data are post-interaction measurements, such as SEM images. These datasets are small and reflect the cumulative effects of multiple hidden processes, making them extremely challenging to interpret.

To address this challenging problem, we focus specifically on modeling the electromagnetic phenomena that occur during the laser pulse itself. Although the experimental procedure was detailed in a previous chapter, we briefly recall it here to motivate the need for a new computational approach. In our setup, multiple spots on a metallic surface are irradiated with a sequence of double laser pulses. Each double pulse consists of a first pulse that rapidly heats and partially melts the surface, followed almost immediately by a second pulse that interacts with the modified surface. After every sequence, the surface is allowed to cool before the next exposure. While this setup provides detailed post-irradiation imaging of the resulting surface structures, the intermediate dynamics during each double pulse sequence remain entirely unobserved, particularly the surface state at the moment the second pulse arrives. To reconstruct and predict the full interaction process with an end-to-end model, the solver that captures the dynamics during the pulse must

be both computationally efficient and capable of strong generalization across a wide range of surface patterns, including those that are never directly observed. For this reason, we propose a fast, differentiable neural solver that approximates Maxwell's equations for predicting laser energy absorption and is capable of strong generalization across a wide variety of surface patterns. The PIML approach leverages the efficiency and adaptability of deep learning while incorporating physical constraints, making it particularly well-suited for modeling the intricate energy distributions generated by nanoscale surface features. Accurately simulating this process requires solving Maxwell's equations in inhomogeneous, lossy materials at the nanoscale, which is computationally intensive with conventional solvers and severely restricts exploration of the parameter space.

Addressing this difficulty resulted in three key contributions:

- We developed a fast, differentiable and generalizable surrogate for absorption prediction. By formulating the problem appropriately, we overcome data scarcity and efficiently train a simple UNet–based model. This solver provides an estimation of energy absorption across a wide variety of surface patterns, making it compatible with modern optimization and learning frameworks. Compared to traditional solvers such as finite-difference time-domain methods, our approach dramatically reduces computational cost while maintaining high accuracy.
- We conducted a comparative assessment of several widely used PIML architectures to evaluate their performance on energy absorption prediction. Our comparison shows that a UNet–based model consistently delivers superior performance in predicting laser energy absorption compared to other approaches.
- We explored the use of the surrogate model to investigate the physics of self-organization. Beyond providing a fast solver, we illustrate how it can be used to assess whether predicted surface patterns are physically coherent and consistent with expected energy absorption behavior. This aspect will be examined in greater detail in the next chapter.

3.2 Related works : MaxwellNet

An attempt to develop a surrogate for solving Maxwell's equations using neural networks guided by physics was proposed in [71]. This approach, called MaxwellNet, combines two elements during training. It embeds the residual of Maxwell's equations into the loss function, ensuring that the network predictions remain consistent with physical laws. At the same time, it makes use of supervised data obtained from conventional electromagnetic solvers, which provide reference solutions that stabilize and improve the training process. This hybrid design highlights both the potential and the limitations of such methods, particularly their reliance on large datasets generated from computationally expensive simulations.

MaxwellNet is designed to model the propagation of a stationary light wave through a transparent medium in two dimensions. By reducing the problem to two spatial dimensions, the approach simplifies the simulation but excludes physical effects that are relevant in fully three dimensional settings. In contrast, our work addresses the interaction of ultrashort laser pulses with metals in three dimensions. Metals are lossy media, meaning that they absorb energy and generate currents due to their finite electrical conductivity. Capturing these interactions requires accounting

for both spatial and temporal variations of the laser pulse as well as the response of free electrons, all of which are crucial for predicting realistic energy deposition.

A further difference lies in the prediction target. MaxwellNet focuses on reconstructing the electric field distribution inside the material. While fundamental, the electric field is not a direct measure of the absorbed energy that drives thermal effects and self organization processes. Our approach instead predicts the absorbed energy itself. This bypasses the intermediate step of computing the full electromagnetic field and provides faster, more interpretable, and physically meaningful predictions that directly connect to material modification.

The computational strategy also differs. MaxwellNet depends on extensive full wave simulations to provide both supervised data and reference solutions for training, which makes dataset generation expensive. Our formulation, by contrast, enables efficient learning even with smaller datasets. Expressing the problem in terms of absorption allows us to exploit inherent physical regularities, and a relatively simple UNet architecture is sufficient to achieve accurate predictions. This design reduces both training cost and inference time, making it possible to evaluate thousands of surface morphologies efficiently.

Finally, our model is tailored to investigate the coupling between surface morphology and absorption in repeated laser irradiation. Self organized structures emerge because absorbed energy from one pulse modifies the surface, which then influences how subsequent pulses are absorbed. By serving as a differentiable and efficient surrogate for absorption, our model enables systematic exploration of this feedback loop and provides insights into how statistical measures of surface complexity such as roughness, periodicity, and irregularity correlate with absorption efficiency.

In summary, MaxwellNet illustrates the use of neural networks that incorporate both physics based constraints and supervised data to model electromagnetic problems. Our work differs by addressing three dimensional interactions with metals under ultrashort laser pulses, by predicting energy absorption rather than reconstructing electric fields, and by enabling large scale investigations of feedback driven self organization.

3.3 Preminary background

In this section, we recall Maxwell's equations and the Finite-Difference Time Domain method as a numerical technique for solving them. While prior knowledge of these concepts is not required to understand the later parts of this thesis, reviewing them provides a clearer understanding of two aspects: (i) the physical phenomena addressed in this study and (ii) the origin of the physics-informed simulation data that serve as training input for our machine learning models.

3.3.1 Creation of self-organized surfaces

This study situates itself within the broader context of nanoscale self-organization, focusing on energy absorption on metallic surfaces representative of those used in physical laser–matter interaction experiments. Producing these self-organized surfaces requires a multi-step process that demands precision and careful control.

The first step is surface preparation, which is critical for reproducible laser processing. Metal substrates are ultra-finely polished to minimize their initial roughness, thereby reducing micro-roughness and contamination that could otherwise distort the subsequent irradiation process.

To investigate self-organized pattern formation under well-controlled conditions, we irradiate a metallic surface at multiple, spatially separated spots using a series of cross-polarized double femtosecond laser pulses. Each spot receives a different number of pulse pairs; for instance, one spot may be exposed to N double pulses, another to $N+1$, and so on, while successive irradiations are separated by sufficient time to allow complete cooling of the material. This experimental design enables to progressively observe the formation of self-organized patterns while maintaining controlled and independent surface evolutions.

In the cross-polarized configuration, two laser pulses with perpendicular polarization states interact with the surface, producing an almost isotropic energy deposition. By precisely adjusting the inter-pulse delay within the picosecond range, different self-organization regimes can be accessed, ultimately determining the morphology of the emerging nanostructures. By varying the laser fluence F, inter-pulse delay Δt, and double pulses number N, four distinct experimental series are generated, each illustrating a different pathway of self-organized pattern formation, as shown in Figure 3.1. Series 1 corresponds to $F = 0.18\ \mathrm{J/cm^2}$ and $\Delta t = 8$ ps; Series 2 to $F = 0.21\ \mathrm{J/cm^2}$ and $\Delta t = 2$ ps; Series 3 to $F = 0.25\ \mathrm{J/cm^2}$ and $\Delta t = 36$ ps; and Series 4 to $F = 0.26\ \mathrm{J/cm^2}$ and $\Delta t = 28$ ps.

A detailed description of the experimental setup and the four series of irradiations is provided in Chapter 4. In this chapter, we limit the discussion to the information necessary to understand the implementation of the neural solver used as a surrogate for the numerical solution of Maxwell's equations for energy absorption.

In the present chapter, our focus is on modeling the energy deposition on the self-organized surfaces during each laser pulse. The energy absorption results from the interaction of the surface with the laser field and is described by Maxwell's equations. Being able to predict this energy deposition quickly is essential for building a framework capable of forecasting the formation of self-organized surface patterns.

3.3.2 Surface topography

The self-organized patterns shown in Fig. 3.1 were characterized using two complementary techniques: scanning electron microscopy (SEM) and atomic force microscopy (AFM). SEM provides high-resolution imaging by detecting secondary electrons emitted from the surface under electron-beam irradiation. It is particularly effective for revealing lateral morphological details across large areas but primarily delivers two-dimensional contrast, without direct access to absolute height information.

In contrast, AFM measures the surface topography directly by scanning a nanoscale probe over the material. This technique yields true three-dimensional maps of the surface with sub-nanometer vertical resolution, making it particularly valuable for quantifying the height, depth, and curvature of nanoscale features. However, AFM measurements are inherently slower, more expensive, and restricted to smaller scan areas compared to SEM.

In this study, SEM imaging was systematically performed on all irradiated samples, while AFM analysis was applied to a representative subset. The quantitative height data obtained from AFM served to calibrate and interpret the SEM results, enabling an approximate reconstruction of three-dimensional morphologies from the more widely available SEM images.

For this reconstruction, grayscale intensities in the SEM images were converted into approximate depth information: the brightest pixels (intensity = 255) were mapped to the highest points (+1), and the darkest pixels (intensity = 0) to the lowest

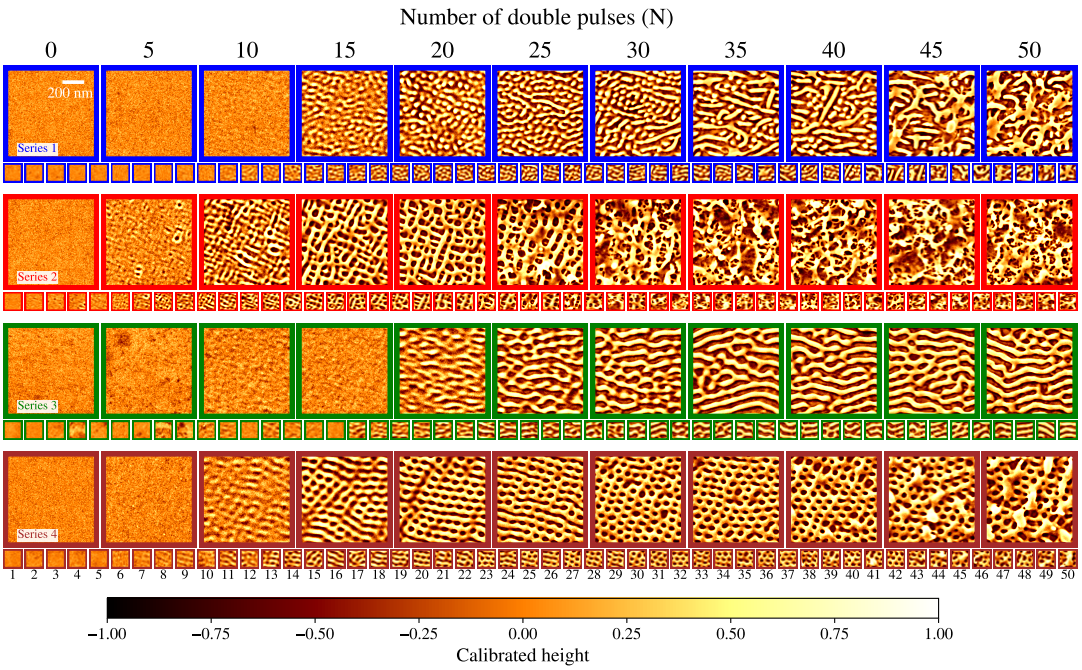


FIGURE 3.1: **SEM images of nickel surfaces irradiated with cross-polarized double pulses** N**.** Each series represents a progressive formation of self-organized patterns for distinct combinations of F and Δt: Series 1 corresponds to $F = 0.18$ J/cm² and $\Delta t = 8$ ps, Series 2 to $F = 0.21$ J/cm² and $\Delta t = 2$ ps, Series 3 to $F = 0.25$ J/cm² and $\Delta t = 36$ ps, and Series 4 to $F = 0.26$ J/cm² and $\Delta t = 28$ ps.

valleys (–1). Although this method remains qualitative, AFM-based calibration provides a reliable reference, allowing the transformation of two-dimensional electron micrographs into physically meaningful three-dimensional surface representations.

The experimental surfaces used in this work comprises four series of scanning electron microscopy (SEM) images, each illustrating the progressive formation of self-organized nanostructures under distinct laser irradiation conditions. As shown in Fig. 3.1, each series contains 51 grayscale images, corresponding to 50 irradiated spots and one non-irradiated reference region. The irradiated spots were exposed to an increasing number of cross-polarized double femtosecond pulses, ranging from 1 to 50, thereby capturing the continuous evolution of surface morphology as a function of the irradiation history. The four series differ by their combinations of laser fluence F and inter-pulse delay Δt, while all other experimental parameters are held constant. By systematically varying these two key parameters, the dataset encompasses a wide range of self-organization regimes.

It is important to note that these SEM images alone do not constitute the training set for our models. To train neural architectures in a supervised fashion, paired input–output data are required. In this work, the SEM images serve as the structural input representations of the irradiated surfaces, while the corresponding energy absorption maps are obtained through numerical simulations of laser–surface interaction based on Maxwell's equations. These simulations provide the ground-truth energy deposition patterns associated with each morphology, thereby enabling the construction of a physically informed dataset for training and validating the neural surrogate models.

From a purely computational standpoint, the categorization of SEM images into separate series is not strictly required for training a neural model to predict energy

absorption. In principle, one could pool all images together and train a network to learn the mapping from surface morphology to energy deposition, irrespective of the irradiation parameters. However, we deliberately preserve the series structure in our dataset to enable a more rigorous evaluation of the model's generalization capabilities. By training on a subset of series and testing on previously unseen ones, we assess the network's ability to predict energy absorption on novel surface morphologies. This strategy mirrors the experimental scenario in which the second pulse of a double-pulse sequence interacts with a surface pattern that has not been directly observed, thereby aligning the surrogate model's learning objective with the physical process of self-organization itself.

3.3.3 Maxwell's equation

Maxwell's equations are used in this work to describe how an incident femtosecond laser pulse interacts with the complex topography of self-organized nickel surfaces, governing the spatial distribution of electromagnetic energy and absorption within the material. These equations are solved in their full time-dependent form to capture the ultrafast evolution of the fields during laser irradiation.

Maxwell's equations [128] are the fundamental laws governing the behavior of electric and magnetic fields in space and time. They unify classical electromagnetism by relating electric fields, magnetic fields, and their sources, and they form the basis for understanding the interaction of light with matter. In differential form, they can be expressed as

$$\nabla \times \vec{E} = \frac{\partial \vec{B}}{\partial t}, \qquad \nabla \times \vec{H} = \frac{\partial \vec{D}}{\partial t}, \tag{3.1}$$

where $\vec{E}(x,y,z,t)$ and $\vec{H}(x,y,z,t)$ denote the electric and magnetic field vectors, $\vec{B} = \mu_0 \vec{H}$ is the magnetic flux density, and μ_0 is the vacuum permeability. The medium response is described by the displacement field $\vec{D}$, which captures how materials polarize under an applied electric field. In the frequency domain, this relation is given by $\tilde{\vec{D}} = \epsilon_0 \epsilon \tilde{\vec{E}}$, where $\tilde{}$ indicates the Fourier transform in time, ϵ_0 is the vacuum permittivity, and ϵ is the relative permittivity of the medium. For vacuum, $\epsilon = 1$, while for nickel we use $\epsilon = \tilde{n}^2 = (2.15 + 4.3i)^2$, with $\tilde{n}$ being the complex refractive index that accounts for both dispersion and absorption.

In Cartesian coordinates, the del operator and the electric field vector can be written as

$$\nabla = \vec{u_x}\frac{\partial}{\partial x} + \vec{u_y}\frac{\partial}{\partial y} + \vec{u_z}\frac{\partial}{\partial z}, \tag{3.2}$$

$$\vec{E} = E_x\,\vec{u_x} + E_y\,\vec{u_y} + E_z\,\vec{u_z}. \tag{3.3}$$

The curl of the electric field, $\nabla \times \vec{E}$, is a vector (cross) product that produces another vector whose components describe the local rotation of $\vec{E}$. Its precise component-wise definition in Cartesian coordinates is

$$\nabla \times \vec{E} = \left(\frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z}\right)\vec{u_x} + \left(\frac{\partial E_x}{\partial z} - \frac{\partial E_z}{\partial x}\right)\vec{u_y} + \left(\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y}\right)\vec{u_z}. \tag{3.4}$$

Each component of the curl vector corresponds to the rotation of the field around one of the Cartesian axes. In the above expressions, the operator $\times$ denotes the vector (cross) product. While the equations are written for the electric field $\vec{E}$ for clarity, the

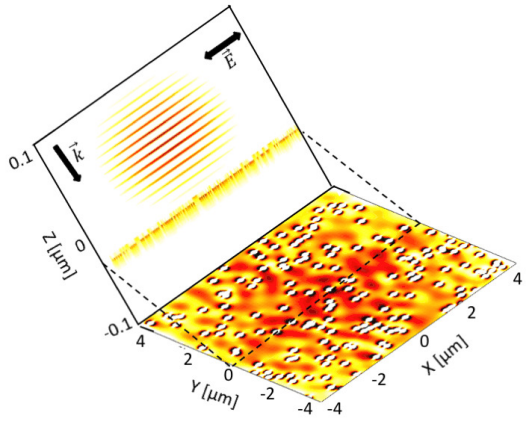


FIGURE 3.2: Illustration of the electric field distribution at the material surface, showing structuring on a scale comparable to the laser wavelength. The arrow above $\vec{E}$ indicates the polarization direction of the electric field, which governs the orientation of the low–spatial–frequency laser-induced periodic surface structures.

same expressions apply to the magnetic field $\vec{H}$. This component-wise form of the curl is directly used in numerical schemes such as the finite-difference time-domain (FDTD) method to compute the time evolution of the electromagnetic fields.

For the scenario considered in this study, a linearly polarized laser pulse propagates along the z-axis. The initial electric field $\vec{E}^0$ can then be written as

$$\vec{E}^0 = E_x^0 \sin(\alpha)\vec{u_x} + E_y^0 \cos(\alpha)\vec{u_y}, \tag{3.5}$$

where E_x^0 and E_y^0 are the amplitudes along the x and y directions, and α is the polarization angle measured in the xy-plane from the y-axis toward the x-axis. In this context, polarization refers to the orientation of the electric field vector in the plane perpendicular to the direction of propagation. When $\alpha = 0$, the field oscillates purely along the y-axis, while for $\alpha = \pi/2$, it oscillates along the x-axis. Intermediate values of α correspond to oblique orientations, describing how the direction of the electric field, and therefore the direction of energy deposition on the surface, rotates within the plane (see Fig. 3.2 for an illustration of the electric field distribution).

3.3.4 Finite-Difference Time-Domain (FDTD) method for electromagnetic simulations

Maxwell's equations are solved numerically using the Finite-Difference Time-Domain (FDTD) method, which provides a fully numerical solution by discretizing both space and time. In this approach, spatial derivatives represented by the curl operators are approximated using central finite differences on a Cartesian grid, while time derivatives are treated in the same way. This transforms Maxwell's equations into a set of update equations for the electric and magnetic field components at discrete grid points and time steps. The FDTD method enables the simulation of electromagnetic wave propagation and interaction with complex material surfaces, capturing reflection, refraction, and absorption phenomena with subwavelength spatial resolution.

A key feature of FDTD is the Yee grid, which staggers the electric and magnetic field components in both space and time. Each component of the electric field $\vec{E}$

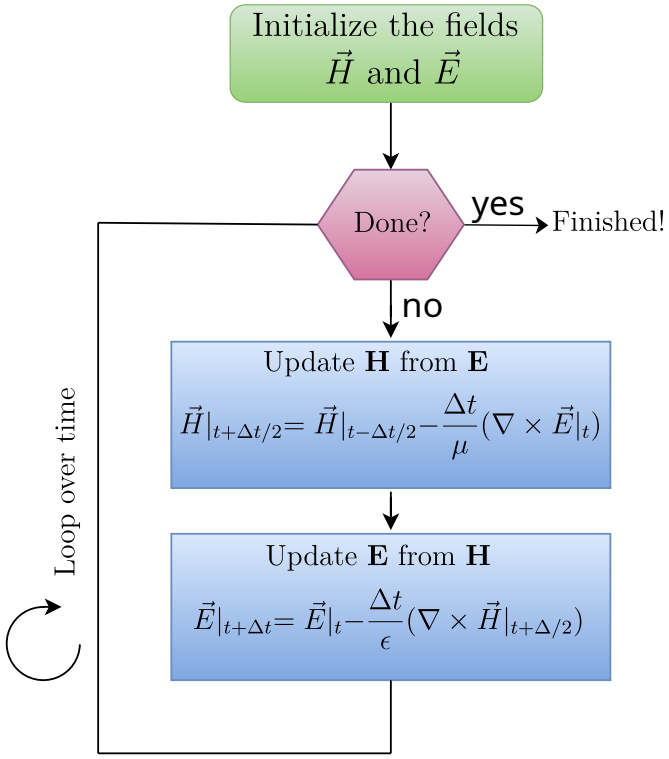


FIGURE 3.3: Illustration of the resolution of Maxwell's equations using the Finite-Difference Time-Domain (FDTD) method. The electric field components (E) are staggered in space relative to the magnetic field components (H) and are updated alternately in time.

is defined at the edges of a grid cell, while each component of the magnetic field $\vec{H}$ is defined at the center of the corresponding faces. This arrangement allows the curl operations in Maxwell's equations to be naturally approximated using finite differences and forms the basis of the leapfrog time-stepping scheme, where updates of $\vec{E}$ and $\vec{H}$ alternate in time.

The leapfrog update scheme alternates the computation of $\vec{E}$ and $\vec{H}$ fields in time. For example, the x-components are updated as:

Electric field (x-component):

$$E_x^{n+1/2}(i,j,k) = E_x^{n-1/2}(i,j,k) + \frac{\Delta t}{\epsilon}\left[\frac{H_z^n(i,j,k) - H_z^n(i,j-1,k)}{\Delta y} - \frac{H_y^n(i,j,k) - H_y^n(i,j,k-1)}{\Delta z}\right] \tag{3.6}$$

Magnetic field (x-component):

$$H_x^{n+1}(i,j,k) = H_x^n(i,j,k) - \frac{\Delta t}{\mu}\left[\frac{E_z^{n+1/2}(i,j+1,k) - E_z^{n+1/2}(i,j,k)}{\Delta y} - \frac{E_y^{n+1/2}(i,j,k+1) - E_y^{n+1/2}(i,j,k)}{\Delta z}\right] \tag{3.7}$$

Updates for the remaining components (E_y, E_z and H_y, H_z) follow similarly with cyclic permutations of indices and derivative directions. The electric field is updated

at half-integer time steps using the magnetic field at integer steps, and the magnetic field is then updated at integer steps using the newly computed electric field. This interleaving naturally propagates the fields forward in time (see Fig. 3.3 for an illustration).

To maintain numerical stability, the time step Δt must satisfy the Courant-Friedrichs-Lewy (CFL) condition. This condition arises from analyzing the propagation of a numerical wave in the discrete grid: for the discretized scheme to accurately capture wave propagation without artificial amplification, the wave must not travel more than one grid cell per time step. For a 3D Cartesian grid, this gives:

$$\Delta t < \frac{1}{c\sqrt{\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} + \frac{1}{\Delta z^2}}},$$

where c is the speed of light in vacuum, and $\Delta x, \Delta y, \Delta z$ are the spatial grid spacings. Exceeding this limit leads to unphysical growth of the fields and numerical instability.

To summarize, we employ the Finite-Difference Time-Domain (FDTD) method to investigate the spatially resolved energy deposition beneath self-organized nanostructured surfaces in fully three-dimensional simulations. The surface topography is represented as a 3D layer of cavities and bumps reconstructed from SEM images, as illustrated in Fig. 3.4. In this 3D computational domain, the presence of material is represented by a value of 1, while vacuum regions are represented by 0. A soft electromagnetic source placed above this layer in air launches a linearly polarized wave with its electric field oriented along either the x or y direction. The wave propagates downward, interacting with the detailed 3D surface morphology, while the full 3D distributions of the electric and magnetic fields, $\vec{E}$ and $\vec{H}$, are computed throughout the domain.

The domain is fully enclosed by Perfectly Matched Layers (PMLs) [6, 7] on all six boundaries to suppress artificial reflections. These layers act as highly efficient absorbers, gradually increasing conductivity with depth to smoothly attenuate outgoing waves. By preventing spurious reflections and mimicking open, semi-infinite space, PMLs ensure accurate 3D modeling of wave propagation, scattering, and energy absorption while maintaining numerical stability and physical realism.

3.3.5 Energy absorption

For a given polarization angle α of the incident laser, Maxwell's equations are solved using the FDTD method to obtain the spatially and temporally resolved electromagnetic fields. These fields then allow us to compute the local energy absorption within the material. This quantity describes the rate at which the electromagnetic energy carried by the laser is converted into heat, governing processes such as heating, melting, ablation, and self-organization. Areas of higher absorption typically act as nucleation sites for morphological changes, making this metric relevant for linking laser parameters and polarization to the resulting surface structures [144, 42].

The local energy absorption at a point (x, y, z, t) is expressed as

$$\varepsilon_{abs}(x, y, z, t) = \frac{1}{2} c\, \epsilon_0 \frac{4\pi\, \Im(\tilde{n})}{\lambda} \Re(\tilde{n}) \left|\tilde{E}(x, y, z, t)\right|^2 \tag{3.8}$$

where $\tilde{E}(x, y, z, t)$ is the complex amplitude of the electric field obtained from the FDTD simulation, capturing both the magnitude and phase of the oscillating field. The complex refractive index of the material, $\tilde{n} = n + ik$, has a real part

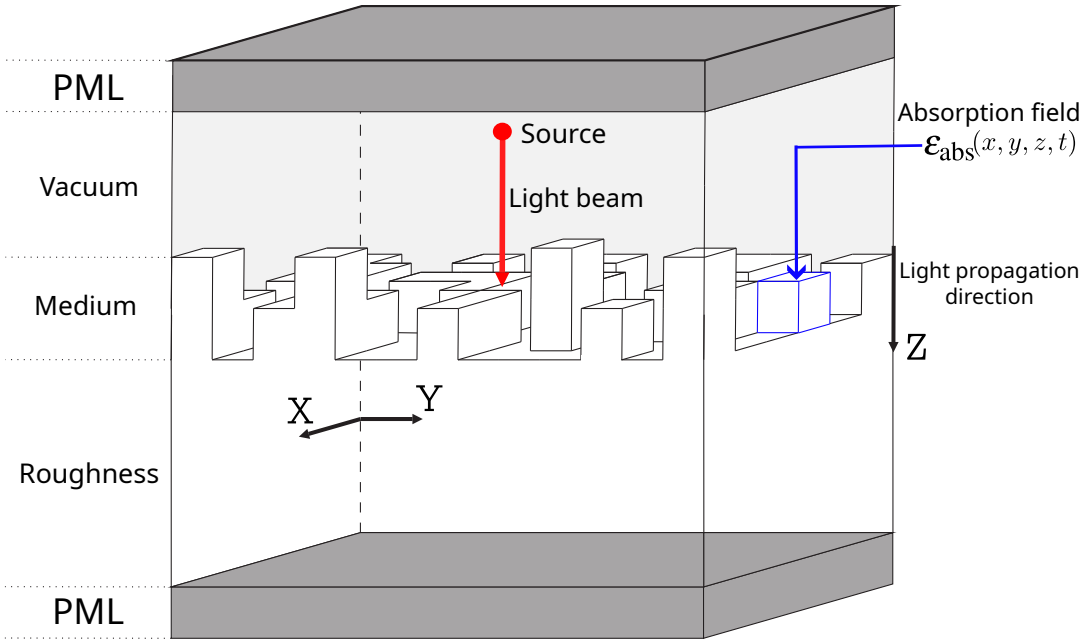


FIGURE 3.4: Simulation domain. The numerical solver simulates the propagation of a laser pulse along the vertical axis and its 3D interaction with the surface roughness, and computes a measure of energy absorbed at every location and instant $\varepsilon_{abs}(x,y,z,t)$. Adapted from [117]

$\Re(\tilde{n}) = n$ that governs the phase velocity of light in the material, and an imaginary part $\Im(\tilde{n}) = k$ that represents intrinsic optical losses responsible for absorption. The laser wavelength in vacuum is denoted by λ, c is the speed of light, and ϵ_0 is the vacuum permittivity.

3.4 Numerical experiments

3.4.1 Image-Image regression task

The absorption field $\varepsilon_{abs}(x,y,z,t)$ defined in Eq. 3.8 is inherently spatio-temporal, varying across both space and time. Directly storing and manipulating such four-dimensional data is computationally and memory-intensive, which makes repeated analyses or model evaluations costly. To address this, we reformulate the prediction task to balance computational efficiency with physical fidelity, retaining the most relevant information while simplifying the problem.

We achieve this by recasting the problem as an image-to-image regression task. The 3D surface topography is projected along the propagation axis z to produce a 2D height map. This pseudo-3D (or 2.5D) representation compresses depth information into a single channel while preserving the essential structural imprint of the material. This reduction is justified because the materials are solid, ensuring that summation along z retains the key geometric features.

The output is similarly represented as a 2D array, encoding the spatial distribution of the cumulative absorption at each (x,y) position:

$$\varepsilon_{abs}(x,y) \equiv \iint_{z,t} \varepsilon_{abs}(x,y,z,t), dz, dt. \tag{3.9}$$

Integration over time t is physically motivated by the ultrafast nature of femtosecond laser pulses: the absorption process occurs effectively instantaneously relative to slower phenomena such as heat diffusion, melting, or convection. Integration over depth z provides a practical trade-off between accuracy and efficiency. While this step discards detailed depth-resolved information, it preserves the most relevant dynamics, as absorption decays exponentially and most energy deposition occurs within the first few micrometers below the surface. Through this dimensionality reduction, the original high-dimensional spatio-temporal problem is transformed into a physically meaningful 2D regression task.

3.4.2 Simulation data generation from SEM images

The goal is to predict the cumulative energy absorption map $\varepsilon_{abs}(x, y)$ from the corresponding two-dimensional surface topography $u(x, y)$. However, $\varepsilon_{abs}(x, y)$ depends on the polarization of the electromagnetic field, denoted by α, which defines the orientation of the electric field. In the laser experiments we aim to emulate, two orthogonal polarization directions of the incident light are considered: $\alpha = 0°$ and $\alpha = 90°$. Consequently, for a given input surface $u(x, y)$, the numerical solver produces two distinct absorption maps, $\varepsilon^0_{abs}(x, y)$ and $\varepsilon^{90}_{abs}(x, y)$.

Training a neural network to simultaneously account for both polarization orientations would complicate the learning process and increase the dimensionality of the problem. To simplify this task, we exploit the isotropic nature of the material. In an isotropic medium, the optical properties, such as the refractive index n and the extinction coefficient k, are identical in all directions. Therefore, the light–matter interaction depends only on the relative orientation between the electric field and the surface features, not on any intrinsic direction of the material itself.

This isotropy allows us to train the neural network for a single fixed polarization (for example, $\alpha = 0°$). Once the network has learned to predict $\varepsilon^0_{abs}(x, y)$ from $u(x, y)$, the absorption map corresponding to the orthogonal polarization, $\varepsilon^{90}_{abs}(x, y)$, can be obtained without retraining by rotating the input surface by 90°, applying the trained model, and then rotating the predicted absorption map back by $-90°$ to restore the original orientation.

In practice, this equivalence arises from two interchangeable approaches for generating $\varepsilon^0_{abs}(x, y)$ and $\varepsilon^{90}_{abs}(x, y)$ from a single surface $u(x, y)$:

1. **Varying the polarization of the incident light:** For a fixed surface $u(x, y)$, one can compute ε^0_{abs} and ε^{90}_{abs} by applying a soft source polarized along the x-axis and y-axis, respectively.

2. **Rotating the surface geometry:** Alternatively, by keeping the polarization fixed and rotating the surface $u(x, y)$ by 90°, one obtains an equivalent result due to the isotropy of the material, effectively reproducing the absorption pattern corresponding to the orthogonal polarization. This second approach is illustrated in Figure 3.5.

For training our neural network, we adopt the second approach described above, which exploits the isotropy of the material by rotating the surface instead of changing the polarization. This strategy is particularly data-efficient, as it effectively doubles the training dataset for the given polarization the model is trained to reproduce. In practice, input topography maps are rotated by either 0° or 90° before being fed into the network, and the resulting predicted absorption map is interpreted according to the corresponding effective polarization.

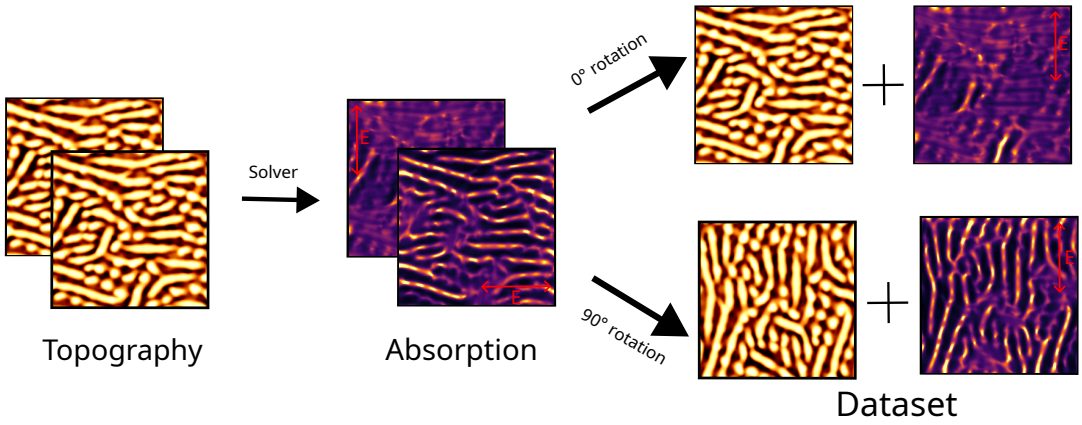


FIGURE 3.5: Illustration of the dataset construction strategy based on material isotropy. The input surface is rotated by 90° to reproduce the absorption map corresponding to the 90° polarization.

3.4.3 Neural networks architectures

We consider a set of representative neural network architectures commonly used for learning complex mappings in physical systems. These models differ in how they capture local and global dependencies, handle multiscale features, and process spatial information, providing a diverse perspective on potential approaches for predicting laser energy absorption from surface topographies.

UNet

The UNet architecture, illustrated in Figure 3.6, was originally introduced for biomedical image segmentation [104], but it has since become a standard tool in many fields requiring pixel-wise prediction. Its strength lies in combining global context with precise localization, making it particularly effective when both coarse and fine-scale features of the input are essential for the task.

The design of UNet follows a U-shaped encoder–decoder structure:

- Encoding (Contracting Path): The encoder consists of repeated applications of convolutional layers, nonlinear activations, and pooling operations. This progressively reduces the spatial resolution of the input while increasing the number of feature channels. The result is a compressed latent representation that captures the global context of the input.
- Bottleneck: At the bottom of the U-shape, the network reaches its most compressed representation. This stage contains the highest-level abstract features and serves as the bridge between encoding and decoding.
- Decoding (Expanding Path): The decoder gradually upsamples the representation back to the original resolution using transposed convolutions (or upsampling operations followed by convolutions). This allows the model to reconstruct fine details in the output.
- Skip Connections: A distinctive feature of UNet is the use of skip connections between corresponding layers in the encoder and decoder. These connections pass high-resolution spatial information from the encoder directly to the decoder, ensuring that fine details are preserved. Without them, much of the local information would be lost during the downsampling process.

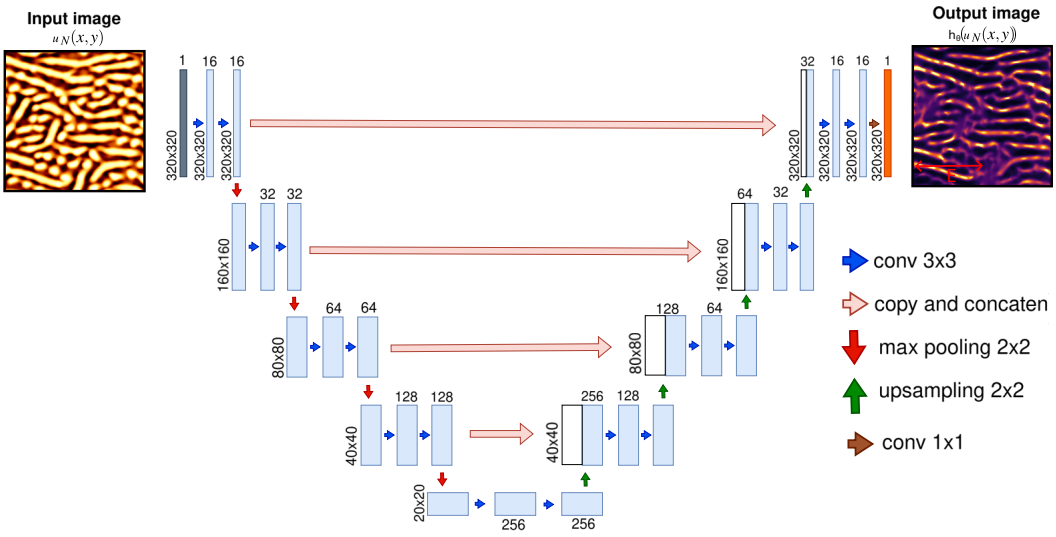


FIGURE 3.6: Schematic representation of the UNet architecture showing its symmetric encoder–decoder structure with skip connections

Through this design, UNet is able to combine global contextual information from the encoder's coarse features with local spatial accuracy from the skip connections. This balance makes it highly effective for segmentation, image-to-image translation, and, more generally, for problems where both detailed boundaries and global structures matter.

FNO

Neural operators are designed to learn mappings between function spaces, making them powerful tools for solving partial differential equations (PDEs). Among them, the Fourier Neural Operator (FNO) [69], illustrated in Figure 3.7, has emerged as one of the most widely used and effective approaches.

The core idea of FNO is to represent the input function (e.g., the surface topography in our case) in the Fourier domain, where convolutional operations can be performed more efficiently. The model consists of three main stages:

- Lifting (L): The input function is first lifted to a higher-dimensional channel space, allowing the network to capture richer representations of the input.
- Fourier Layers: Each Fourier layer contains two parallel branches:

  In the upper branch, the input is transformed into the frequency domain via a Fourier transform. A linear transformation is then applied, but crucially only to the lower Fourier modes (the dominant frequencies), which significantly reduces computational cost while still capturing the essential global structure. The result is then transformed back to the spatial domain using the inverse Fourier transform.

  In the lower branch (W), a standard linear transformation is applied in the spatial domain to capture local information that may not be fully represented by the global Fourier modes.
- Nonlinear Activation and Projection (P): The outputs of the two branches are combined, passed through a nonlinear activation, and finally projected back to

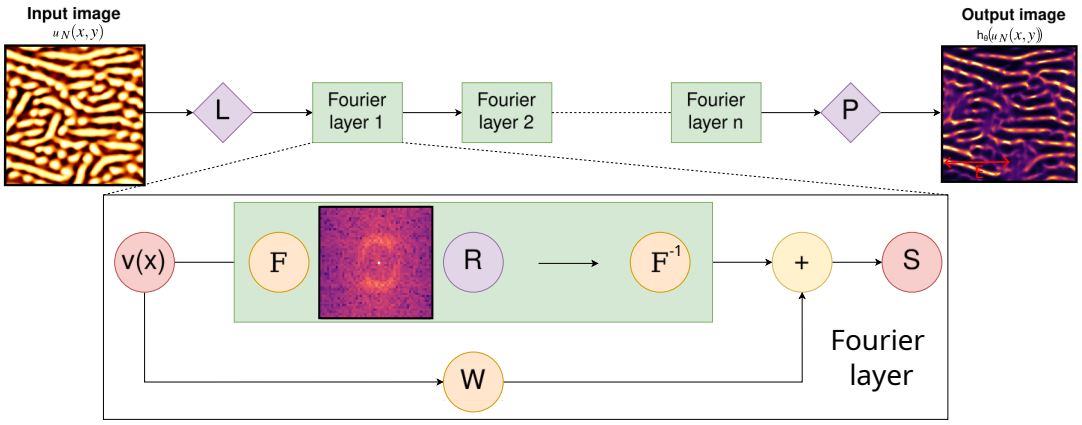


FIGURE 3.7: Schematic representation of the Fourier Neural Operator (FNO) architecture, which learns mappings between function spaces by applying convolution-like operations in the Fourier domain.

the target space, yielding the predicted output function (in our case, the energy absorption).

By alternating between the Fourier and spatial domains, FNO is able to capture both long-range dependencies (through the Fourier representation) and fine-scale local features (through the linear spatial mapping). This balance makes FNO particularly well-suited for approximating the solution operators of PDEs with complex dynamics.

UNO

The U-shaped Neural Operator (UNO) [100], illustrated in Figure 3.8, builds on the same fundamental idea of Fourier layers but introduces a different architectural design inspired by the classical UNet architecture.

Instead of directly applying a sequence of Fourier layers, UNO uses an encoder–decoder structure with skip connections, enabling the model to efficiently capture both global and local features across multiple scales:

- Encoding: The input function is progressively downsampled, being mapped into representations defined on smaller spatial domains. This process compresses the information and helps the model learn global, coarse-scale features of the input.
- Fourier Layers: At each stage of the encoding and decoding processes, Fourier layers are used to integrate information in the frequency domain, similar to FNO.
- Decoding: The compressed representation is then upsampled back to the original resolution, reconstructing the output function.
- Skip Connections: To avoid loss of fine-scale details during downsampling, skip connections are introduced between corresponding encoder and decoder stages. This ensures that local information captured early in the network is preserved and combined with the global features learned in the compressed representation.

Compared to FNO, UNO places greater emphasis on multi-scale feature extraction. While FNO is particularly efficient in capturing global dependencies, UNO enhances

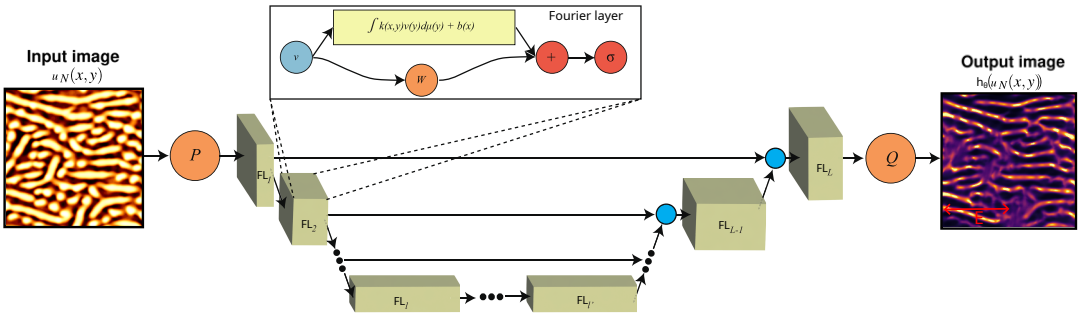


FIGURE 3.8: Schematic representation of the U-shaped Neural Operator (UNO) architecture, combining the global representation power of Fourier Neural Operators with the multiscale feature extraction of UNet. Adapted from [100].

this with its U-shaped structure, making it more effective at resolving local variations and fine geometric structures in the input.

FNO conv3x3

The FNO conv3x3 architecture is a variant of the standard Fourier Neural Operator (FNO) in which the linear transform branch, usually denoted as W, is replaced by a local convolutional operator with a 3×3 kernel.

In the original FNO, the model captures global information by transforming the input into the Fourier domain, modifying the lower Fourier modes through a learned linear transformation, and then returning to the spatial domain. In parallel, a spatial branch applies a learned linear transformation W (typically implemented as a dense layer or 1×1 convolution) to preserve local information that may not be represented in the frequency domain. In the FNO conv3x3 variant, this spatial branch is replaced by a 3×3 convolution that operates directly on the spatial grid.

This modification enhances the model's ability to capture fine-scale local interactions that are not fully represented through Fourier modes alone. The 3×3 convolution acts as a localized receptive field, similar to those in conventional convolutional neural networks, enabling the network to jointly exploit global, frequency-aware representations (via the Fourier transform) and detailed spatial features (via convolution). As a result, FNO conv3x3 often achieves better performance on tasks where the solution depends on intricate local structures in addition to long-range global dependencies.

FNO+UNet

The FNO+UNet architecture represents a hybrid framework that combines the complementary strengths of the Fourier Neural Operator (FNO) and the UNet. The FNO component excels at learning global operators through Fourier layers, efficiently capturing long-range dependencies across the domain. However, it may lack the spatial precision required to recover fine-scale local features. Conversely, the UNet architecture specializes in capturing detailed spatial structures through its encoder–decoder pathway with skip connections, making it highly effective for modeling local variations.

In FNO+UNet, the outputs of the two networks are integrated to merge global and local representations. The FNO branch processes the input in the frequency domain to extract global dependencies, while the UNet branch operates in the spatial domain

to reconstruct localized features at multiple scales. Their outputs are then combined, yielding a final prediction that maintains global coherence while preserving fine structural details.

This hybrid design is particularly advantageous for problems governed by multiscale dynamics, where both long-range correlations and small-scale features are critical. By leveraging the global operator learning capabilities of FNO together with the spatial precision of UNet, the FNO+UNet model achieves a more complete and balanced representation of the underlying physical process.

3.5 Training procedure

3.5.1 Dataset construction.

Laser–matter interaction was simulated on four sets of 51 SEM-scanned surfaces of size 800×800 pixels (Fig. 3.1), using a refractive index $n = 2.15$ and an extinction coefficient $k = 4.3$. To reduce imaging noise, all SEM images were preprocessed using a Gaussian blur with $\sigma = 4$.

The FDTD solver models the interaction of a femtosecond laser with wavelength $\lambda = 1.03\,\mu$m, using spatial resolutions of $dx = dy = 1.67\,$nm and $dz = 1\,$nm. The computational domain spans approximately $1.5\,\mu$m in each lateral direction, with the material occupying half of the z-axis and including about 20 nm of surface roughness. Perfectly Matched Layers (PMLs) of 50 cells are used to absorb outgoing waves and prevent artificial reflections.

For each surface, two orthogonal polarizations are considered, resulting in a total of $408 = 4 \times 51 \times 2$ topography–absorption pairs. To further augment the dataset and enable patch-based training, each image is divided into nine overlapping crops of 320×320 pixels, producing a final dataset of $3672 = 408 \times 9$ pairs.

Finally, all absorption maps are normalized by the energy absorbed by a flat reference surface. This normalization compensates for non-uniform laser fluence and ensures that the resulting maps express relative absorption variations compared to a flat surface.

3.5.2 Cross-validation procedure

To ensure robust model evaluation, we adopt a **leave-one-series-out cross-validation** strategy over the four SEM datasets (series). Each series in turn plays the role of the test set, while the remaining three are split as follows:

- Two series are used for training the models across all hyperparameter configurations.
- The third series is used as a validation set to assess accuracy for each configuration.
- This process is repeated across three folds, each time permuting the role of the validation set.

The best hyperparameters for each architecture are selected by averaging the validation accuracy over the three folds. With these optimal hyperparameters, the final model is retrained on the union of the three series (training + validation) and evaluated on the held-out test series. The full process is repeated four times so that every series is tested exactly once.

This procedure ensures that:

1. Every series contributes once as the test set, yielding an unbiased evaluation.
2. The selection of hyperparameters is based on unseen validation data, reducing the risk of overfitting.

3.5.3 Architectures and hyperparameter settings.

We benchmark and compare five popular Physics-Informed Machine Learning (PIML) architectures on the task of predicting laser energy absorption from SEM-derived surface topographies. Although these architectures were originally developed for a variety of physical and scientific learning problems, they provide a representative overview of the main families of models currently used for operator learning and multiscale feature extraction. The considered architectures and their key configurations are summarized below:

1. **UNet** [104], a convolutional encoder-decoder network, with the number of downsampling/upsampling blocks treated as a hyperparameter, chosen from $\{3, 4, 5, 6\}$.
2. **FNO** [69], with three key hyperparameters: the number of Fourier layers $\in \{3, 4, 5\}$, the lifting dimension $L \in \{16, 32, 64\}$, and the number of retained Fourier modes $\in \{128, 192, 256\}$ (the Fourier modes determine how many frequency components are kept in the global operator).
3. **FNO conv3x3**, a variant of FNO in which the convolution operator W is implemented with a kernel of size 3×3.
4. **FNO+UNet**, a hybrid model that combines the outputs of FNO and UNet to exploit complementary feature extraction capabilities.
5. **UNO** [100], a recently proposed operator learning architecture. We employ the best hyperparameters found with FNO and set *out_channels*=[16,32,32,16] (number of feature channels at each stage) and *scalings*=[[1.0,1.0], [0.5,0.5], [0.5,0.5], [4.0,4.0]] (spatial scaling factors for downsampling/upsampling at each stage).

3.5.4 Training objective

For each method, the goal is to learn a mapping

$$h_\theta^* : u \mapsto \varepsilon_{\text{abs}},$$

where u is a 320×320 topography image and ε_{abs} is the corresponding 320×320 absorption map. The model h_θ^* is parameterized by θ (the trainable parameters of the chosen architecture) and obtained by minimizing the following loss function:

$$\mathcal{L}(\theta) = \frac{1}{M \times 320^2} \sum_{i=1}^{M} \|h_\theta(u_i) - \varepsilon_{\text{abs},i}\|_{\mathcal{F}}^2. \tag{3.10}$$

The optimal parameters are then given by

$$h_\theta^* = \arg\min_\theta \mathcal{L}(\theta). \tag{3.11}$$

Here, M denotes the number of training pairs $(u_i, \varepsilon_{\text{abs},i})$, and $\|\cdot\|_{\mathcal{F}}$ is the Frobenius norm (i.e., the pixelwise squared error summed over the 320×320 grid). $\varepsilon_{\text{abs},i}(x, y)$ denotes the ith ground-truth absorption map, while $\widehat{\varepsilon_{\text{abs},i}}(x, y) = h_\theta(u_i)$ is the corresponding prediction.

In our leave-one-series-out setting, each run involves $M = 2 \times 51 \times 2 \times 9 = 1836$ training pairs, corresponding to 2 series $\times$ 51 surfaces per series $\times$ 2 polarizations $\times$ 9 crops per image. A small weight decay of 10^{-7} is applied for regularization, and all networks are trained for 100 epochs using the Adam optimizer.

3.6 Analysis of results

3.6.1 Table analysis

The results of the five PIML methods on the four test series are reported in Table 3.1, showing the mean absolute errors and standard deviations computed over all pixels of the test images. Considering that energy absorption values range from 0 to about 1000 depending on the topography, even the largest errors remain relatively small. Overall, all models achieve reasonably good predictive performance, but notable differences emerge in terms of accuracy, robustness, and computational efficiency. These differences become particularly meaningful when considering the low-data regime and the heterogeneous nature of the series.

	Test (on series 1)		Test (on series 2)		Test (on series 3)		Test (on series 4)		Av. training	Nb of
Model	mean	(σ)	mean	(σ)	mean	(σ)	mean	(σ)	time (s)	parameters
UNet	8.6	(9.9)	6.0	(6.5)	5.1	(4.5)	12.0	(14.6)	1740	∼2e+6
FNO	20.2	(22.3)	13.7	(16.0)	10.0	(11.3)	19.2	(20.3)	5127	∼1e+8
FNO conv3x3	19.3	(23.8)	14.4	(15.0)	9.6	(10.8)	18.9	(20.9)	5247	∼1e+8
FNO+UNet	16.7	(20.7)	11.2	(13.9)	8.9	(9.3)	17.7	(20.0)	13317	∼1e+8
UNO	25.2	32.4	15.7	18.5	10.9	11.9	30.8	33.2	4273	∼1e+8

TABLE 3.1: Comparison of the 5 methods on the 4 test series in terms of errors (mean and standard deviation σ) as well as average training time and number of parameters.

Among all models, the simple UNet architecture stands out as the best performer. It consistently achieves the lowest errors across most test series while requiring the fewest parameters (∼2M) and shortest training time (∼1740 s). This demonstrates that UNet effectively captures the spatial dynamics of the patterns in this low-data regime. Its performance is particularly strong on series 3, which exhibits delayed pattern emergence and relatively simple structures. The slightly higher errors on series 4 can be explained by the presence of cavities that are under-represented in the other series, making them more challenging to predict. Overall, UNet's combination of local convolutional layers and skip connections provides a robust inductive bias, allowing it to generalize well even when data is limited or partially overlapping.

The three FNO-based models show more nuanced behavior, illustrating the impact of architectural modifications. The standard FNO achieves relatively high errors across all series, reflecting its strength in modeling global dependencies but its limitations in capturing fine local details when training data is scarce. Introducing a local convolution (FNO conv3x3) slightly reduces the error on some series, suggesting that local receptive fields help capture finer spatial features, but the improvement is limited compared to the significant increase in training time (∼5247 s) and parameter count (∼100M). The FNO+UNet combination produces the best results among the FNO variants, with moderate gains in error reduction, particularly on series 2 and

3, but at a very high computational cost (∼13317 s). This indicates that combining global Fourier modes with local convolutional refinement is useful, but the gains are marginal in this low-data regime. Series-specific analysis further reveals that FNO and FNO conv3x3 perform similarly on series 1, suggesting that early-stage patterns are sufficiently simple for both models to capture, while more complex patterns like those in series 4 benefit more from the UNet refinement.

UNO exhibits the highest errors across most series despite having a parameter count similar to FNO and FNO+UNet (∼100M). Its training time (∼4273 s) is intermediate, but the results indicate that the model struggles to generalize from limited data, particularly for series 4, where cavities are more prevalent. This highlights a key limitation: while UNO is designed to handle multi-scale features and global dependencies, it lacks the inductive bias that benefits simpler architectures like UNet when data is scarce and heterogeneous.

Interestingly, for all models, the test loss is observed to be lower than the training loss. This effect can be attributed to partial data overlap: the initial flat surface in all series leads to early-stage patterns in the test set that resemble training samples from other series. While this inflates apparent test performance, it does not diminish the overall conclusion: UNet remains the most efficient and reliable model. Moreover, analyzing the standard deviations across series provides additional insights. series 4 consistently exhibits higher standard deviation σ, reflecting the difficulty of predicting under-represented cavities, while series 3 shows low variability, confirming its relative simplicity.

In summary, the comparative analysis confirms that simplicity and appropriate inductive bias are crucial in low-data regimes. UNet achieves low errors, handles heterogeneous series robustly, and is computationally very efficient. The FNO family demonstrates that adding local convolutions or combining with UNet can slightly improve accuracy but at a high cost, and performance varies across series. UNO, despite its complexity, fails to compete in these limited-data conditions. The results also highlight the importance of dataset structure: overlapping features between training and test sets can mask true model limitations, emphasizing the need for careful experimental design in low-data predictive tasks.

3.6.2 Figure analysis

Figure 3.9 presents representative prediction results for the four series. From left to right, the columns show: the topography u (for a fixed vertical polarization), ground truth energy absorption ε_{abs}, UNet prediction $h_\theta(u)$, UNet relative error $(h_\theta(u_N) - \varepsilon_{\text{abs}})/\varepsilon_{\text{abs}}$, a histogram of UNet's relative errors, and finally, the relative errors of FNO, FNO conv3x3, FNO+UNet, and UNO.

The first column shows the topography $u(x, y)$, which corresponds to the surface to be irradiated with a vertical polarization. This column serves as the reference input for the models, providing the spatial structure on which energy absorption occurs.

The second column shows the ground truth energy absorption $\varepsilon_{\text{abs}}(x, y)$. Comparing this with $u(x, y)$, it is clear that absorption patterns closely follow the underlying topography, but also reflect non-trivial interactions such as local polarization effects. These maps serve as a reference for evaluating the accuracy of the model predictions.

In the third column, UNet predictions $h_\theta(u_N(x, y))$ reproduce the ground truth with remarkable fidelity. Both global trends and fine local structures are captured, including subtle variations in high-absorption regions and along complex topographical features. The ability to generalize even to patterns not previously observed in the

training set highlights the model's robustness, especially given the limited number of training examples.

The fourth column, showing UNet's relative error $(h_\theta(u_N) - \varepsilon_{\text{abs}})/\varepsilon_{\text{abs}}$, reveals that most prediction errors are small and localized. Lighter areas indicate better accuracy, while residual structured errors often correspond to regions with complex topography or abrupt polarization-induced variations. These structured errors suggest that while the model captures the main features well, there is still room for refinement in areas where fine-scale topographical or polarization interactions dominate.

The fifth column presents a histogram of UNet's relative errors. This provides a complementary quantitative perspective, showing that the vast majority of pixels have very small errors and only a few exhibit larger deviations. This histogram reinforces the observation from the spatial error map that UNet predictions are both precise and consistent.

The final four columns display the relative errors for FNO, FNO conv3x3, FNO+UNet, and UNO. Interestingly, these four models have very similar error patterns, indicating that their limitations are consistent across architectural variations. However, their errors are clearly larger and more diffuse compared to UNet. These models tend to underperform in regions with strong local variations or early-emerging patterns in the topography, likely due to their inability to fully exploit the low-dimensional 2D height representation and polarization-informed input rotation.

3.6.3 Computational speedup

The numerical solver used in our simulations was implemented on a GPU, with the simulation size chosen to fit within 24 GB of available memory. On average, a single simulation corresponding to one SEM image requires about $\sim$30 minutes ($\sim$1857 s) to complete. Extrapolated to the entire dataset of 408 images, this amounts to approximately 8.5 GPU-days of computation.

In contrast, training the U-Net is remarkably efficient. Completing 100 epochs takes less than $\sim$30 minutes on the same hardware. Once trained, the network can process full-resolution 800×800 images almost instantaneously. Because the model is fully convolutional, it can operate on arbitrary image sizes without modification. Averaged over the dataset with a batch size of 30, the inference time per image is just 12.5 ms. Compared to the 30 minutes required for a single numerical simulation, this corresponds to a speedup of more than four orders of magnitude ($\sim 1.4 \times 10^5$).

Even under stricter latency conditions, with a batch size of 1, inference remains extremely fast at 97 ms per image, still several thousand times quicker than the numerical solver. On a CPU, the network continues to offer a substantial advantage, requiring only 1.3 s per full-size image.

Overall, these results underline the dramatic computational benefit of the trained U-Net. By combining GPU acceleration with a fully convolutional design, the model delivers high-fidelity predictions in real time, enabling large-scale analysis that would otherwise be computationally prohibitive with direct numerical simulations.

3.7 Preliminary physical insights from the surrogate

As a preliminary exploration, we use our trained UNet as a fast surrogate to probe basic questions about energy absorption mechanisms. Specifically, we examine

whether insights obtained from the UNet predictions align with those from the high-fidelity numerical solver.

These initial investigations focus on the total absorption of a surface, reducing the full spatio-temporal absorption field to a single scalar quantity via integration over all spatial coordinates and time:

$$E_{\text{abs}} := \iint_{x,y,z,t} \varepsilon_{\text{abs}}(x, y, z, t)\, dx\, dy\, dz\, dt. \tag{3.12}$$

This scalar captures the net energy dissipated into the material and summarizes the total absorption over the surface and irradiation time. While these studies are preliminary, they illustrate the potential of the UNet surrogate for rapid analysis. In Chapter 4, we present a more comprehensive study that would have been impractical using the numerical solver alone due to its high computational cost.

3.7.1 Evolution of total absorption with increasing double pulses

The first question investigates how the emergence of self-organized surface structures affects the material's ability to absorb energy and whether the absorption behavior changes as self-organization develops. To address this, we measure the evolution of the total absorption with the number of double pulses. Each double pulse consists of two consecutive cross-polarized irradiations at 0° and 90°, and the total absorption is computed as the sum $E^0_{\text{abs}} + E^{90}_{\text{abs}}$. We note that, unlike in the physical experiments that produce self-organized surfaces, the surface is assumed to remain unchanged between the two pulses in this preliminary analysis.

This time, the UNet is applied not to cropped 320×320 patches, but to the full 800×800 irradiated surfaces. Because the network is fully convolutional, it can process larger inputs directly without retraining, reusing the same learned filters from the smaller patches. This approach assumes that the statistical distribution of features at the patch scale is representative of the entire surface. While it efficiently captures large-scale trends, it may miss rare or spatially extended features that were under-represented in the training patches, potentially introducing local inaccuracies in the predicted absorption. Using this method, we can obtain a global prediction of the total energy absorption in a single forward pass, which we then normalize by its initial value at $N = 0$ (no irradiation) to account for the arbitrary scale of the numerical solver's raw outputs.

Figure 3.10 compares the total absorption obtained from UNet predictions with the ground-truth solver outputs across both training and test series. When analyzing the evolution of absorption as a function of the number of double pulses, several distinct regimes emerge.

Initially, for low values of N, the numerical solver and the UNet proxy indicate that absorption remains stable and relatively low. This regime reflects the early phases of irradiation, where the surface topography is still uniform and close to its pre-irradiation state since surface reorganization has not completely started.

As irradiation continues, the system enters an intermediate regime in which absorption increases significantly. This rise corresponds to the effects of structural modifications induced by repeated laser pulses. The surface gradually develops features such as grooves, ripples or other self-organized patterns, which enhance the coupling between the incident electromagnetic field and the material. These features multiply until the system approaches saturation. At this stage, absorption reaches a plateau, indicating that the formation of new absorbing channels is balanced by

the stabilization of existing structures. Physically, this plateau marks the point at which the surface has acquired sufficient complexity to maximize interaction with the incident light, and further irradiation primarily preserves or slightly perturbs this state.

A closer examination reveals a systematic bias in the UNet predictions. The surrogate consistently underestimates the absolute absorption values but it reproduces the correct trend of the curve. It indicates that, although quantitative adjustments may be necessary to align the predicted values with the numerical solver, the network accurately captures the physical dynamics. The identification of growth, stabilization, and decline regimes remains intact, demonstrating that the UNet can reproduce the essential physical transitions governing energy absorption even when its raw estimates differ from the solver.

Beyond these general trends, series-specific differences provide additional informations. In some series, the solver indicates that absorption decreases after many double pulses. This reduction is associated with the degradation of surface structures, which diminishes their ability to couple effectively with the incoming light. Other series display continuous growth in absorption without plateauing or declining. For example, in the third series, absorption continues to rise steadily with increasing N. This behavior aligns with the observation that the corresponding morphology, shown in Fig. 3.10, has not yet reached a state of chaotic patterns. The structures remain sufficiently ordered to maintain efficient absorption over extended irradiation. Despite requiring a lower computational cost, the network retains the capacity to support the same physical interpretations as the numerical solver.

3.7.2 Relative absorption between orthogonal polarization

The second question investigates the anisotropy of absorption, which refers to the directional dependence of energy absorption: do the self-organized structures formed under irradiation absorb energy differently depending on the polarization, or are they largely polarization-independent? Anisotropic absorption arises when the surface morphology develops a preferential orientation, causing the material to interact more strongly with light polarized along a certain direction. In contrast, isotropic structures, which are uniform or disordered at the relevant length scales, absorb energy similarly regardless of polarization. This distinction is important because polarization dependence provides a direct indication of the underlying order in the surface morphology.

To quantify this effect, we compute the ratio $E^0_{\text{abs}} / E^{90}_{\text{abs}}$, where E^0_{abs} and E^{90}_{abs} denote the total absorbed energy under the two orthogonal polarizations. A ratio consistently above or below one indicates that one polarization couples more efficiently to the nanostructures, whereas values fluctuating around one suggest polarization independence.

Figure 3.11 shows the predicted and ground truth evolution of this ratio for the different experimental series. Two distinct behaviors are observed. In some series, the ratio departs consistently from unity, indicating that the surface has developed an anisotropic response. This reflects the formation of self-organized structures, such as ripples or grooves, that are preferentially oriented along one direction. The gradual increase or stabilization of the ratio away from unity signals the strengthening of this directional preference as irradiation progresses. These results demonstrate that the model can track not only the overall magnitude of absorption but also the more subtle polarization-dependent features associated with self-organization.

In other series, the ratio fluctuates around unity without any sustained deviation. This indicates that the self-organization does not favors a specific orientation for energy absorption. Such behavior corresponds to chaotic or highly disordered surfaces, where the incident light couples similarly to both polarizations. In these cases, the UNet accurately reproduces the qualitative distinction between anisotropic and isotropic absorption.

Although the UNet predictions are generally smoother than the raw solver outputs, this smoothing can be beneficial, as it reduces small-scale numerical noise while preserving the essential physical trends. Importantly, both the UNet and the solver lead to the same physical conclusions: series that develop anisotropy exhibit systematic polarization dependence, whereas isotropic or disordered surfaces produce ratios close to unity.

3.8 Conclusion and perspectives

In this chapter, we presented a module for predicting the energy absorption of irradiated surfaces by transforming the problem into an image-to-image regression task. By leveraging both the structure of the data and the inherent physical symmetries, we showed that a UNet architecture, despite its relative simplicity compared to more complex models, achieves better performance in the low-data regime. The network successfully captures the relationships between surface morphology and energy absorption distribution, providing accurate predictions while remaining computationally efficient. Notably, preliminary results reveal a strong correlation between absorbed energy and the degree of self-organization of the surfaces, demonstrating that the model not only interpolates the data but also encodes physically meaningful behavior.

The main perspective opened by this work lies in leveraging this differentiable and extremely fast neural solver as an element of the end-to-end framework for learning of the dynamics leading to the formation of self-organized patterns.

However, the significance of this approach extends beyond mere speed. Access to fast and differentiable absorption predictions unlocks the possibility of large-scale exploration of irradiation scenarios that would be prohibitively expensive with direct solvers of Maxwell's equations. This capability can be harnessed for inverse design tasks, where the objective is not only to predict absorption but also to identify surface patterns optimized for specific optical responses such as maximal or minimal global absorption and directional selectivity. Such functionality would constitute an interesting tool for designing metasurfaces with applications in optical devices, sensing platforms and energy-harvesting technologies. Moreover, the neural solver's speed allows rapid predictions of energy absorption distribution on unseen surfaces in order to have a functional end-to-end framework for self-organization modelling.

Focusing only on two-dimensional absorption maps represents a first but limited step. Many of the most intricate phenomena in laser–matter interactions, particularly those governing long-term surface evolution, depend critically on the three-dimensional distribution of deposited energy within the material. Updating the solver to consider 3D absorption would therefore provide a more complete picture of the self-organization mechanisms. This could be achieved either by computing absorption at full spatial resolution or by using a different but physically meaningful representations such as predicting three absorption values per spatial location corresponding to the surface, shallow and deep layers.

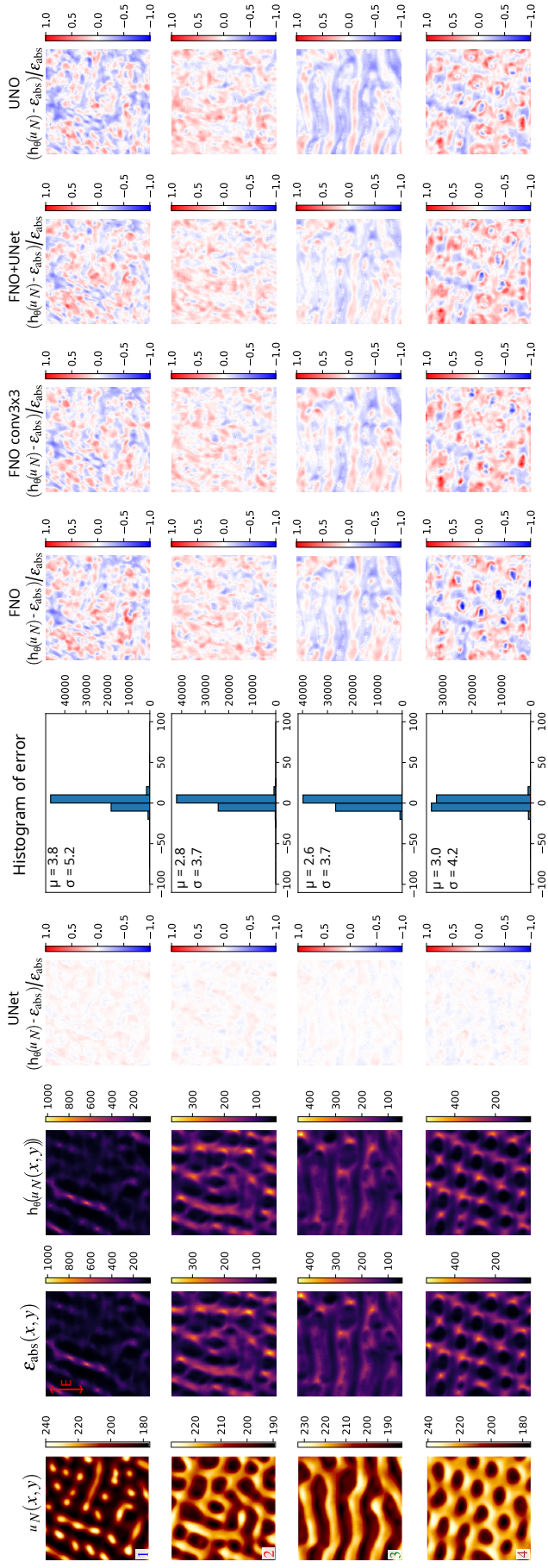


FIGURE 3.9: Illustration of prediction results. From left to right: Topography $u_N(x,y)$, ground truth energy absorption $\varepsilon_{abs}(x,y)$, predicted energy absorption $h_\theta(u_N(x,y))$ by UNet, relative error of UNet prediction $(h_\theta(u_N) - \varepsilon_{abs})/\varepsilon_{abs}$, relative error histogram, relative error of FNO, FNO conv3x3, FNO+UNet and UNO.

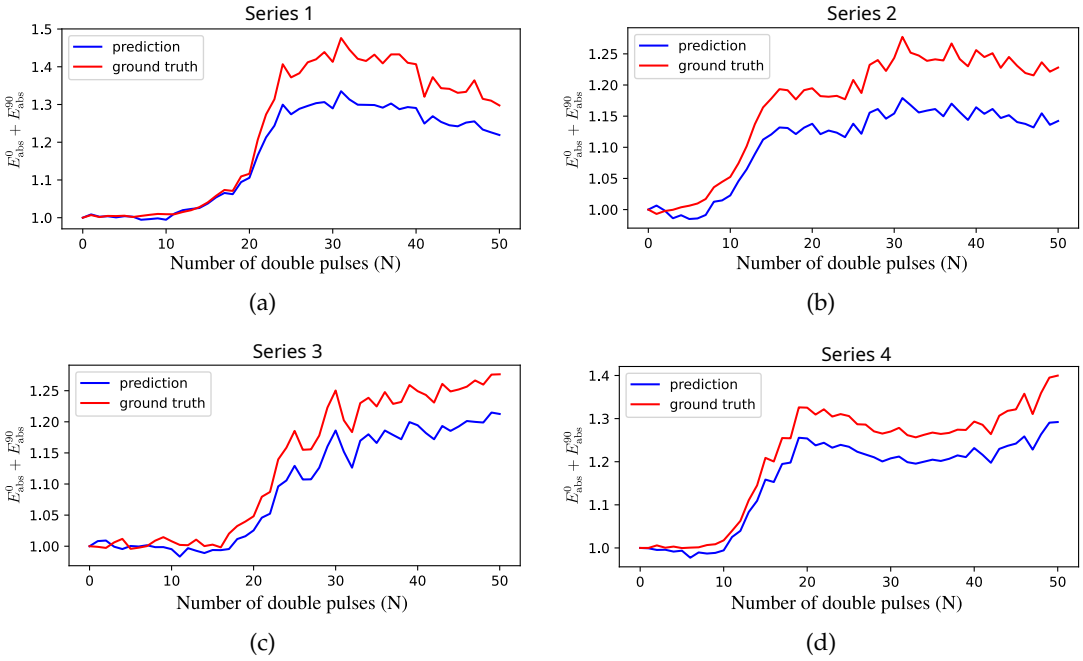


FIGURE 3.10: Overall absorption (sum of the two pulses) as a function of the number of cross-polarized pulses within the series. The absorption tends to increase with self-organization. The predictions closely follow the ground truth.

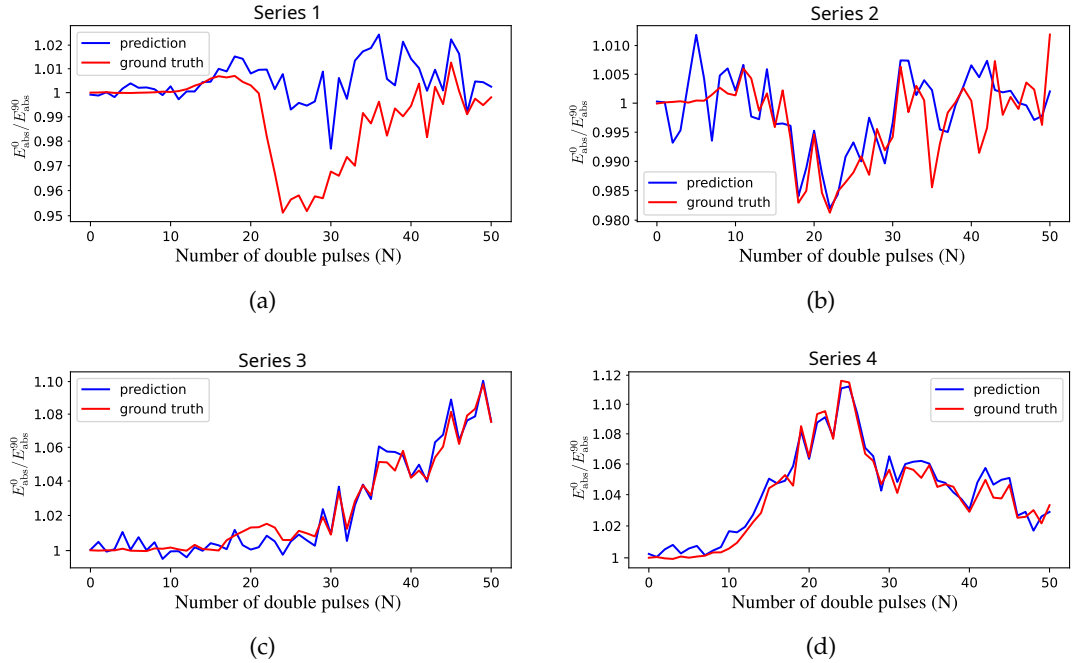


FIGURE 3.11: Absorption difference between the two polarizations presented as a ratio (1 means no difference). The first two series show no directional preference, while the others specialize to an orientation after self-organization starts. The predictions closely follow the ground truth.

Chapter 4

Learning to Learn in Laser–Matter Interactions: Linking Self-Organization to Learning Dynamics

In this chapter, we present the third contribution of this thesis, showing how self-organized surface patterns left by ultrashort laser pulses act as a structural memory, guiding the absorption of subsequent light. Rather than passive imprints, these patterns participate in a learning process, adaptively reshaping the surface morphology to optimize energy capture over multiple pulses. Our analysis leverages the fast, differentiable, and generalizable surrogate model for absorption prediction introduced in the previous chapter, enabling efficient exploration of the underlying dynamics.

4.1 Introduction

Intelligence extends beyond consciousness to the ability to interact and adapt to the subtleties of the surrounding universe, whether living or inanimate. This principle transcends the boundaries of biology to apply to material entities, such as surfaces excited by light. The interaction between a metallic surface and a pulsed energy flux, such as an ultrafast laser, offers a unique opportunity to understand the intrinsic mechanisms of learning, as the signature of the learning process can be sequentially imaged by the pulse-to-pulse irradiation. Norbert Wiener's cybernetic viewpoint reveals that machine intelligence is defined by its ability to efficiently process information to achieve specific goals[136]. This translates into a learning process where the machine adjusts its behaviors based on feedback from its environment, akin to the functioning of neural networks in artificial intelligence, which adjust their weights and connections based on training data. For photoexcited surfaces, the transient morphology reflects the nuanced response to electromagnetic fields on the surface reliefs, adapted or learned over time. This chapter explores how materials progressively self-organize in response to successive light flux, focusing on the evolution of surface asperities under repeated pulses[13, 144], which, we show, can be seen as a *learning mechanism*.

Intelligent surfaces, at the intersection of condensed matter physics, nanotechnology, and materials science, explore the emergent properties resulting from organization and structuring at the nanoscale and microscopic level[121, 123]. At these scales, fundamental physical phenomena such as surface interactions, chemical bonds, Van der Waals and intermolecular forces, as well as thermal fluctuations and near-field electromagnetic interactions dominate[107, 105], influencing the collective behavior

of microscopic entities and their macroscopic reaction. For example, by controlling roughness and texture at the nanoscale, significant modifications can be made to the tribological properties of a surface, affecting friction and adhesion [133]. Similarly, inspired by natural phenomena such as plant microstructures, surfaces with specific functionalities, such as self-cleaning or controlled adhesion, are designed using biomimetism[146, 122]. By integrating metaphotonic approaches for manipulating light at the nanoscale[2], new perspectives emerge for creating tailor-made intelligent surfaces, with applications in optical sensors or 5D optical data storage[114, 138]. Thus, intelligent surfaces highlight how an ordered microscopic structuring can lead to exceptional macroscopic behaviors, opening new possibilities for the development of photonic functionalities and adaptive materials[56, 83].

Self-organized surfaces emerge from systems out of equilibrium in which entropy flux optimization favors the formation of dissipative structures at the nanoscale [105, 98]. Today, it is widely accepted that under the impact of ultra-short laser pulses, hydrodynamic instabilities, including thermoconvective phenomena, can be controlled reproducibly and predictively[106, 129]. These systems exhibit remarkable collective effects at scales well below the wavelength of light, significantly altering local optical properties and capable of generating hybrid electromagnetic modes, localized surface plasmons, or local field enhancement effects[107, 94].

Under normal conditions, materials are not expected to modify their optical absorption properties simply due to steady illumination. A painted surface, for instance, does not darken when exposed to sunlight. However, in cases where the material's albedo decreases upon exposure, a positive feedback loop can emerge: increased absorption alters surface properties in a way that further reduces reflectivity, amplifying energy uptake until the system reaches a critical threshold[80].

Some systems inherently regulate absorption up to a specific limit, beyond which instability ensues. Plants exemplify this delicate balance[33]. As they grow, they optimize light capture by adjusting leaf morphology, including size, shape, and orientation[134]. However, excessive absorption can be detrimental. Mediterranean plants have evolved specialized strategies such as evaporative cooling and protective coatings, which contribute to their distinctive aromatic emissions[93]. Additionally, plants have developed selective light absorption mechanisms: leaf epidermal layers accumulate flavonoids in response to UV-B radiation, reducing epidermal transmittance and mitigating potential damage, while remaining sensitive to other wavelengths essential for photosynthesis[126, 88]. These adaptations, shaped by evolutionary pressures, enable them to dynamically modulate light absorption through microstructural variations, pigmentation adjustments, and controlled leaf orientation, thereby mitigating the risks associated with excessive energy uptake[67, 25].

In contrast, inanimate materials lack such adaptive mechanisms, making them highly susceptible to uncontrolled absorption feedback. A perfectly smooth metallic surface behaves as a mirror, absorbing energy in the visible-IR range only within a shallow skin depth[82]. However, even nanoscale surface roughness disrupts this behavior, introducing spatial inhomogeneities in absorption and rendering it highly sensitive to the polarization of incident light[44, 143]. Energy uptake on rough metallic surfaces arises from both radiative and non-radiative contributions: far-field scattering, governed by surface features relative to the incident wavelength, and near-field interactions, which involve localized plasmonic excitations and evanescent waves, leading to enhanced subwavelength-scale energy deposition[90]. Under intense laser irradiation, this intricate interplay between near- and far-field effects dictates the absorbed energy distribution, driving nanoscale pattern formation, localized heating, and surface restructuring[105].

Informed by the reconstructed dynamics of laser-induced self-organized systems and guided by solutions of Maxwell's equations, we develop a numerical framework that leverages machine learning to uncover the progressive stages of photonic information processing. In this approach, we apply the fast, differentiable, and generalizable surrogate for absorption prediction introduced in the previous chapter, enabling efficient evaluation of energy absorption across complex surface configurations. Building on this methodology, this chapter makes the following key contributions to the understanding of laser–matter interactions:

- We show that the self-organized patterns forming on metallic surfaces act as an adaptive structural memory rather than passive imprints of laser exposure. The surface topography dynamically encodes information from previous irradiations, regulating and optimizing the absorption of subsequent pulses. This process exhibits hysteresis: the system's response depends on its history, with past interactions shaping and reinforcing emerging patterns while filtering irrelevant fluctuations. Overall, this demonstrates a learning-like mechanism, where information delivered by the light stimulus is stored and exploited to guide the progressive structuring of the surface.
- We identify a direct and robust correlation between statistical measures of surface complexity and the efficiency of energy absorption. In particular, the Lopez-Ruiz–Mancini–Calbet (LMC) [77] complexity emerges as a predictive metric, providing quantitative insight into how the degree of order and information content within the nanostructured surface governs optical performance. This finding establishes a rigorous tool for evaluating and optimizing laser-induced surface morphologies for targeted energy manipulation.
- We show that when metallic surfaces are subjected to multiple laser irradiations, self-organization occurs in a way that maximizes energy absorption. The resulting surface patterns are not arbitrary but represent configurations that optimize the coupling and retention of laser energy. To validate this, we perturb the supposedly optimal patterns in two distinct ways: first, by replacing a random portion of the pattern with a synthetic patch generated from the Swift–Hohenberg equation; second, by substituting a random portion with a patch extracted from a different pattern.

4.2 Preliminary background

4.2.1 Learning and memory

Memory and learning are fundamental concepts in understanding how organisms interact with their environment. Memory can be understood as a theoretical construct that allows us to determine when a subject's current behavior is influenced by previous experiences. It encompasses the storage, retention, and retrieval of information gained through experience. Learning, on the other hand, refers to a relatively enduring change in behavior or neural processes resulting from interaction with environmental events. The two concepts are inherently interconnected: memory provides the substrate that makes learning possible, while learning actively shapes and modifies memory by encoding new experiences. Both memory and learning are inferred from behavior and performance rather than being directly observable, making them abstract but essential constructs in neuroscience and psychology [34, 118].

Types of learning

Learning processes can occur both consciously and unconsciously, and they can be classified broadly into explicit (declarative) and implicit (non-declarative) forms [23]. Explicit learning involves conscious effort to acquire facts, knowledge, or events that can be verbally reported, whereas implicit learning involves unconscious, automatic adjustments to behavior or performance based on experience, without requiring conscious awareness.

(i) Associative learning Within implicit memory, several types of learning are distinguished. Associative learning refers to processes in which relationships between stimuli or between behaviors and their consequences are established [59]. In classical conditioning, a neutral stimulus is repeatedly paired with a naturally significant stimulus, eventually causing the neutral stimulus to elicit a conditioned response. Pavlov's classic experiments [10] exemplify this mechanism, in which dogs began to salivate in response to a sound that had been associated with food. Operant conditioning, also known as trial-and-error learning, involves modifying voluntary behaviors based on their consequences. Behaviors that are followed by positive outcomes, such as rewards, tend to be repeated more often, while behaviors followed by negative outcomes, such as pain or failure, tend to be suppressed. Associative learning is fundamental to adaptive behavior, allowing organisms to predict and respond to significant environmental cues by forming meaningful connections between experiences.

(ii) Non-associative learning Non-associative learning involves changes in behavior in response to repeated exposure to a single stimulus, without forming associations between multiple stimuli [59]. This type of learning is often simpler than associative learning but is essential for survival, as it allows organisms to efficiently allocate attention and energy based on the significance of environmental events. Two primary forms of non-associative learning are habituation and sensitization [59]. Habituation occurs when repeated exposure to a non-threatening stimulus results in a decreased response over time [63]. This mechanism is adaptive because it prevents organisms from wasting energy and attention on stimuli that are irrelevant or harmless, allowing them to focus on more important environmental cues. Examples include ignoring the constant sound of distant traffic or ceasing to respond to a repeated, mild tactile stimulus. Sensitization, in contrast, results in an increased response to a stimulus, often following exposure to a strong or noxious stimulus. It enhances an organism's ability to detect and react to potentially important or dangerous events. Sensitization is observed across many sensory modalities and is a form of primitive learning that allows organisms to improve their reactivity and survival. For instance, repeated exposure to painful stimuli can heighten responsiveness, ensuring faster reactions in the future. In more complex organisms, sensitization contributes to attentional modulation and adaptive responses to environmental changes.

(iii) Structural learning Beyond associative and non-associative mechanisms, learning can also occur at the level of system structure in complex adaptive systems. Structural learning refers to the ability of a system to adapt its internal organization or configuration in response to repeated interactions with its environment, thereby improving performance over time [72, 74]. In such systems, individual components interact through feedback loops, and the collective behavior encodes information about past experiences. Unlike classical forms of learning that primarily modify

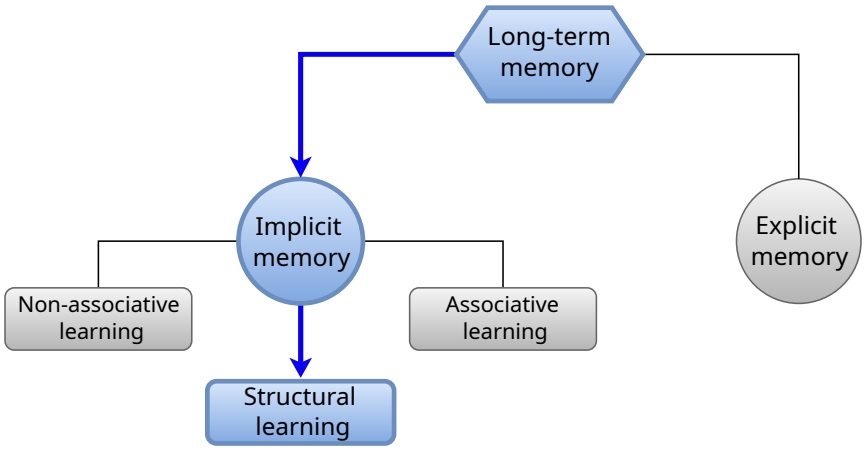


FIGURE 4.1: Classification of long-term memory and learning systems. Structural learning, highlighted here, represents a higher-order process within implicit memory that enables the extraction of regularities and rules from experience. This concept provides a direct analogy for how materials evolve their surface architecture under repeated irradiation. Learning and memory systems are shown as interdependent rather than isolated mechanisms.

responses or behaviors, structural learning reshapes the system's architecture itself, creating a memory of effective configurations. A key aspect of structural learning is "learning to learn" [18], which can be defined as the system's ability to improve its future learning efficiency by leveraging knowledge of past structural adaptations. In other words, rather than only optimizing for a single task, the system organizes itself to extract invariants and reduce the dimensionality of the space it must search for solutions. This allows faster adaptation to novel but structurally related tasks, effectively transferring organizational knowledge from one learning experience to another. Structural learning thus provides a mechanism for targeted, experience-dependent acceleration of adaptation, in contrast to non-specific strategies that simply speed up parameter changes without exploiting task structure.

Taken together, associative and non-associative learning highlight the fundamental strategies by which biological systems adapt to repeated experiences. Associative learning allows the formation of predictive links between stimuli and outcomes, while non-associative mechanisms such as habituation and sensitization regulate responsiveness to recurring inputs. These processes are traditionally considered core components of implicit long-term memory, as illustrated in Fig. 4.1, which also emphasizes structural learning, a higher-order form of implicit learning responsible for extracting and internalizing abstract regularities from experience. In this work, we extend these concepts to materials science, proposing that self-organization under repeated laser irradiation can be interpreted through a similar learning framework. Sensitization provides a particularly compelling analogy: with each successive pulse, the surface undergoes cumulative structural reorganization, increasing both morphological complexity and absorption efficiency. Beyond this, the process exhibits characteristics of structural learning, whereby the material effectively identifies recurring spatial and energetic patterns from prior exposures and reorganizes its architecture to optimize future responses. In this sense, irradiated surfaces integrate principles of associative, non-associative, and structural learning, progressively learning to enhance energy absorption through self-organization.

Learning in non-living system

Early attempts to extend the concepts of learning and memory beyond living organisms often focused on physical, chemical, and electronic systems that could retain traces of past inputs and modify their responses accordingly [62, 142, 38]. These investigations did not claim that such systems were "learning" in the biological or cognitive sense, but they highlighted structural, dynamical, or functional parallels that offered new perspectives on adaptation and memory-like behavior in non-living matter.

In material science, magnetic materials provided one of the earliest examples. Their magnetization depends not only on the current applied magnetic field but also on the history of previously applied fields. This phenomenon, known as hysteresis, illustrates a memory effect: the system carries information about its past into its present state [62, 8]. Similarly, ferroelectric and shape-memory alloys exhibit comparable history-dependent behaviors, where past thermal or mechanical inputs determine their present configuration and properties [32]. These materials effectively encode previous interactions in their internal structure, providing a direct analogy to memory storage.

In chemistry, dynamical systems such as the Belousov–Zhabotinsky (BZ) reaction became a reference point [45, 40]. These oscillating reactions display phase-dependent responses, where the outcome of a new stimulus depends on the phase of prior oscillations. In other words, the system "remembers" its temporal history through altered oscillatory patterns. More generally, autocatalytic and feedback-driven chemical networks were interpreted as chemical memories, since they retained the influence of earlier inputs over long timescales through concentration shifts or stable pattern formation [19].

In the field of mechanics, researchers turned to elastoplastic and viscoelastic materials, which store information about past mechanical loads [116, 43]. For example, when such a material is repeatedly stressed, its future stress-strain behavior is altered, revealing a "memory" of previous deformations. This concept was later extended to granular matter, where the way grains rearrange under compression depends strongly on the sequence of prior loadings [61, 127]. These systems demonstrated that structural memory could emerge purely from mechanical interactions without any biochemical or neural substrate.

The advent of electronics introduced another powerful analogy through memristors [28] and other memory-based circuit elements. Memristors possess a resistance that depends on the integral of past currents. This means that their electrical response is shaped by their input history, enabling them to behave analogously to biological synapses that strengthen or weaken depending on prior activity. This discovery opened the door to neuromorphic engineering, where circuits inspired by brain-like learning processes are built using non-living substrates [84].

Overall, these early iterations demonstrated that memory could be understood broadly as a system's capacity to retain information about past states, while learning could be framed as an adaptive modification of response patterns when similar inputs are encountered again. Although these systems lack cognition or awareness, they established a framework for exploring parallels between biological learning and physical processes. They also laid the groundwork for modern interpretations of adaptation and self-organization in non-living systems, where feedback, history dependence, and structural changes play roles strikingly reminiscent of learning and memory in organisms.

4.2.2 Physical experiments on laser-induced self-organization

The experimental investigation of laser–matter interaction provides the foundation for understanding how surfaces can adaptively reorganize under repeated irradiation. In this section, we present the physical experiments designed to generate laser-induced self-organized patterns. By carefully controlling irradiation conditions, we aim to reveal how cumulative exposure drives the emergence of complex surface morphologies, offering a direct experimental framework to explore learning-like processes in materials.

A Coherent Legend Elite Series Ti:Sapphire laser system was employed to perform the experiments. This ultrafast amplifier delivers linearly polarized femtosecond pulses with a minimum pulse duration of approximately 40 fs, a repetition rate of 1 kHz, an average output power of 3 W, and a central wavelength of 800 nm. The pulses are first generated by the oscillator, subsequently amplified by the pump laser, and gated through a pulse picker, ensuring well-controlled irradiation conditions.

To trigger self-organization phenomena on the material surface, the system was configured for cross-polarized irradiation, implemented using a modified Mach–Zehnder interferometer. This configuration enables isotropic energy deposition, which is essential for promoting the development of complex nanostructures rather than anisotropic, polarization-driven ripples (such as standard LIPSS).

Experimental Setup

A schematic of the experimental configuration is shown in Fig. 4.2. The incoming laser beam is first divided by a non-polarizing beam splitter (BS1) with a 50/50 splitting ratio, producing two beams that propagate along separate optical arms. These beams are then recombined at a second beam splitter (BS2).

- **Temporal control:** The optical path length of one arm is finely adjusted by a motorized delay stage, which allows precise tuning of the relative arrival time between the two beams on the picosecond timescale. When the path lengths of the two arms are equal, a temporal overlap ($\Delta t = 0$) is achieved. This overlap is verified using an autocorrelator when both beams are set to the same polarization (0°).
- **Polarization control:** Each arm is equipped with a half-wave plate ($\lambda/2$) and a polarizer (P), enabling independent control over both the energy and polarization state of each beam. Cross-polarization is obtained when the polarization vectors of the two beams are oriented at 90° relative to one another. The condition of perfect cross-polarization is confirmed with an analyzing polarizer, allowing for rapid and precise alignment.
- **Beam delivery:** After recombination, the beams are collimated using a 25 cm focal length lens and focused onto the sample surface under normal incidence. The laser spot exhibits a Gaussian intensity distribution, with a beam diameter of approximately 60 µm (at $1/e^2$), determined experimentally using the D^2 method [73]. Careful calibration of the spot size ensures reproducibility and accurate fluence control during irradiation.

Formation of Nanostructures

The resulting laser–matter interaction leads to the development of self-organized nanostructures on the Ni(001) surface. Fig. 4.3 presents atomic force microscopy

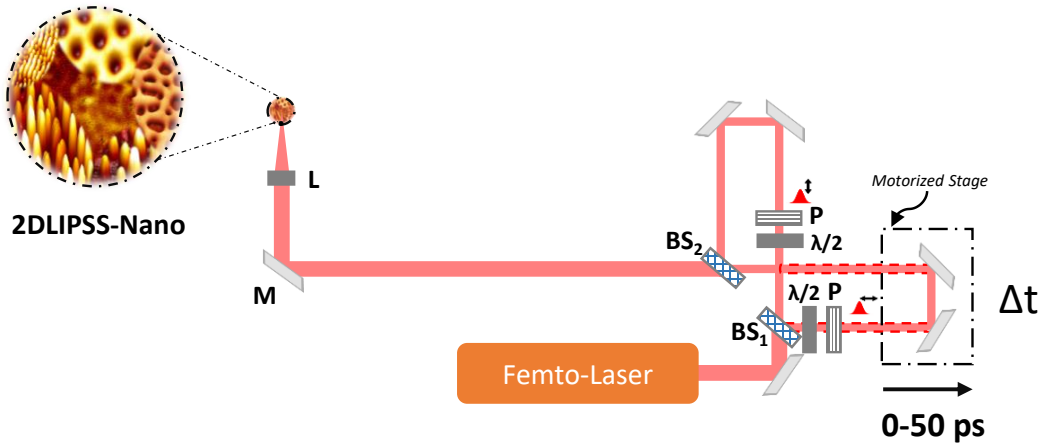


FIGURE 4.2: **Schematic illustration of a femtosecond laser double-pulse setup with temporal control.** BS1 and BS2 refer to the beam splitters, $\lambda/2$ represents the half-wave plate, P denotes the polarizer, M the mirror, and L the lens, which has a focal distance of 25 cm from the sample.

(AFM) images that provide complementary height-resolved information to the Scanning Electron Microscopy (SEM) observations.

While SEM provides high-resolution images of the surface morphology by detecting electrons emitted from the sample, it primarily offers two-dimensional contrast and requires conductive or coated samples. In contrast, AFM directly measures the surface topography using a nanoscale probe that scans the surface, producing true 3D height-resolved information. This allows AFM to quantify vertical features with sub-nanometer precision, complementing the SEM observations and giving a more complete characterization of the self-organized nanostructures.

AFM measurements are significantly more expensive and time-consuming compared to SEM imaging. Consequently, while SEM images were acquired for all surfaces, AFM characterization was only performed on a subset of samples. Nevertheless, AFM provides accurate height information, enabling precise estimation of nanostructure topography. This height data is then used to calibrate and infer the 3D structure from the 2D SEM images, allowing us to reconstruct the full surface morphology for further analysis.

AFM measurements reveal a rich variety of laser-induced surface morphologies. At lower fluences or shorter exposure, the surface initially develops nanopeaks, which are sharp, high-aspect-ratio protrusions that represent the tallest features observed. With increasing fluence or dose, these peaks evolve into interconnected nanowebs, forming filamentary networks across the surface. Under certain irradiation conditions, the system transitions instead into disordered but continuous nanolabyrinths, maze-like structures that extend over larger surface areas. Finally, localized nanocavities emerge, appearing as void-like depressions that punctuate the irradiated regions.

The emergence of these distinct morphologies is strongly dependent on the fluence (F) and the temporal delay (Δt) between the cross-polarized beams. As both parameters are varied, the system transitions between different regimes of self-organization. Increasing the cumulative laser dose further accentuates the structural evolution: nanostructures become progressively more defined, and their morphology reflects the interplay between ultrafast energy deposition, thermal and electronic dynamics, and laser-driven convection processes in the molten surface layer.

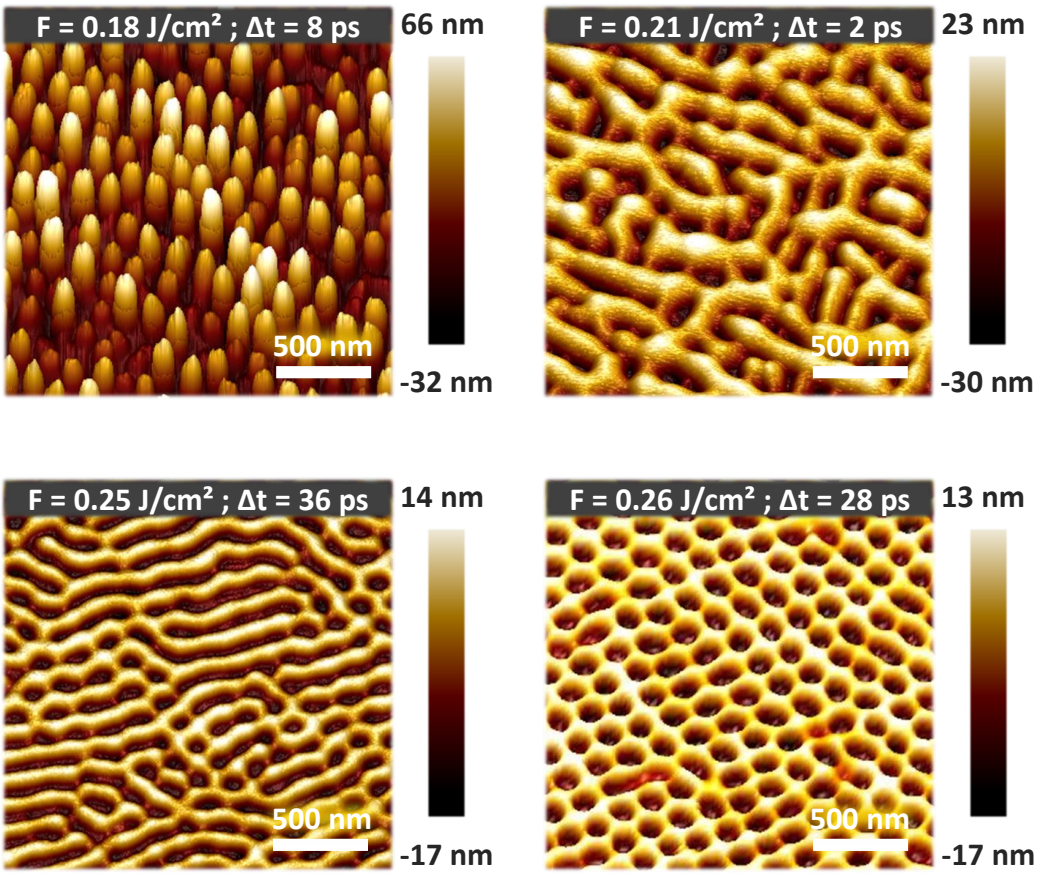


FIGURE 4.3: **Atomic Force Microscopy (AFM) images of laser-induced nanostructures formed on Ni(001)**. The structures form at different laser parameters with $N = 25$ double pulse sequences. The observed nanostructures include nanopeaks, nanowebs, nanolabyrinths, and nanocavities. These structures form progressively at different laser doses: (0.18 J/cm^2, 8 ps), (0.21 J/cm^2, 2 ps), (0.25 J/cm^2, 36 ps), and (0.26 J/cm^2, 28 ps). The height of these nanostructures varies from tens of nanometers up to approximately 100 nm.

Quantitatively, the heights of these nanostructures range from about 10 nm to 100 nm, with nanopeaks exhibiting the most pronounced vertical growth relative to their lateral size. These findings confirm that cross-polarized femtosecond irradiation provides a robust route to inducing complex three-dimensional self-organized patterns, far beyond the simpler periodic ripples typically associated with linearly polarized beams.

Patterns description

Fig 4.4 shows high-resolution SEM images of surface patterns, highlighting the evolution of two-dimensional nanostructures as a function of laser fluence F, temporal delay Δt, and pulse number N. While SEM provides a detailed view of surface morphology, the images primarily convey contrast linked to local intensity variations rather than absolute height. To approximate depth information, grayscale conversion is applied: the brightest pixels (value = 255) are mapped to the highest surface features (+1), while the darkest pixels (value = 0) are assigned to the deepest valleys (–1). This

approach allows a qualitative translation of the SEM images into three-dimensional height information.

In Series 1, corresponding to $F = 0.18$ J/cm^2 and $\Delta t = 8$ ps, the surface undergoes a progressive transformation with increasing pulse number. At very low doses ($N < 5$), the morphology is dominated by random nanoroughness, reflecting the early stages of surface modification. Between $5 \leq N \leq 10$, the system organizes into convection-cell patterns, which provide a template for subsequent self-organization. By $N = 25$, well-defined nanopeaks emerge, reaching their maximum concentration and aspect ratio. Atomic force microscopy confirms these peaks to be around 100 nm tall with diameters of about 20 nm. At this stage, the nanopeaks are preferentially located along the boundaries of the convection cells, suggesting a strong coupling between fluidic instabilities in the molten layer and nanoscale patterning. When the dose is further increased ($N > 30$), individual nanopeaks begin to merge, giving rise to elongated, linear features. Beyond $N = 40$, these merge further into intricate "dragon-like" motifs, marking a transition to chaotic structures. This sequence illustrates a cumulative feedback process, where each additional pulse interacts with a surface already reshaped by previous irradiation.

In Series 2, observed at $F = 0.21$ J/cm^2 and $\Delta t = 2$ ps, the surface follows a different evolutionary pathway. After the initial formation of convection cells, nanowebs appear for $N > 10$. These interconnected filamentary structures become increasingly pronounced, reaching their most organized state around $N = 20$. However, when the dose exceeds $N = 25$, the nanowebs lose coherence and disintegrate into chaotic structures, highlighting the destabilizing effect of excessive irradiation.

In Series 3, carried out at $F = 0.25$ J/cm^2 and $\Delta t = 36$ ps, the transition sequence is again distinct. At $N = 10$, the surface still exhibits convection cells. As the pulse number increases, these cells gradually deepen into nanocavities, which dominate the morphology for $N > 20$. At higher doses ($N \geq 35$), the cavities begin to connect laterally, giving rise to labyrinth-like networks. Interestingly, unlike the other series, the patterns in Series 3 do not degenerate into chaotic structures, even at high N, maintaining a relatively stable labyrinthine organization.

Finally, Series 4, corresponding to $F = 0.26$ J/cm^2 and $\Delta t = 28$ ps, produces another distinct trajectory. Already at $N = 10$, well-defined hexagonal nanocavities are observed, reflecting a higher degree of ordering from the onset compared to Series 3. With further irradiation ($25 \leq N \leq 35$), these cavities become increasingly regular, suggesting that certain fluence–delay combinations favor symmetry in self-organization. At still higher doses ($N > 45$), small nanobumps start to appear atop the existing cavities, resulting in chaotic hierarchical structures.

Overall, all series except Series 3 exhibit a transition to chaotic patterns at high pulse numbers, illustrating the delicate interplay between ordering mechanisms and destabilizing factors. The feedback mechanism, where each pulse modifies the evolving topography and thereby influences subsequent energy absorption, drives the system through different morphological regimes, from nanopeaks and nanowebs to nanocavities and labyrinths, and ultimately toward hierarchical, multi-scale architectures in the chaotic cases.

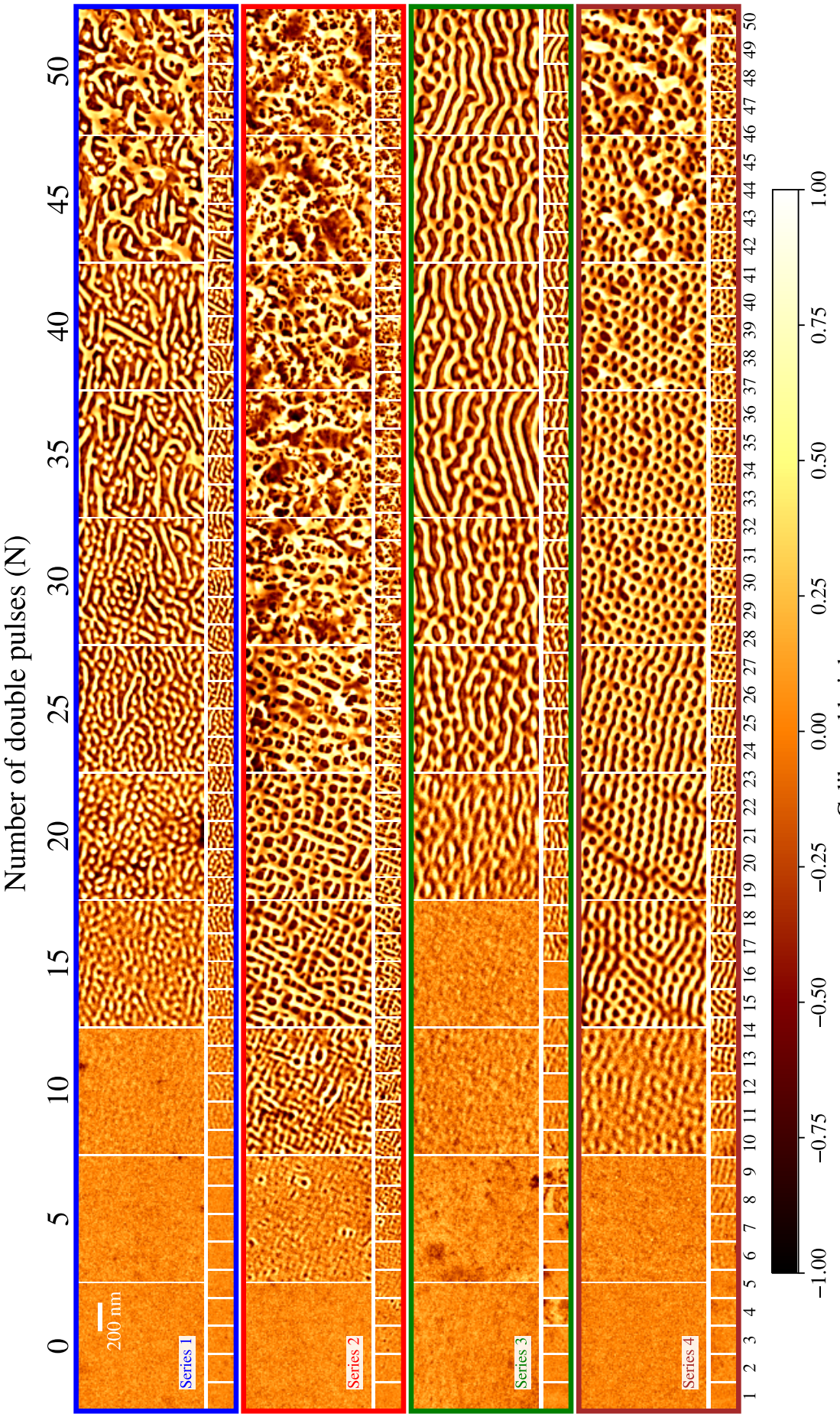


FIGURE 4.4: **SEM images of nickel surfaces irradiated with cross-polarized double pulses** N. Each series represents a progressive formation of self-organized patterns for distinct combinations of F and Δt: Series 1 corresponds to $F = 0.18\,\mathrm{J/cm^2}$ and $\Delta t = 8$ ps, Series 2 to $F = 0.21\,\mathrm{J/cm^2}$ and $\Delta t = 2$ ps, Series 3 to $F = 0.25\,\mathrm{J/cm^2}$ and $\Delta t = 36$ ps, and Series 4 to $F = 0.26\,\mathrm{J/cm^2}$ and $\Delta t = 28$ ps.

4.3 Numerical results

4.3.1 Correlation between Absorbed Energy and Complexity in Self-Organized Patterns

Self-organized surface patterns arise as a direct consequence of laser energy deposition: the absorbed energy E_{abs} produces localized heating, which drives material transport through pressure gradients and surface tension forces. Because the absorption is spatially inhomogeneous, feedback mechanisms naturally emerge. Regions that absorb more energy undergo stronger redistribution, while neighboring areas remain relatively stable. This imbalance amplifies small surface perturbations and ultimately leads to instabilities that evolve into well-defined patterns [106], marking the onset of self-organization.

To investigate how ultrafast light interacts with these evolving structures, we perform numerical simulations of Maxwell's equations for different polarization states. The surface morphology after N pulses is represented by the height map $u_N(x, y)$, which serves as the input for both absorption and complexity calculations. In this way, the electromagnetic field distribution is explicitly coupled to the evolving surface profile, showing how geometry controls the absorbed energy and, in turn, how this energy drives further pattern formation.

In the following, we interpret the emerging order as *complexity* $\mathscr{C}$, understood as the balance between randomness and structure that characterizes self-organized morphologies. To establish a quantitative link between absorbed energy and complexity, we introduce three complementary tools: complexity measures to distinguish ordered patterns from noise; kernel density estimation (KDE) to infer probability distributions directly from a single SEM snapshot; and mutual information to quantify the statistical dependence between E_{abs} and complexity $\mathscr{C}$. Together, these notions provide a framework to analyze how absorption and structural order reinforce each other during laser-induced self-organization.

Definition of complexity

Taylor Complexity is a measure of dynamical complexity that incorporates shape diversity information via the *derivative pattern distribution*, originally proposed by [15] in the context of machine learning for self-organization. The measure extends permutation entropy [5] and is inspired by statistical complexity introduced in [77]. It is straightforward to compute, interpret, and potentially applicable to broader domains.

We begin by noting that complexity is easy to identify but difficult to define. A number of complexity measures have been proposed in the literature, ranging from computational complexity, such as Kolmogorov complexity or Lempel-Ziv entropy; to information-theoretical measures such as Shannon entropy; measures of dynamical complexity in the sense of chaos, such as Lyapunov exponents or Kolmogorov-Sinai entropy, or fractal dimension; or a number of ad-hoc measures tailored to specific problems or fields.

The difficulty in defining a measure of complexity that is suitable to a given problem arises from the need to clarify the three different points: (i) *what exactly are we measuring the complexity of?* (ii) *what exactly do we mean by complexity?* (iii) *what measure are we going to use?* We shall examine each of these points in turn.

(i) Measuring complexity of self-organization dynamics from a single SEM image
We seek a measure of dynamical complexity that is applicable to the self-organization

process. Moreover, this measure should be estimated via a single snapshot of that dynamics given as an SEM image (cf. Fig. 4.4). Entropy alone would only quantify randomness, failing to distinguish structured patterns from noise, while order alone would overlook the diversity inherent in nontrivial structures. Complexity, by combining both order and disorder, is therefore required to capture the intermediate regime characteristic of self-organization: patterns that are neither perfectly ordered nor completely random. In this sense, complexity quantifies the richness of the structures that encode the dynamical process, providing a meaningful proxy for how much the system has "learned" through self-organization.

SEM images analyzed in this study are characterized by *patterns*, i.e., structures that display both regularity and diversity. Each structure emerges under quasi-uniform initial conditions through the same hydrodynamic self-organization process. To formalize this, we model an SEM image as a scalar field $u(\mathbf{x}, t)$ whose dynamics are governed by the local evolution law

$$\frac{\partial u}{\partial t}(\mathbf{x}, t) = F(u(\mathbf{x}, t)), \tag{4.1}$$

where F denotes a functional of the field u, its spatial derivatives, and possibly explicit dependencies on space $\mathbf{x}$ and time t. In this view, a single SEM image can be interpreted as a collection of patches, each representing the long-term outcome of slightly perturbed initial conditions and averaged boundary conditions, all evolving under the same dynamical rule. Because the evolution is driven by local interactions through F, and long-range interactions are absent, correlations between distinct structures in the image arise only as compositions of these local processes.

Under these assumptions, a measure of the *diversity* of the patches is a measure of sensitivity to initial conditions, in the sense of Lyapunov exponents. We are thus able to estimate dynamical complexity from a single SEM image, provided satisfying definitions of *pattern* and *complexity* are given.

(ii) Complex: diverse, but not too much In [77], the authors propose a novel statistical measure of complexity based on the following key insight: physical systems that are complex lie in an intermediate regime between two extremes: an ideal gas, which is perfectly disordered, and a crystal, which is perfectly ordered. A meaningful statistical measure of complexity should therefore combine a measure of *order* with a measure of *disorder*.

The most widely used measure of statistical disorder is *Shannon entropy* [112]. Consider a system X which can occupy one of N states with probabilities $P(X = i)$. Denoting $P(X = i) := p_i$ and collecting these values in the probability mass function $p = \{p_i\}_{i=1\ldots N}$, Shannon entropy is defined as

$$\mathscr{H}(p) = -K \sum_{i=1}^{N} p_i \log p_i, \tag{4.2}$$

where the constant K sets the base of the logarithm. In what follows, we set $K = 1$, so that entropy is measured in nats.

Equation (4.2) captures the *expected surprise* when sampling from a random variable X with probability distribution p. Entropy is continuous in p, reaches its maximum $\mathscr{H}(u) = \log N$ for the uniform distribution, and is unaffected by zero-probability states. Moreover, for two discrete random variables X and Y, entropy

satisfies the decomposition

$$\mathscr{H}(X,Y) = \mathscr{H}(X) + \sum_j P(Y=j)\,\mathscr{H}(X|Y=j), \tag{4.3}$$

where $\mathscr{H}(X,Y)$ is the joint entropy, $\mathscr{H}(X|Y=j)$ is the conditional entropy of X given $Y=j$, and $P(Y=j)$ is the probability that the random variable Y takes the value j.

Shannon entropy, first introduced in the context of Information Theory, has since become a foundational concept with applications across Physics, Machine Learning, and Biology. As Shannon himself remarked in [112], its defining properties are natural, in the sense that they correspond to intuitive expectations for a measure of statistical diversity. These four properties, known as the Shannon–Khinchin axioms [64], uniquely characterize entropy and thereby justify its status as the canonical measure of statistical disorder.

To complement a measure of disorder, one requires a measure of order. A simple choice is any decreasing function of entropy, such as $(1-\mathscr{H})$, so that a trivial example of a complexity measure would be the product $\mathscr{H}(1-\mathscr{H})$, which depends only on $\mathscr{H}$. However, the choice of order functional is crucial: it determines which probability distributions maximize complexity and thus carries important semantic meaning.

In [77], the authors propose *disequilibrium* as a measure of order, defined as the squared distance from the uniform distribution:

$$\mathscr{D}(p) = \sum_{i=1}^{N} \left(p_i - \frac{1}{N}\right)^2. \tag{4.4}$$

This definition is motivated by expanding entropy H around the uniform distribution, which maximizes entropy. For a perturbation of size δp we obtain

$$\mathscr{H}(p) = \mathscr{H}_{\max} - \frac{N}{2}\sum_{i=1}^{N}\left(\delta p_i - \frac{1}{N}\right)^2 + o(\|\delta p\|^2).$$

Hence, up to second order in the perturbation, disequilibrium coincides with the trivial choice of order functional, since

$$\mathscr{D}(p) \;\approx\; 1 - \mathscr{H}(p) + o(\|\delta p\|^2).$$

Combining Shannon entropy (Eq. 4.2) with disequilibrium (Eq. 4.4), López-Ruiz, Mancini, and Calbet defined the statistical complexity measure now known as *LMC complexity*:

$$C = \mathscr{H} \times \mathscr{D}. \tag{4.5}$$

We adopt this measure in the present report due to its intuitive interpretation, ease of computation, and demonstrated versatility across a wide range of applications, including image processing, fundamental physics, and biomedical systems [77]

(iii) Taylor Complexity Self-organized surface patterns exhibit not only variations in intensity but also characteristic local slopes and curvatures that are crucial to their structure. Standard LMC complexity only captures the variability of pixel values, neglecting this geometric information. To better quantify the structural diversity of such patterns, we use **Taylor complexity**, an extension of LMC that incorporates local derivatives. Taylor complexity thus captures both the variability of pixel values

and the contribution of local slopes and curvatures, providing a more complete description of self-organized geometrical structures.

Given a field $u(x, y)$ representing a self-organized pattern, we examine all overlapping 3×3 patches of the image. This patch size is chosen as the minimal neighborhood that allows meaningful evaluation of local slopes and curvatures, while keeping the number of possible sign patterns manageable. Using larger patches, such as 5×5, would dramatically increase the number of possible configurations (2^{25}), leading to sparse statistics and reduced sensitivity to fine structural details.

For each patch, we first subtract its mean intensity so that only local deviations remain. Each pixel in the patch is then assigned a binary sign $\{+, -\}$ depending on whether its value is above or below the mean, yielding a **sign-pattern field** Pu with 2^9 possible configurations per patch.

To include local geometric information, we extend this approach to derivatives of u. Specifically, we construct sign-pattern fields Pu_x and Pu_y from first-order derivatives, and Pu_{xx}, Pu_{xy}, and Pu_{yy} from second-order derivatives.

The **Taylor complexity** $\mathscr{C}$ is computed as the second-order Taylor expansion of LMC complexity and is given by

$$\begin{aligned}\mathscr{C} = {} & C(Pu) + C(Pu_x) + C(Pu_y) \\ & + \frac{1}{2}C(Pu_{xx}) + C(Pu_{xy}) + \frac{1}{2}C(Pu_{yy}),\end{aligned} \tag{4.6}$$

where $C(\cdot)$ denotes the LMC complexity computed over the distribution of all 3×3 patches for each field or derivative. First-order derivatives capture local slopes, while second-order derivatives capture curvature, providing a richer description of local structures.

We use a second-order Taylor expansion because the field itself, its slopes, and its curvature capture the majority of local geometric variations in self-organized patterns. Higher-order derivatives contribute progressively less at the scale of a single patch and would increase computational complexity without substantially improving sensitivity to relevant structures.

For simplicity, we refer to $\mathscr{C}$ as **complexity** throughout the manuscript. This measure simultaneously encodes the distribution of local values and the geometry of local features, making it particularly well-suited for analyzing SEM images of self-organized surfaces. Additional derivations and implementation details are provided in Appendix A.1.

Figure 4.5 illustrates the relationship between complexity $\mathscr{C}$, Shannon entropy $\mathscr{H}$, and disequilibrium $\mathscr{D}$. Different regions correspond to characteristic physical states: low entropy and high disequilibrium describe the initial flat surface, high entropy and low disequilibrium represent disordered or chaotic morphologies, while intermediate values of both, which maximize $\mathscr{C}$, correspond to well-organized, self-assembled patterns emerging after irradiation.

Definition of Absorbed Energy

As introduced in the previous chapter, the absorbed energy E_{abs} quantifies the total amount of electromagnetic energy transferred from the incident laser field into the material. It provides a macroscopic measure of how much energy is deposited into the surface after each laser pulse, taking into account both the evolving surface morphology and the polarization state of the incoming light.

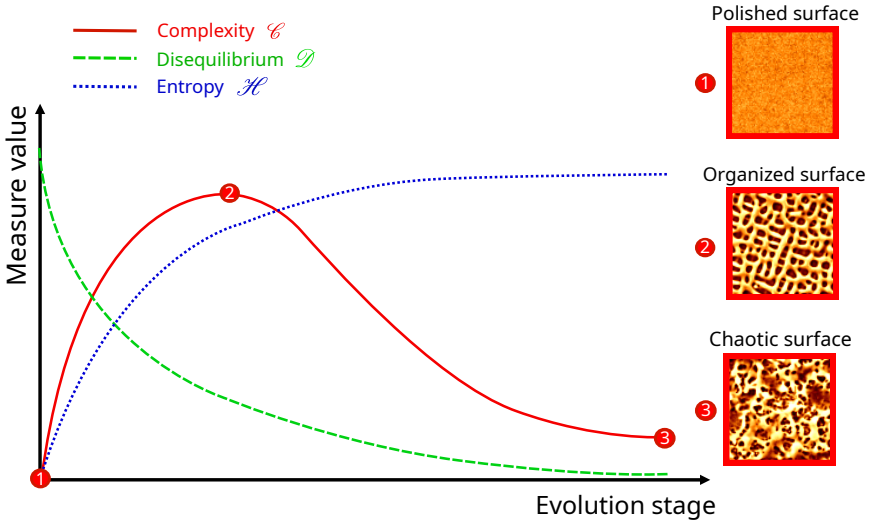


FIGURE 4.5: Schematic illustration of complexity $\mathscr{C}$, Shannon entropy $\mathscr{H}$, and disequilibrium $\mathscr{D}$, highlighting how their combination determines overall structural complexity in physical systems. The amplitudes of the depicted quantities are exaggerated for illustrative purposes.

Here, polarization refers to the orientation of the electric field vector of the laser beam. Since the interaction between the laser and the surface strongly depends on this orientation, E_{abs} is evaluated as the mean absorbed energy over six linear polarization angles, $\alpha = k\pi/6$ with $k = 0, \ldots, 5$. For each polarization and surface morphology $u_N(x, y)$, the absorbed energy is obtained by integrating the two-dimensional absorption distribution $\varepsilon_{\text{abs}}(x, y)$ over the entire irradiated area. This distribution represents the cumulative energy deposited per unit depth into the surface.

At the microscopic scale, the local absorption density is governed by the material's optical response. Specifically, the instantaneous absorption at position (x, y, z) and time t is given by

$$\varepsilon_{\text{abs}}(x, y, z, t) = \tfrac{1}{2}\, c\epsilon_0 \,\frac{4\pi\, \Im(\tilde{n})}{\lambda}\, \Re(\tilde{n})\, \left|\tilde{E}(x, y, z, t)\right|^2, \tag{4.7}$$

where λ is the free-space wavelength, $\tilde{n}$ is the complex refractive index, $\Re(\tilde{n})$ governs refraction, $\Im(\tilde{n})$ (the extinction coefficient) controls absorption, and $\tilde{E}$ is the electric field amplitude. The constants ϵ_0 and c denote, respectively, the vacuum permittivity and the speed of light in free space.

To connect this local picture to the macroscopic absorbed energy, we integrate over depth and time:

$$\varepsilon_{\text{abs}}(x, y) := \iint_{z,t} \varepsilon_{\text{abs}}(x, y, z, t)\, dz\, dt, \tag{4.8}$$

resulting in a two-dimensional absorption map $\varepsilon_{\text{abs}}(x, y)$ that encodes the spatial distribution of cumulative absorption.

Measure of correlation

Before quantifying the relationship between absorbed energy and structural complexity, it is important to establish a robust statistical framework. Direct inspection

of raw data may overlook subtle or nonlinear dependencies, while simple correlation measures are limited to linear relationships. To capture the full range of interactions between these two quantities during the learning process, we employ information-theoretic approaches. Specifically, we estimate the underlying probability distributions using kernel density estimation (KDE) and then compute mutual information (MI) to measure the strength of statistical dependence. This combination provides a data-driven and flexible approach to analyze how absorption and complexity co-evolve across successive irradiation pulses.

(i) Kernel Density Estimation Mutual information quantifies the statistical dependence between two variables, but its computation requires knowledge of their probability distributions. Since we only have a finite set of data points obtained from SEM images, these distributions are not available in closed form. To address this, we use kernel density estimation (KDE), a widely used non-parametric method for estimating probability density functions directly from data [92]. Unlike parametric approaches, which assume a specific functional form for the distribution, KDE infers the distribution from the data itself. This flexibility makes it particularly effective for capturing complex, multi-modal, or irregular distributions and allows us to compute mutual information in a consistent, data-driven manner.

The central idea of KDE is to represent the probability density as a sum of smooth kernel functions, each centered at a data point. For a dataset $\{x_1, x_2, \ldots, x_M\}$ consisting of M samples, where each data point $x_N = [E^N_{\text{abs}}, \mathscr{C}^N]^T$ encodes the absorbed energy E^N_{abs} and the complexity $\mathscr{C}^N$ after N double pulses, the estimated density at a point $\mathbf{x} = [E_{\text{abs}}, \mathscr{C}]^T$ is given by:

$$\hat{f}(\mathbf{x}) = \frac{1}{M} \sum_{N=1}^{M} K(\mathbf{x} - x_N), \tag{4.9}$$

where M is the total number of observed configurations (one per pulse number) and $K(\mathbf{x})$ is a kernel function that determines the local contribution of each sample to the density estimate.

In this work, we use a bivariate Gaussian kernel, defined as:

$$K(\mathbf{x}) = \frac{1}{2\pi\sqrt{\det(H)}} \exp\left(-\frac{1}{2}\mathbf{x}^T H^{-1} \mathbf{x} \right), \tag{4.10}$$

where H is a 2×2 positive-definite bandwidth matrix. The matrix H controls the shape and width of the kernel, effectively determining the degree of smoothing: larger eigenvalues of H produce wider kernels and a smoother density estimate, while smaller eigenvalues allow finer details to be resolved.

The KDE approach thus constructs a smooth, continuous approximation of the underlying probability distribution by summing localized contributions from each data point. It is particularly useful for visualizing and analyzing complex datasets, identifying modes, and estimating probabilities without assuming a restrictive parametric model. By providing a continuous estimate of the joint distribution of absorbed energy and complexity, KDE enables a reliable computation of mutual information.

(ii) Mutual information Mutual information (MI) [66] is a fundamental information-theoretic measure that captures both linear and non-linear dependencies between two random variables. Intuitively, it quantifies how much knowing the value of

one variable reduces the uncertainty about the other. A higher mutual information value corresponds to stronger statistical dependence, whereas a value of zero implies complete independence between the variables. Unlike correlation, which is restricted to detecting linear relationships, mutual information is sensitive to more general interactions, making it particularly useful for analyzing complex systems.

Since mutual information requires knowledge of the joint and marginal probability distributions of the variables, we estimate these distributions using kernel density estimation (KDE) as introduced in the previous subsection. KDE provides a smooth, continuous approximation of the probability densities, allowing for a reliable and non-parametric computation of MI from our SEM data.

Formally, for two continuous random variables, the absorbed energy E_{abs} and the complexity $\mathscr{C}$, the mutual information can be expressed in terms of probability distributions [30] as:

$$MI(E_{\text{abs}};\mathscr{C}) = \sum_{e,c} P_{E_{\text{abs}}\mathscr{C}}(e,c) \log\left(\frac{P_{E_{\text{abs}}\mathscr{C}}(e,c)}{P_{E_{\text{abs}}}(e)\,P_{\mathscr{C}}(c)}\right), \tag{4.11}$$

where $P_{E_{\text{abs}}}(e)$ and $P_{\mathscr{C}}(c)$ denote the marginal probability densities of E_{abs} and $\mathscr{C}$, respectively, and $P_{E_{\text{abs}}\mathscr{C}}(e,c)$ is their joint probability density. If the two variables are independent, then $P_{E_{\text{abs}}\mathscr{C}}(e,c) = P_{E_{\text{abs}}}(e)P_{\mathscr{C}}(c)$, and the mutual information is zero. Any departure from independence leads to a positive MI, reflecting the amount of shared information between the two quantities.

An equivalent and often more intuitive formulation uses the concept of entropy. Let $\mathscr{H}(X)$ denote the Shannon entropy of a random variable X, and $\mathscr{H}(X,Y)$ their joint entropy. Mutual information can then be written as:

$$MI(E_{\text{abs}};\mathscr{C}) = \mathscr{H}(E_{\text{abs}}) + \mathscr{H}(\mathscr{C}) - \mathscr{H}(E_{\text{abs}},\mathscr{C}), \tag{4.12}$$

which highlights its interpretation as the reduction in uncertainty of one variable due to knowledge of the other. In this view, MI represents the overlap of information content between E_{abs} and $\mathscr{C}$.

In practice, mutual information is computed for $N \in [2,50]$, since at least three observations are required in each dataset for a meaningful estimation. This restriction ensures statistical reliability when quantifying the evolving dependency between absorbed energy and complexity as the system undergoes successive irradiation pulses.

Complexity drives absorption and absorption fuels complexity

In order to investigate how ultrafast light interacts with these evolving structures, we perform numerical simulations of Maxwell's equations for different polarization states. The surface morphology is represented by the height map $u_N(x,y)$, corresponding to the topography after N laser pulses. This approach allows us to explicitly capture the coupling between the electromagnetic field distribution and the surface profile, thereby revealing how the geometry of the surface influences energy absorption and how this, in turn, drives further self-organization.

(i) Evolution of absorption and complexity under repeated pulses Fig. 4.6 shows how absorbed energy and complexity evolve together as the number of laser pulses, N, increases. The absorbed energy, E_{abs}, quantifies the energy taken up by the surface at the N-th pulse, averaged over six polarization states $\alpha = k\pi/6$ with

$k = 0, \ldots, 5$. For each surface morphology $u_N(x, y)$, it is computed by integrating the local absorption over the entire surface.

In parallel, we compute the *Taylor complexity* $\mathscr{C}$ of each surface morphology $u_N(x, y)$, a statistical measure of structural complexity [15]. Taylor complexity is defined as a second-order Taylor expansion of the LMC complexity [77] over the sign-pattern fields of the surface and its derivatives, as given in Eq. (4.6). Unlike standard LMC complexity, which is computed only from pixel intensity distributions, Taylor complexity incorporates local derivatives to capture variations in slopes and curvatures in addition to amplitude fluctuations.

The evolution of absorbed energy E_{abs} and complexity $\mathscr{C}$ with the number of double pulses N reveals a close correspondence between energy uptake and the development of surface structure, as illustrated in Fig. 4.6. To enable meaningful comparisons across conditions, absorbed energy is normalized by the energy absorbed by a flat surface, and complexity is normalized to range from 0 to 1. This normalization ensures that both quantities reflect relative changes rather than absolute magnitudes.

Initially, for small N, both normalized E_{abs} and $\mathscr{C}$ remain approximately constant, reflecting the absence of clear surface patterns. The material has yet to respond favorably to the laser irradiation so the surface morphology remains almost uniform.

As N increases beyond a certain threshold, self-organized structures begin to emerge. The pattern formation is accompanied by a gradual increase in both absorbed energy and complexity. The SEM images above the curves illustrate this progression, showing how initially small and irregular topographical features evolve into well-defined, organized patterns. In this regime, higher complexity correlates with enhanced absorption, indicating that the formation of structured surface features actively facilitates more efficient interaction with the incident laser field.

Eventually, for most series, the organized patterns reach a maximum before gradually breaking down, marking the emergence of chaotic structures. During this stage, absorbed energy typically decreases as the coherent surface features that previously promoted efficient energy uptake become disrupted. The behavior of complexity in this regime is more series-dependent: in some cases, $\mathscr{C}$ decreases alongside absorption as the system loses order, while in others, local disorder can temporarily increase $\mathscr{C}$ even as global pattern coherence diminishes. Notably, in series 3, no chaotic regime is observed, and both absorbed energy and complexity continue to increase, reflecting persistent self-organization without structural breakdown.

Overall, these curves highlight a strong relationship between surface morphology, complexity and energy absorption. The initial plateau indicates an inert surface, the synchronized increase of E_{abs} and $\mathscr{C}$ captures the constructive formation of ordered patterns, and the eventual divergence in behavior at the end illustrates how disruption of self-organized structures can decouple absorption from complexity.

The observed correlation between absorbed energy and complexity suggests that self-organized surface structures emerge primarily to enhance energy absorption. As the number of laser pulses increases, the development of ordered patterns corresponds to a rise in E_{abs}, indicating that the formation of these structures facilitates more efficient coupling between the incident electromagnetic field and the material. Conversely, when the organized features begin to break down and disorder sets in, the energy absorption decreases, reflecting the loss of coherent structures that previously optimized light-matter interaction. This behavior highlights the adaptive nature of pattern formation: the surface morphology evolves in a manner that maximizes energy uptake during the self-organization process, while the disappearance of these structures reduces this efficiency.

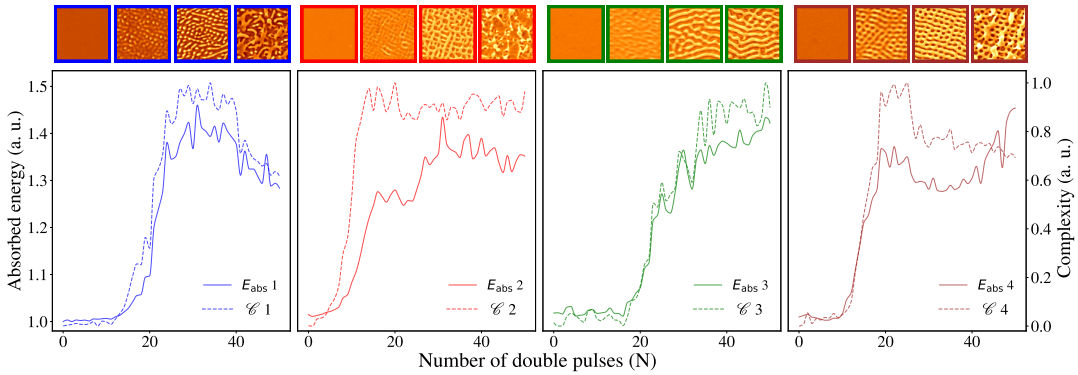


FIGURE 4.6: **Absorbed energy E_{abs} and to complexity $\mathscr{C}$ versus the number of double pulses N.** E_{abs} (solid line) and $\mathscr{C}$ (dashed line) as a function of N, ranging from 0 to 50 for the four series. Representative SEM images above the curves illustrate the corresponding evolution of surface patterns with increasing irradiation.

More broadly, these observations can be understood in the framework of *structural learning*. In complex adaptive systems, structural learning refers to the process through which a system gradually modifies its internal organization in response to repeated inputs. Unlike simple learning, which may only involve changes in magnitude (for instance, a stronger or weaker response), structural learning alters the configuration of the system itself, embedding information into its very architecture. This allows the system to develop a form of memory, enabling it to anticipate, adapt and stabilize its response to external stimuli over time. Structural learning thus provides an explanation for the surface reorganization, where patterns are not imposed externally but instead emerge from within the system as a consequence of its own dynamics.

In the context of laser–matter interaction, structural learning manifests as the progressive modification of surface morphology under repeated ultrafast excitation. Each laser pulse perturbs the surface energy distribution, leading to subtle reconfigurations in the material structure. These reconfigurations alter the way subsequent pulses are absorbed, creating a feedback loop between morphology and light–matter coupling. Over many iterations, this feedback drives the surface toward self-organized states that encode the interaction history in their geometry. In this sense, the material effectively "learns" from the laser input by storing information in its structure, stabilizing recurring patterns and eventually reaching a limit where adaptation can no longer be sustained. This perspective allows us to interpret laser-induced self-organization not only as a physical instability but also as a form of structural learning, where memory, adaptation and breakdown emerge naturally from the underlying dynamics.

The dynamics of absorbed energy and complexity unfold in four successive regimes, which are depicted in Fig. 4.8.a. These regimes illustrate how the surface undergoes a *learning process* as it adapts to repeated ultrafast laser excitation.

- **Phase I: Response.** The process begins with an exploratory stage where the surface first reacts to the incoming pulses. Small local perturbations appear in the topography but hydrodynamic instabilities are not yet sustained. Both E_{abs} and $\mathscr{C}$ remain almost unchanged, reflecting an adaptation phase comparable to an initial learning attempt.

- **Phase II: Iterative learning.** A sudden and simultaneous increase in absorbed energy and complexity marks the beginning of an active adaptation. Here,

relationship between the surface morphology and the incident light becomes pronounced: surface features enhance absorption, which in turn reinforces further structural development. During this stage, self-organization progressively stabilizes hydrodynamic instabilities and the system locks into recurrent patterns. The dynamics resemble an iterative learning process, where the system refines its response through repeated exposure. By the end of this phase, the patterns stabilize into a persistent configuration, effectively leaving a permanent signature on the material surface.

- **Phase III: Memory stabilization.** After self-organization is achieved, the system consolidates its adaptive state. Absorbed energy no longer increases substantially, while complexity reaches a plateau, reflecting a form of memory storage within the surface morphology. The system preserves its structured patterns and becomes resilient to perturbations, analogous to memory consolidation in a learning process.

- **Phase IV: Destruction.** With prolonged irradiation, the system enters a breakdown stage. Absorbed energy continues to rise, but complexity declines, signaling a decoupling between energy capture and structural order. The established self-organized patterns are progressively destroyed as instabilities exceed the system's capacity for stable organization. This stage resembles overlearning, where excessive input erodes previously consolidated memory and leads to disordered, chaotic behavior.

Overall, these four phases illustrate how the surface morphology evolves in a feedback-driven manner, reflecting a form of structural learning. The rise of both absorbed energy and complexity during Phases II and III highlights the adaptive role of self-organization: patterns emerge and stabilize precisely to optimize energy uptake. Conversely, the breakdown observed in Phase IV emphasizes the limits of this adaptation, where structural order can no longer sustain efficient absorption. This progression shows that the development and decay of surface patterns are closely linked to changes in energy absorption, motivating the need for quantitative measures such as mutual information to rigorously capture the correlation between complexity and absorbed energy throughout the learning process.

(ii) Measure of structural learning To characterize the system's evolution prior to the onset of chaotic structures, the data distribution is analyzed within the learning regime, focusing on the relationship between absorbed energy E_{abs} and complexity $\mathscr{C}$. As the number of double pulses N increases, the system eventually transitions from the learning to the chaotic regime. The transition point is defined as the value of N marking the onset of this shift. For kernel density estimation (KDE), only data points up to this transition are considered, namely $x_1, x_2, \ldots, x_N$, with $x_N = [E_{\text{abs}}^N, \mathscr{C}^N]^T$, where E_{abs}^N and $\mathscr{C}^N$ denote, respectively, the absorbed energy and complexity after N double pulses.

For intuitive visualization, the KDE contours and colormap are adjusted to reflect mass distribution. As shown in Fig. 4.7, the probability mass is divided into five levels ranging from 0–20% to 80–100%, with colors progressing from light to dark blue. Across the four series, the resulting distributions display a similar structure: contours are spread along a diagonal axis, indicating that higher values of E_{abs} are generally accompanied by higher values of $\mathscr{C}$. The spread is broad, showing no concentration in a single region.

This diagonal alignment reveals a positive correlation between E_{abs} and $\mathscr{C}$ in the learning regime. As surface complexity increases, self-organization becomes more pronounced, and the patterns grow more structured. This reorganization enhances the material's ability to absorb energy, showing that absorption is not random but governed by an adaptive mechanism that promotes order prior to the onset of chaos.

To quantify this relationship, the same kernel density estimation (KDE) framework used to represent the joint distribution of $(E_{\text{abs}}, \mathscr{C})$ is employed to compute the mutual information (MI) between the two variables. Fig. 4.8.b illustrates how the mutual information (MI) evolves between the absorbed energy, E_{abs}, and the complexity, $\mathscr{C}$, as the number of double pulses, N, increases. Mutual information is particularly well suited for this analysis because it does not restrict the dependency measure to linear correlations. Instead, it quantifies both linear and non-linear relationships, offering a more complete description of how energy absorption and structural complexity are coupled during laser matter interaction. A high MI value reflects a strong dependency between the two variables, whereas a low MI indicates that they evolve independently. In this way, MI functions as a statistical lens through which the system's ability to "learn" structurally from repeated laser exposure can be evaluated. To enable comparisons across conditions, MI values are normalized between 0 and 1, where 1 corresponds to the strongest dependency observed.

From the perspective of structural learning, E_{abs} represents the system's input signal (the amount of energy injected), while $\mathscr{C}$ reflects the organized outcome encoded in the surface morphology. MI then quantifies how much of the input is effectively translated into meaningful structural change. When MI is low, the system is unable to use the absorbed energy in a consistent way to reorganize its surface, which corresponds to the earliest stage of learning. When MI rises, it indicates that energy absorption and complexity are becoming tightly coupled: the surface is actively transforming energy intake into increasingly ordered patterns. The stabilization of MI corresponds to a regime in which the system has "memorized" an efficient mapping from energy absorption to pattern formation, maintaining order across further excitations. Finally, a decline in MI signals the breakdown of this learned relationship. Although energy is still being absorbed, it no longer translates into coherent structures but instead generates disorder. This final stage reflects a failure of structural learning, where previously acquired organization is lost.

The temporal profile of MI exhibits a characteristic trajectory that reflects the dynamics of structural learning. For very small N, MI remains low and constant. This corresponds to the *response phase* (I), where the system reacts passively to incoming energy without establishing a consistent structure. As N increases, MI rises sharply, marking the onset of the *iterative learning phase* (II), in which absorbed energy and complexity become increasingly coupled and self-organization intensifies. Eventually, MI reaches a plateau, which signifies the *memory stabilization phase* (III). Here, the system has established a robust mapping between energy absorption and structural outcomes, maintaining coherent patterns over successive pulses. In most cases, MI eventually declines, reflecting the transition to the *destruction phase* (IV). In this regime, the system "forgets": absorbed energy no longer leads to predictable increases in complexity but instead produces chaotic or irregular structures.

Notably, one dataset (series 3) does not exhibit this decline. Instead, MI continues to increase, indicating that chaotic patterns never emerge and that the correlation between absorbed energy and complexity remains strong. This persistence suggests that in some regimes, structural learning can proceed without eventual breakdown, pointing to conditions under which the system maintains organization indefinitely.

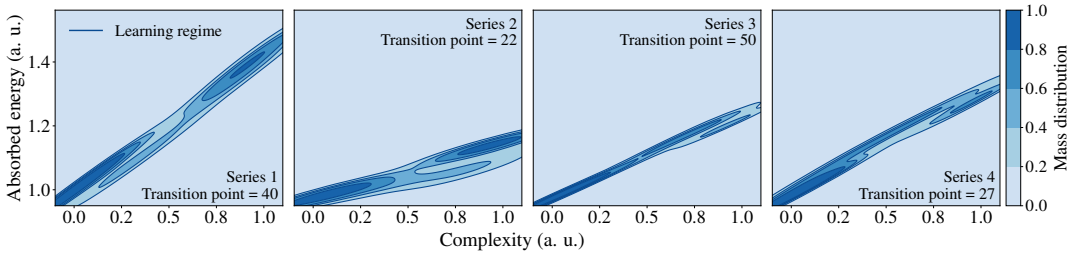


FIGURE 4.7: **Estimation of data distribution during learning regime**. Each column represents different series, with the transition point indicating the number of data points used in distribution estimation. The contour divides the distribution into five regions, covering mass ranges from $0-20\%$ to $80-100\%$.

In summary, mutual information offers a quantitative link to the framework of structural learning. Its increase, plateau, and eventual decline correspond closely to the stages of learning: initial adaptation, active pattern acquisition, stabilization of structural memory, and eventual breakdown. Viewed in this way, laser–matter interaction emerges as a process in which the surface effectively "learns" from repeated excitation, encodes information into organized patterns, and, depending on the irradiation conditions, either preserves or loses this acquired structural memory.

4.3.2 Effect on polarization angle on absorbed energy

To clearly separate the global response of the system, which captures how the presence of self-organized patterns influences overall energy absorption, from the specific response, which isolates the effect of laser polarization, the absorbed energy, E_{abs}, is analyzed as a function of the polarization angle, α. This distinction is crucial: the global response reflects the structural memory encoded in the surface, whereas the specific response highlights how this memory is accessed or activated by a particular light stimulus. If the surfaces have effectively "learned" from previous irradiation, a distinct response is expected for the polarization states $\alpha = 0$ and $\alpha = \pi/2$, corresponding to the orientations used experimentally to generate the self-organized patterns under study.

(i) Polarization as a privileged stimulus in structural learning As illustrated in Fig. 4.9, four distinct self-organized surface patterns are considered. To ensure that comparisons across polarization angles remain unbiased, we restrict ourselves to circular pattern shapes. This guarantees that the effective absorbing surface area is identical under any rotation, preventing artifacts that could arise if certain structures were present or absent depending on orientation. For each pattern, we then compute six absorption maps corresponding to polarization angles $\alpha = k\pi/6$, with $k = 0, 1, \ldots, 5$. The polarization states $\alpha = 0$ and $\alpha = \pi/2$ are particularly relevant for interpretation because the self-organized patterns were generated using cross-polarized double pulses at these orientations. These angles effectively define the directions along which the surface structure was initially driven into order, making them privileged stimuli for analyzing the system's response. They can be regarded as the experimental "training directions" through which the system was initially driven into order, and thus act as privileged stimuli in the structural learning process.

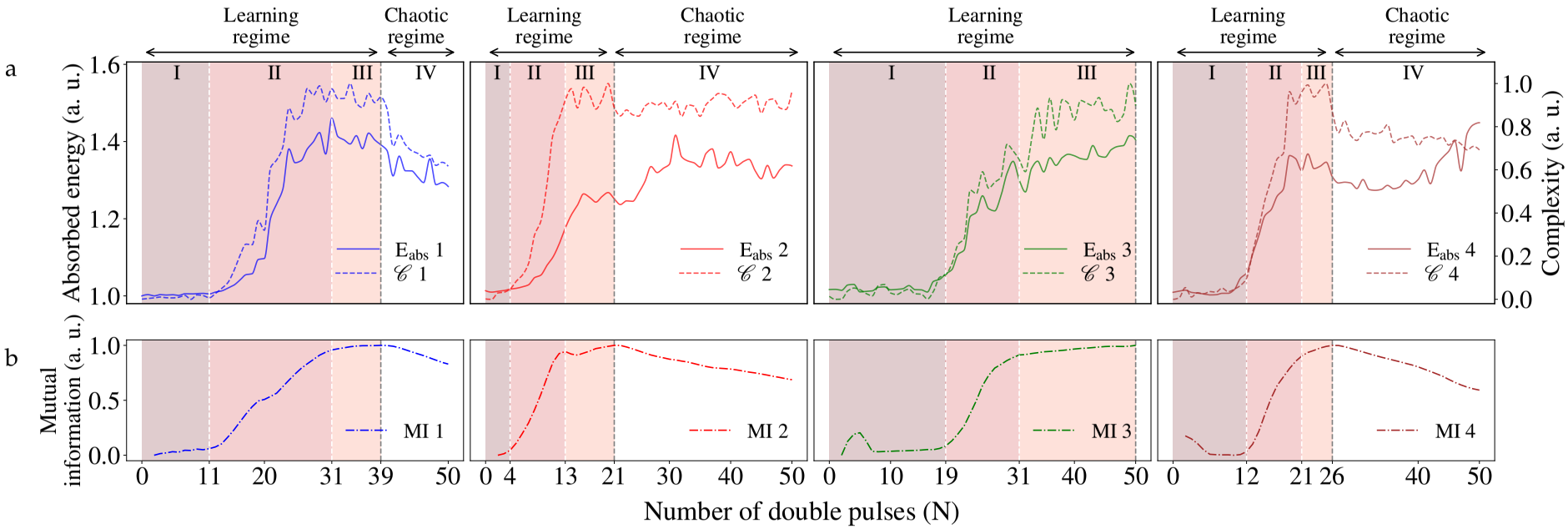


FIGURE 4.8: **Absorbed energy E_{abs} correlated to complexity $\mathscr{C}$. (a)** E_{abs} (solid line) and $\mathscr{C}$ (dashed line) as a function of N, ranging from 0 to 50 for the four series. The two variables go through 4 phases that reflect two regimes. The learning regime comprises: response phase (I), iterative learning phase (II), and memory stabilization phase (III). The chaotic regime is equivalent to the destruction phase (IV). **(b)** The mutual information MI variation is the criterion used to split E_{abs} and $\mathscr{C}$ into 4 phases.

For any given experimental series, after N double pulses, the resulting absorption maps exhibit marked differences depending on how the orientation of the self-organized structures aligns with the incoming electric field, $\vec{E}$. When the surface features are parallel to the polarization direction, absorption is maximized: sharp peaks with high curvature act as localized hotspots, capturing more energy. Conversely, valleys or regions misaligned with $\vec{E}$ absorb significantly less. This anisotropic response illustrates how the surface morphology acts as a "filter" that preferentially couples to specific polarization states, a hallmark of how the system encodes directional information through self-organization.

The polarization dependence becomes even more evident when examining how E_{abs} evolves with the number of double pulses, N. Across all polarization angles, the absorption curves exhibit a qualitatively similar progression, yet they remain distinctly separated over the entire range of N values (from 0 to 50). Initially, in the absence of pronounced surface structures, all curves overlap and remain essentially flat: no significant difference exists between polarization states because the surface lacks organization. As N increases, however, the absorption curves diverge. This separation directly reflects the emergence of oriented structures: once the surface develops a memory of orientation, the absorbed energy becomes strongly dependent on whether the polarization of the incoming light matches or opposes that orientation.

Among the various polarization states, the experimental angles $\alpha = 0$ and $\alpha = \pi/2$ play a special role precisely because the patterns themselves were generated by repeated stimulation at these orientations. Their corresponding absorption curves consistently form the upper and lower bounds of the other polarization responses. Interestingly, which one acts as the upper bound (strongest absorption) and which serves as the lower bound (weakest absorption) is not universal but varies between experimental series. This variability highlights the system's sensitivity to initial conditions and history, which are central features of self-organizing processes.

Taken together, these results emphasize the concept of stimulus specificity in laser–matter interaction. The efficiency of energy absorption depends not only on the global presence of patterns but also on how well the incoming stimulus aligns with the structural memory imprinted on the surface. When the polarization of a new pulse is consistent with the orientation established by the previous $N-1$ pulses, the system exhibits a highly specific and enhanced response: absorption is targeted and concentrated along the structural alignment, reinforcing the existing order. In contrast, when the polarization differs from the trained orientation, the stimulus is less specific. Absorption then becomes more evenly distributed, leading to weaker reinforcement of existing patterns.

This polarization-sensitive behavior can be interpreted as a form of structural recognition or selective response. The surface does not simply absorb energy passively; it evaluates the incoming stimulus in relation to its prior "learning history." Stimuli that are congruent with the established structural memory elicit a stronger, more efficient response, while incongruent stimuli dilute the structural coherence of absorption. This interplay between specificity and generality mirrors fundamental aspects of structural learning theory, where a system adapts not only to external inputs but also to their consistency with previously acquired organizational rules.

4.3.3 Effect of perturbating a pattern

In order to maximize the absorbed energy, the material surface undergoes self organization, progressively adjusting its morphology. This evolution results in variations of structural complexity, denoted $\mathscr{C}$, that ultimately stabilize into an optimal pattern $\mathcal{P}$.

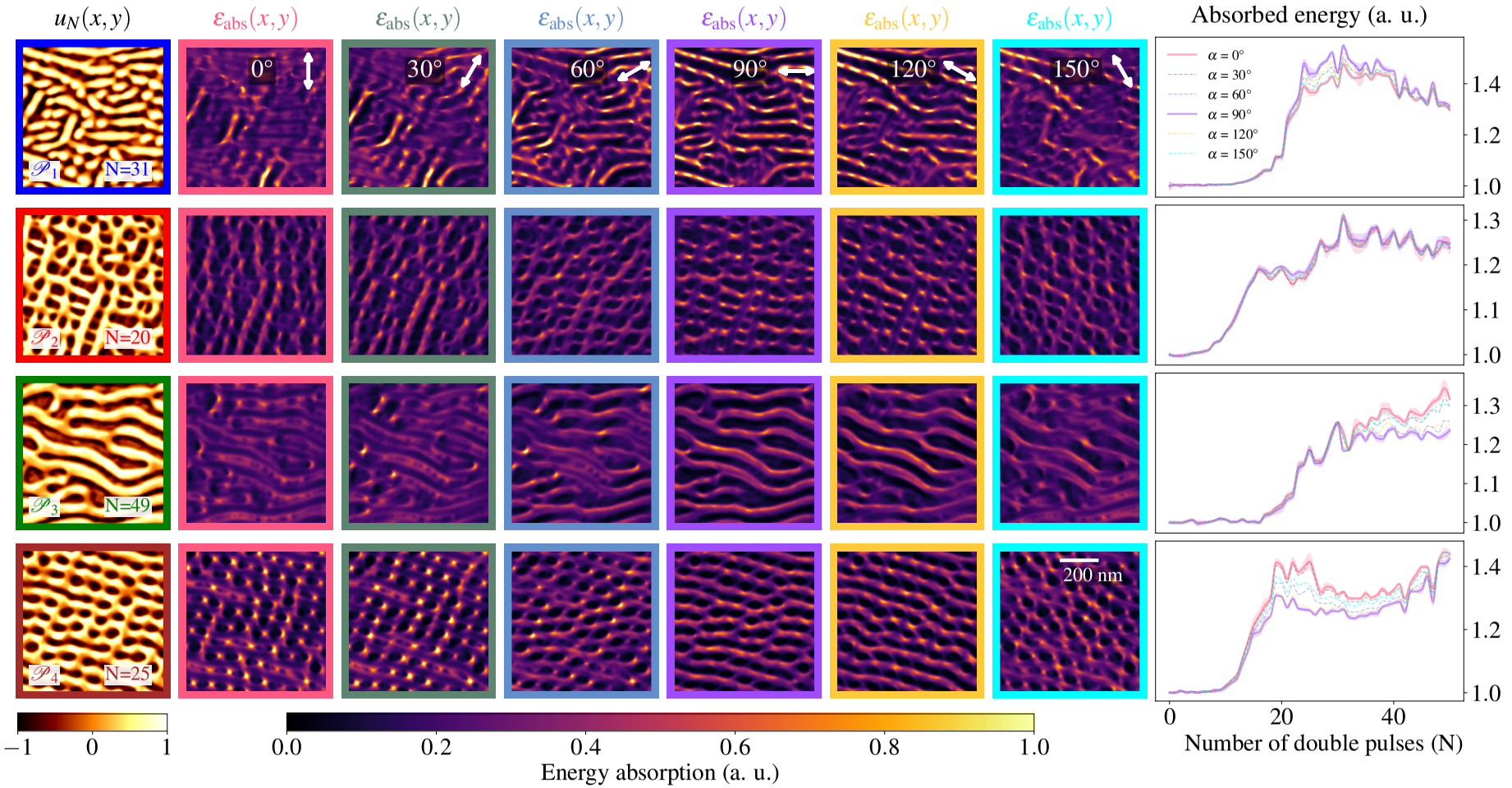


FIGURE 4.9: **Effect of laser polarization angle α on absorbed energy $E_{\mathbf{abs}}$.** Each row corresponds to a specific series i ($i = 1$ to 4). The columns, from left to right represent: the pattern height map $u_N(x,y)$ corresponding to $\mathcal{P}_i$, the pattern maximizing E_{abs}^{N+1} in series i: $\mathcal{P}_1$ (series 1, at $N = 31$), $\mathcal{P}_2$ (series 2, at $N = 20$), $\mathcal{P}_3$ (series 3, at $N = 49$), and $\mathcal{P}_4$ (series 4, at $N = 25$); the energy absorption from the subsequent pulse $\varepsilon_{\text{abs}}^{N+1}(x,y)$ for $\alpha = k\pi/6$ for $k = 0$ to 5 and the last column is the integrated absorbed energy $E_{\text{abs}}^{N+1} = f(N,\alpha)$. The shaded region for $\alpha = 0$ and $\alpha = \pi/2$ represents a confidence band corresponding to the standard deviation of E_{abs}^{N+1} calculated across different locations.

The emergence of such patterns is therefore not arbitrary: it reflects an optimization process driven by the maximization of energy absorption.

To probe this hypothesis, we analyze the sensitivity of absorption to perturbations in $\mathcal{P}$. By artificially modifying surface images, we can evaluate how deviations from the optimal morphology affect the absorbed energy E_{abs}. Our working expectation is that arbitrary changes to $\mathcal{P}$, once it has reached a configuration that maximizes E_{abs}, should impair the efficiency of light capture and thus reduce absorption.

Formally, we describe absorbed energy as a function of structural complexity:

$$E_{abs} = E_{abs}(\mathscr{C}(\mathcal{P})), \tag{4.13}$$

where complexity $\mathscr{C}$ itself depends on the underlying pattern $\mathcal{P}$.

Differentiating with respect to $\mathcal{P}$ yields:

$$\frac{dE_{abs}}{d\mathcal{P}} = \frac{dE_{abs}}{d\mathscr{C}} \cdot \frac{d\mathscr{C}}{d\mathcal{P}}. \tag{4.14}$$

This expression reveals that the influence of $\mathcal{P}$ on absorption is mediated entirely through its effect on complexity. In particular:

- Neutral perturbations. If a modification to $\mathcal{P}$ does not alter complexity, i.e. $\frac{d\mathscr{C}}{d\mathcal{P}} = 0$, then absorption remains unchanged: $\frac{dE_{abs}}{d\mathcal{P}} = 0$. In this case, the pattern is insensitive to that class of perturbations.
- Decreasing complexity. If the pattern is perturbed in a way that reduces complexity ($\frac{d\mathscr{C}}{d\mathcal{P}} \leq 0$), then absorbed energy decreases accordingly. This is consistent with a monotonic relationship $\frac{dE_{abs}}{d\mathscr{C}} > 0$: more complex structures are more efficient at capturing energy.
- Increasing complexity. If a perturbation tends to increase complexity ($\frac{d\mathscr{C}}{d\mathcal{P}} \geq 0$) but absorbed energy remains constant ($\frac{dE_{abs}}{d\mathcal{P}} = 0$), then from Eq. (4.14) it follows that $\frac{d\mathscr{C}}{d\mathcal{P}} = 0$. In other words, even though local modifications are attempted, the system cannot increase complexity further while preserving efficiency. This identifies the observed pattern $\mathcal{P}$ as a local maximum of both complexity and absorption.

Taken together, this framework provides a mechanistic explanation for the emergence of self-organized patterns. They correspond to stable configurations where absorption and complexity are jointly optimized. The system evolves toward these morphologies because they maximize energy capture while maintaining a high degree of structural organization. Once such a configuration is reached, it exhibits robustness against arbitrary perturbations since small changes in the surface morphology do not significantly alter absorption. In this sense, the pattern effectively stabilizes as a natural state of the material under irradiation.

Artificial images generation process

To assess the robustness of experimentally obtained self-organized patterns, we generate artificial perturbations of a reference field $\mathcal{P}$. These perturbations serve as controlled variations that allow us to probe whether absorption efficiency is preserved when the underlying structure is modified. Two complementary strategies are employed: (i) the Swift-Hohenberg method, which introduces physically consistent perturbations that remain compatible with self-organization dynamics, and

(ii) the random patch method, which inserts non-physical modifications to test the system's sensitivity to arbitrary disorder. Together, these approaches provide a systematic framework to disentangle robustness from fragility in the learned structural configurations.

(i) Swift-Hohenberg Method Patterns that arise through self-organization can be effectively modeled by the Swift–Hohenberg (SH) equation [16], originally introduced in the study of Rayleigh–Bénard convection [124]. This partial differential equation provides a minimal yet powerful framework for describing how systems driven far from equilibrium can spontaneously develop spatially ordered structures. Its solutions are known to exhibit stationary, periodic, and even chaotic patterns that closely resemble Turing-like morphologies observed in a variety of physical contexts, including the periodic nanostructures generated on laser-irradiated surfaces [16].

For a scalar field $u(\mathbf{x}, t)$, representing the state of the system at position $\mathbf{x}$ and time t, the equation is written as:

$$\frac{\partial u}{\partial t}(\mathbf{x}, t) = ru(\mathbf{x}, t) - \left(\mathbf{q}_0^2 + \nabla^2\right)^2 u(\mathbf{x}, t) + \gamma u(\mathbf{x}, t)^2 - u(\mathbf{x}, t)^3, \tag{4.15}$$

which we denote compactly as $SH(r, \gamma, \mathbf{q}_0)$. Each term in this equation plays a distinct role in shaping the dynamics. The linear term ru acts as a control parameter: when $r > 0$, it destabilizes the uniform state and drives the system toward pattern formation. The operator $\left(\mathbf{q}_0^2 + \nabla^2\right)^2 u$ serves as a spatial filter, selecting modes with a characteristic wavelength $\lambda = 2\pi/|\mathbf{q}_0|$ and suppressing others, thereby fixing the typical size of emergent structures. The quadratic nonlinearity γu^2 breaks the symmetry of the solutions under sign inversion ($u \to -u$), allowing for asymmetric pattern growth. Finally, the cubic term $-u^3$ provides nonlinear saturation, preventing unbounded growth and stabilizing the amplitude of patterns once they form.

Together, these ingredients make the Swift-Hohenberg equation a canonical model for studying self-organization. Its ability to generate a rich spectrum of patterns, including stripes, hexagons and localized structures, explains its central role in connecting fundamental nonlinear dynamics with experimentally observed morphologies such as those produced by ultrashort laser irradiation of solids.

To test the constrained maximization hypothesis, we perturb the original field $\mathcal{P}$ using physically meaningful perturbations δu_{SH}. These perturbations are generated by applying the Swift-Hohenberg equation, $SH(r, \gamma, \mathbf{q}_0)$, to small patches P of the original field. Here, r is the bifurcation parameter controlling the proximity to the instability threshold, γ introduces asymmetry by breaking the sign-inversion symmetry of the solutions, and $\mathbf{q}_0$ represents the typical spatial wavenumber of the pattern. The initial condition for the patch is set as:

$$u(\mathbf{x}, 0) = \mathcal{P}|_P(\mathbf{x}), \tag{4.16}$$

with Neumann boundary conditions ensuring continuity with the surrounding field:

$$u|_{\partial P}(\mathbf{x}, t) = \mathcal{P}|_{\partial P}(\mathbf{x}). \tag{4.17}$$

We use a pseudospectral solver previously developed [17] and sample the parameters $r \in (0, 1]$ and $\gamma \in [-1, 1]$ to explore a wide variety of physically consistent pattern solutions. This procedure guarantees that the generated perturbations remain compatible with the underlying physics of the original field.

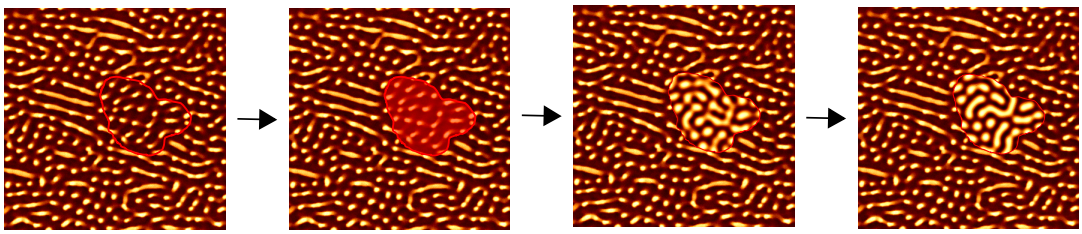


FIGURE 4.10: **Schematic representation of the Swift-Hohenberg method for generating artificial surface patches.** The red contour highlights a selected region that evolves progressively under the Swift-Hohenberg dynamics. For illustration purposes, the shape shown is not convex, as explained in the main text.

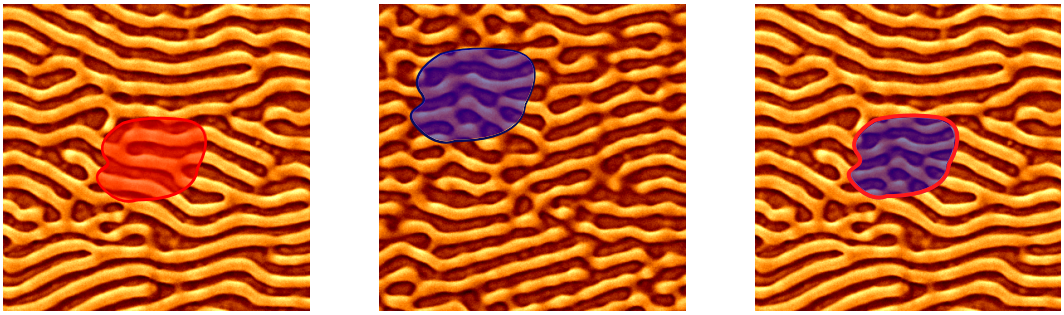
To quantify the magnitude of the perturbation, we define it in terms of the area of the patch. The specific shape of the patch, however, can introduce unphysical artifacts. A patch that minimizes the perimeter for a given area would naturally be circular, but circular domains tend to enforce radial symmetry in the solution, which is undesirable. Instead, we require the patch to locally minimize the perimeter while remaining convex. Using the Valtr algorithm [131], we generate 100-vertex convex polygons uniformly sampled to define these patches.

To prevent artifacts related to the scale of the perturbation, we set $\mathbf{q}_0$ according to the modal spatial frequency derived from the radially averaged power spectrum of the patch. This ensures that the perturbation respects the dominant length scales of the original pattern.

The overall procedure for generating a physically consistent perturbation $\mathcal{P} + \delta\mathcal{P}$ can be summarized as follows:

1. Select the original field $\mathcal{P}$ and choose a center c for the domain of the patch.
2. Generate a convex patch P of $\mathcal{P}$ with 100 vertices using the Valtr algorithm.
3. Determine $\mathbf{q}_0$ from the modal spatial frequency of the radially averaged power spectrum within P.
4. Sample $r \sim U(0, 1]$ and $\gamma \sim U([-1, 1])$ to define the Swift-Hohenberg dynamics.
5. Solve the Swift-Hohenberg equation on P for a short evolution time t using the initial condition $u(\mathbf{x}, 0) = \mathcal{P}|_P(\mathbf{x})$ and Neumann boundary conditions $u|_{\partial P}(\mathbf{x}, t) = \mathcal{P}|_{\partial P}(\mathbf{x})$.
6. Replace the original patch P in $\mathcal{P}$ with the transformed patch from the Swift-Hohenberg evolution.

This systematic approach generates a diverse ensemble of physically admissible perturbations that remain consistent with the underlying material dynamics, allowing us to probe the sensitivity of the pattern $\mathcal{P}$ to variations compatible with the physical process. Fig. 4.10 provides a schematic illustration of the method, while Fig. 4.12 shows examples of patterns obtained using this approach. The resulting patterns after this replacement are denoted by $\mathcal{P}_{SH}$.



(a) (b) (c)

FIGURE 4.11: **Schematic illustration of the random patch method.** a) The target field $\mathcal{P}$, b) the source field $\mathcal{S}$, and c) the resulting artificial surface after replacing a patch. For visualization purposes, the patch shown here is not convex, as explained in the main text.

(ii) Random patch method In this method, we perturb the original field $\mathcal{P}$ using a perturbation that does not arise naturally from the physical self-organization process. The goal is to introduce non-physical variations while maintaining some structural consistency in terms of patch shape. We start by identifying a target pattern, represented by the field $\mathcal{P}$, and a source pattern $\mathcal{S}$ obtained from the learning regime. The perturbation is generated by replacing a small patch P from the target field $\mathcal{P}$ with a corresponding patch S from the source field $\mathcal{S}$.

To ensure consistency in the patch shapes, both P and S are defined on convex domains, generated using the Valtr algorithm, as in the Swift-Hohenberg perturbation method. This algorithm produces convex polygons with 100 vertices, sampled uniformly, guaranteeing that the shapes of the patches are comparable across fields and do not introduce artifacts from irregular boundaries.

The procedure for generating these non-physical perturbations can be summarized as follows:

1. Select the source field $\mathcal{S}$ and the target field $\mathcal{P}$, and sample uniform centers, c_s and c_p, to define the locations of the patches in each field.

2. Generate a convex patch P of the target field $\mathcal{P}$ and a corresponding convex patch S of the source field $\mathcal{S}$. Both patches are 100-vertex convex polygons, sampled uniformly using the Valtr algorithm.

3. Replace the patch P in the target field $\mathcal{P}$ with the patch S from the source field $\mathcal{S}$.

This procedure creates a diverse set of non-physical perturbations of the original field, introducing features that do not result from the inherent dynamics of self-organization. These perturbations allow us to investigate how the system responds to externally imposed variations. Fig. 4.11 provides a schematic illustration of the method, while Fig. 4.12 shows examples of patterns obtained using this approach. The resulting patterns after this replacement are denoted by $\mathcal{P}_{RP}$.

Predicting absorption in modified patterns with a surrogate model

We compare the variation of absorbed energy between the original experimental surfaces and their artificially modified counterparts. For this analysis, we deliberately

focus on four representative series corresponding to surfaces that naturally emerged through self-organization and experimentally achieved the highest absorption. These optimal configurations are: series 1 ($N = 31$, $\alpha = \pi/2$), series 2 ($N = 20$, $\alpha = \pi/2$), series 3 ($N = 49$, $\alpha = 0$), and series 4 ($N = 25$, $\alpha = 0$). In total, 8000 modified surfaces are analyzed, with 4000 generated by the Swift–Hohenberg perturbation method and 4000 by the random patch method (1000 surfaces per method for each series).

Given the size of this dataset, directly evaluating absorption maps $\varepsilon_{\text{abs}}(x, y)$ using a numerical solver would be prohibitively time-consuming. Instead, we employ a surrogate neural network model, which, once trained, can provide rapid predictions across large ensembles. The workflow follows the approach detailed in the previous chapter, though it is restated here for clarity. Three-dimensional SEM images $u_N(x, y, z)$ are first used in conjunction with the numerical solver to compute absorption distributions $\varepsilon_{\text{abs}}(x, y, z)$, where the grayscale pixel contrast encodes the surface height in the z direction. Both inputs and outputs are integrated along z, transforming the problem into an image-to-image regression task. For this task, we adopt the U-Net neural network [104], originally developed for biomedical image segmentation. Its encoder–decoder architecture efficiently captures both global and local features, making it well suited for predicting absorption patterns. Here, the U-Net is adapted for regression, mapping $u_N(x, y)$ and the polarization angle α to the corresponding absorption field $\varepsilon_{\text{abs}}(x, y)$.

Sensitivity of self-organized structures to perturbations

To investigate how sensitive experimentally obtained self-organized patterns are to external modifications, we take a reference pattern, denoted by $\mathcal{P}$, that maximizes the absorbed energy E_{abs} within a given experimental series i. We then systematically perturb this pattern and evaluate how the absorption changes. Fig. 4.12 summarizes the results, where we plot the relative variation in absorbed energy, ΔE_{abs}, as a function of the Frobenius norm that quantifies the distance between the original pattern $\mathcal{P}$ and its modified versions. To facilitate comparison, the Frobenius norm $\|\mathcal{P} - \mathcal{P}'\|_F$ is normalized to the interval $[0, 1]$, so that values close to zero correspond to patterns very similar to $\mathcal{P}$ while values close to one indicate strong deviations. The absorption of the unperturbed experimental pattern, $E_{\text{abs}}(\mathcal{P})$, is shown as a horizontal dashed line, serving as a reference against which all variations can be assessed.

We explore two complementary strategies for perturbation, each designed to test a different aspect of structural stability.

The first method introduces random perturbations. A small patch of the original pattern $\mathcal{P}$ is replaced with another patch randomly extracted from patterns within the same experimental series, but specifically taken from configurations that still belong to the "learning regime" (where structures are not yet fully stabilized). This yields artificially modified surfaces denoted $\mathcal{P}_{RP}$. Such replacements intentionally break the self-consistency of the original self-organized structure, producing local regions that do not follow the same organizing principles. When computing absorption for these randomized surfaces, $E_{\text{abs}}(\mathcal{P}_{RP})$, the scatter points cluster tightly in a small region of the diagram, consistently below the dashed reference line. Importantly, this drop occurs irrespective of how close or far the modified surface is from the original pattern in terms of Frobenius distance. In other words, even small, localized random perturbations disrupt the carefully tuned balance that maximizes absorption, revealing that self-organized structures are fragile to arbitrary disorder.

The second method introduces structured perturbations by replacing a portion of $\mathcal{P}$ with a patch generated from a Swift–Hohenberg (SH) solver. The SH equation is a canonical model for hydrodynamic instabilities and is well known for producing pattern-like morphologies that bear strong resemblance to the structures observed in our experiments. It has been shown that the physics of self-organization in laser–matter interaction can, at least approximately, be captured by such equations [16]. This procedure generates modified patterns, $\mathcal{P}_{SH}$, that do not simply insert arbitrary disorder but instead preserve the essential dynamical rules of pattern formation. When we compute absorption for these surfaces, $E_{\text{abs}}(\mathcal{P}_{SH})$, the scatter points spread more widely across the diagram and are distributed both above and below the reference line. However, the deviations remain relatively small in magnitude, regardless of how different the modified pattern is from the experimental one. This indicates that as long as the modifications themselves respect the physical principles of self-organization, the system retains a degree of robustness and continues to absorb energy at nearly optimal levels.

Taken together, these two perturbation strategies reveal a fundamental asymmetry in how self-organized patterns respond to change. Random substitutions, which introduce disorder without regard for the underlying organizing rules, consistently degrade absorption performance even when the perturbation is small. This demonstrates that the efficiency of a self-organized pattern is not a trivial byproduct of its geometry but instead the result of a highly tuned internal consistency where each local feature is shaped by and contributes to a global organizational logic. In structural learning terms, such random perturbations correspond to "noise" that lies outside the learned structural space. Since the system has never learned how to accommodate such disordered elements, performance collapses immediately upon their introduction.

By contrast, perturbations generated with the Swift–Hohenberg (SH) solver, although they alter the microscopic details of the surface, still preserve the broader dynamical rules of pattern formation. These modifications are therefore structurally consistent with the learning process because they represent alternative solutions within the same learned space of configurations. As a result, the absorption performance of the modified patterns remains close to optimal, with only minor fluctuations around the experimental baseline. From a structural learning perspective, this indicates that self-organization does not simply produce one fixed solution but rather stabilizes a family of structurally related solutions, all of which capture energy efficiently. The resilience observed here reflects a hallmark of learning systems: robustness to perturbations that respect the rules of the learned representation, even as they remain fragile to unstructured noise.

This duality, involving fragility to random disorder and robustness to structurally meaningful variation, captures the essence of self-organization as a form of physical learning. The experimental patterns can be seen as knowledge states encoded by the surface, where every local feature is integrated into a coherent whole. Random alterations disrupt this learned coherence in the same way that introducing meaningless symbols into a sentence destroys its readability. In contrast, SH-generated perturbations act like grammatical variations: the surface sentence changes in appearance, but the underlying structure remains intact and the meaning, here understood as efficient absorption, is preserved.

More broadly, these results suggest that self-organization in laser–matter interaction embodies a structural learning process where the system explores, stabilizes, and eventually locks into a subset of morphologies that optimize function. Once

established, these configurations are robust to structured perturbations that still respect the learned rules, but remain exquisitely vulnerable to unstructured noise. This fine balance explains why experimental patterns appear both universal, since they emerge reliably under a wide range of conditions, and delicate, since they require the interplay of physical principles rather than arbitrary geometries.

4.4 Discussion: Learning to learn in self-organized surface

The experimental observations presented above reveal that surfaces exposed to repeated laser irradiation do not merely undergo passive modification, but instead follow a trajectory that can be meaningfully interpreted in terms of learning. Much like biological systems that adapt their behavior through associative and non-associative mechanisms, irradiated materials exhibit cumulative, history-dependent changes that progressively optimize their interaction with light. Beyond these parallels, the notion of structural learning provides an additional layer of interpretation: the evolving surface morphology itself acts as a memory substrate, encoding past irradiation events into persistent spatial features that guide subsequent responses. In this discussion, we explore how concepts of associative, non-associative, and structural learning together provide a powerful framework to describe self-organization in laser–matter interactions. By drawing parallels to adaptation, sensitization, memory formation, and even over-optimization, we show that the evolution of surface structures can be understood as a learning-like process unfolding across successive stages of irradiation.

(i) From adaptation to over-optimization How does matter learn to adapt when it is repeatedly exposed to light? The key lies in the emergence of complexity that arises from hydrodynamically driven transformations [12, 105]. Each interaction between the material and a sequence of light pulses modifies the surface ever so slightly, but the cumulative effect is profound. With every new exposure, the material not only absorbs more energy but also develops increasingly intricate surface features. This progressive increase in both absorption and morphological complexity signals the onset of a learning-like process.

At the beginning, repeated irradiation enhances the material's sensitivity, gradually making it more responsive to the incoming light. As this process continues, a self-organized response emerges: the surface structures reconfigure themselves in a way that optimizes the exchange between the incident light and the absorbing medium. In this phase, the material acts much like a system that has learned to adapt its interface, improving the efficiency of energy transfer through carefully coordinated surface features.

Yet this process is not indefinitely beneficial. As the topography becomes more refined, even small structural deviations start to reduce the efficiency of energy capture, since the surface is tuned toward very specific configurations. Over time, the same mechanisms that once promoted adaptation can turn into sources of fragility. Under prolonged irradiation, the system may reach a state of over-optimization, where the structures are so specialized that they amplify local effects such as hotspots or excessive curvature. These local features disrupt the balance of the global structure, ultimately degrading the overall ability of the surface to harness light efficiently.

(ii) Learning and chaotic regimes of self-organization As surface self-organization advances and the patterns evolve, as illustrated in Fig. 4.4, there is a continuous

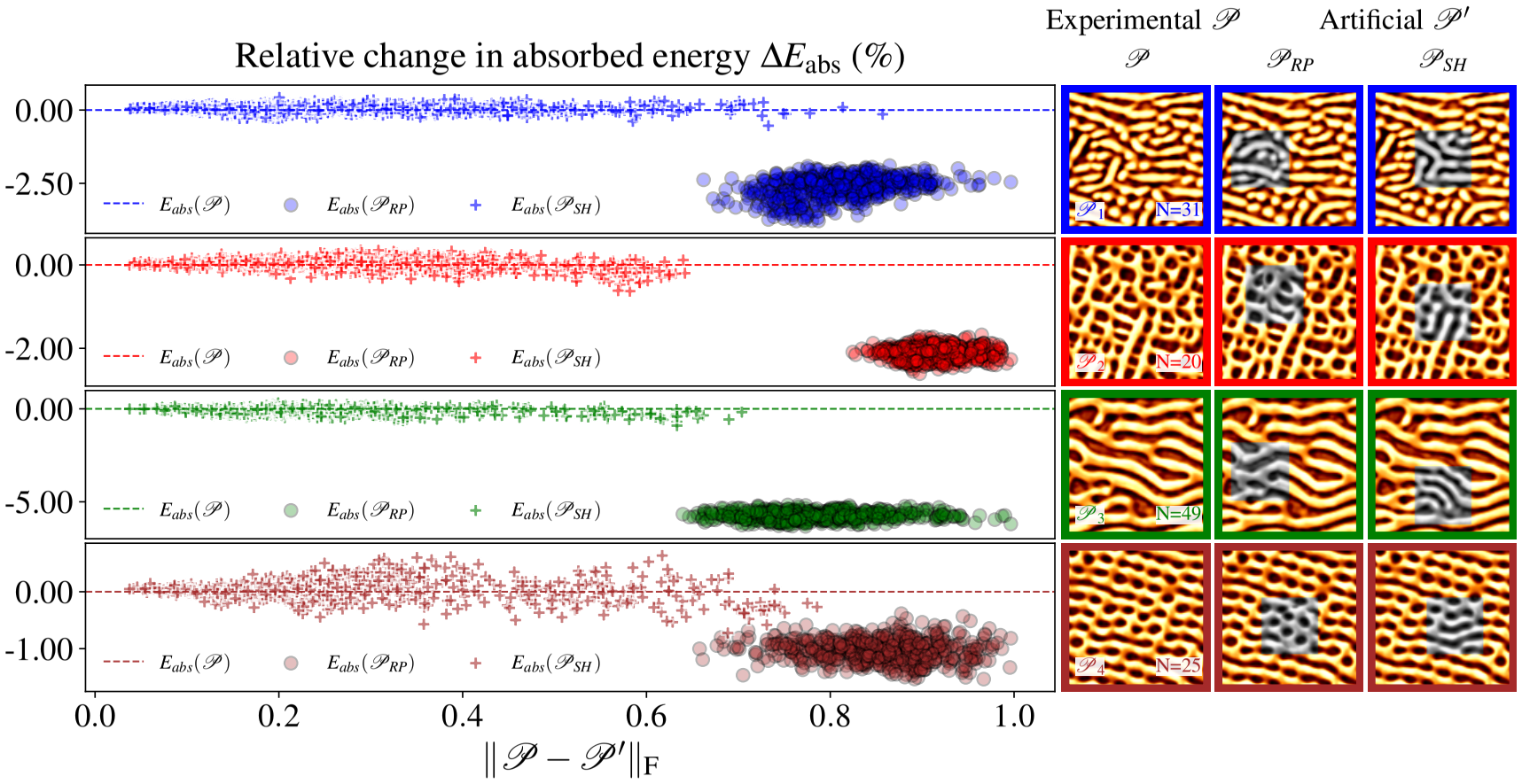


FIGURE 4.12: **Effect of modifying the structure of a pattern $\mathcal{P}$ that maximizes energy absorption.** Each row correspond to a distinct series i ($i = 1$ to 4). On the right, the columns represent, from left to right : the experimental pattern $\mathcal{P}$ that maximizes E_{abs}^{N+1}, the artificial pattern $\mathcal{P}_{RP}$ created by randomly replacing a part of $\mathcal{P}$ by a patch from the same series and the artificial pattern $\mathcal{P}_{SH}$ obtained by replacing a part of $\mathcal{P}$ by a patch produced by Swift-Hohenberg numerical solver. The highlighted gray areas indicate the modified areas relative to $\mathcal{P}$. On the left, the scatter plot shows, the relative change in absorbed energy $\Delta E_{\text{abs}} = 100 \cdot \frac{E_{abs}(\mathcal{P}') - E_{abs}(\mathcal{P})}{E_{abs}(\mathcal{P})}$ as a function of the Frobenius norm $\|.\|_F$ between the experimental pattern $\mathcal{P}$ and its corresponding artificial version $\mathcal{P}'$ ($\mathcal{P}_{RP}$ or $\mathcal{P}_{SH}$)

reciprocal influence between structural complexity and the optical response. This mutual interaction becomes clear when examining the joint evolution of absorbed energy E_{abs} and complexity $\mathscr{C}$ in Fig. 4.8, which reveals two distinct behavioral regimes.

At the beginning of the *learning regime*, the system displays a strong correlation between absorbed energy and structural complexity under fixed laser conditions. As patterns become more visible, they enhance the coupling between light and matter, allowing the surface to progressively "learn" how to maximize energy absorption. However, the dynamics change once this process reaches a certain threshold. The system then enters a *chaotic regime*, where the clear correlation between complexity and absorption breaks down. Now, small surface structures at the nanometer scale start to control how light interacts with the material. Instead of globally enhancing absorption, these extreme local features scatter and redistribute energy irregularly, reducing the efficiency of the overall process.

Thus, the trajectory of self-organization can be understood as a progression from a regime of adaptive learning, where structural growth consistently improves optical performance, to a chaotic regime, where over-complexity undermines the efficiency it previously sustained.

(iii) Structural sensitization: a parallel with biology The analysis of Fig. 4.8 shows that during the process of self-organization, the surface does not simply undergo structural rearrangement. Instead, it progressively acquires new functional properties that enhance its ability to interact with light. This suggests that structural rearrangement under repeated irradiation follows a selective trajectory that gradually tunes the material toward configurations that are increasingly effective at absorbing and redistributing laser energy.

This trend is more explicit in Fig. 4.9, where absorption is observed to increase across all polarization angles but the response is not uniform. The material demonstrates its highest and lowest sensitivities at $\alpha = 0$ and $\alpha = \pi/2$, which correspond to polarization orientations used to generate the structures experimentally. This observation indicates that the surface becomes particularly responsive to the specific conditions under which it was irradiated, almost as if it remembers and adapts to those conditions.

This behavior closely parallels the biological phenomenon of sensitization, a well-characterized form of non-associative learning. In neuroscience, sensitization refers to the process whereby repeated exposure to a stimulus results in a progressively stronger response to that stimulus. Unlike habituation, where responsiveness decreases after repeated exposure, sensitization amplifies the reaction, making the organism more alert or more reactive to the same input. A well-known example concerns the intensified gill-withdrawal reflex in the sea slug Aplysia after repeated tactile stimulation which demonstrates increased sensitivity to repeated harmful stimuli. Sensitization arises through mechanisms of neuroplasticity, where neural circuits alter their strength and connectivity. At the cellular level, it often involves enhanced synaptic transmission, increased neurotransmitter release, or changes in receptor density that together make the nervous system more responsive to repeated inputs.

By analogy, irradiated materials exhibit a comparable form of sensitization. Instead of neurons and synapses, the learning substrate is the surface morphology. Repeated light exposure triggers an active reorganization of the surface, where each successive irradiation enhances sensitivity to the same optical conditions. The patterns that emerge on the surface can thus be viewed as a form of structural memory:

they encode information about the polarization states of the incoming light and reshape the surface's response accordingly. Each newly formed feature is influenced by the preexisting structures, creating a feedback loop where the surface increasingly reinforces configurations that maximize absorption for the specific stimulus it has experienced. Over successive pulses, the material effectively "learns" which structural arrangements are most efficient at capturing and redistributing light energy. This learning process is gradual and cumulative with each laser pulse acting as a training step that fine-tunes the system in order to reinforce constructive structural correlations and suppress inefficient configurations. In this way, the material exhibits a form of memory and adaptive behavior akin to that observed in biological systems where past exposure guides future responses, leading to increasingly optimized energy absorption.

This analogy highlights an essential point: sensitization is not simply about becoming more responsive but about developing condition-specific sensitivity. In organisms, this ensures faster or stronger responses to biologically relevant stimuli. In materials, it ensures that the surface becomes finely tuned to the polarization states it has repeatedly experienced. As a result, self-organization under laser irradiation does not just optimize absorption in general but selectively enhances absorption under specific and experimentally relevant conditions.

(iv) Phases of learning in materials and biology At the microscopic scale, the self-organization of laser-irradiated surfaces begins with localized energy absorption, which triggers small, initial modifications that act as seeds for pattern development. This initial stage corresponds to the *response* phase (I), where the surface effectively "senses" the incoming laser stimulus and reacts by developing surface roughness. The analogy with biological systems is striking: for instance, plants exposed to low-light environments activate survival mechanisms that modify their growth in response to the stimulus. Both systems display an early, exploratory adaptation to external input.

As irradiation continues, the system transitions into the *iterative learning* phase (II). In laser-structured materials, the surface gradually self-organizes into spatially resonant modes that enhance light absorption. This process mirrors the way plants progressively adjust their structure through photomorphogenesis, which is the light-dependent regulation of plant growth and development. Photomorphogenesis encompasses a variety of adaptive responses, including stem elongation, leaf expansion and orientation, chloroplast movement, and modulation of growth rates, all orchestrated to optimize light capture for photosynthesis and survival. In both cases, adaptation is cumulative: each successive interaction builds upon prior modifications, guiding the system toward configurations that improve its efficiency or performance under the given stimulus.

Eventually, both materials and plants reach the *memory stabilization* phase (III). In materials, the patterns established during prior pulses become robust and stable, effectively "recording" the history of exposure. In plants, prior photomorphogenic adaptations are consolidated into irreversible structural changes. In both cases, these modifications persist due to hysteresis effects, reflecting a form of structural memory that ensures sustained functional performance.

Finally, prolonged or excessive exposure can push the system into the *destruction* phase (IV). In laser-irradiated surfaces, accumulated thermal and mechanical stresses can erode the previously established structures, reducing absorption efficiency and destabilizing the surface morphology. In plants, extreme light deprivation or overstimulation can hinder growth or even be fatal, as light acts simultaneously as an energy source and a directional guide.

These four phases illustrate a learning-like process in laser-matter interactions. While the surface does not possess consciousness, its continuous exposure to light drives adaptive changes in structure that resemble biological sensitization. The term "learning" is thus used in a broader sense to describe adaptation based on cumulative experience. Here, the system integrates feedback from previous interactions, adjusting its morphology to improve energy absorption and performance in subsequent exposures. This perspective allows us to interpret pattern formation not as a random occurrence or a fixed, static response but as a dynamic, history-dependent process where each modification shapes the future behavior of the surface. By continuously adapting and consolidating structural features, the surface effectively encodes the outcome of its interactions, similar to how a biological system accumulates experience over time, making the material's response increasingly efficient and robust until external conditions or overstimulation disrupt the process.

4.5 Conclusion

To conclude, our results demonstrate that the emergence of self-organized structures under repeated laser irradiation can be interpreted as the outcome of a learning-like process. Each sequence of laser pulses triggers subtle modifications in the surface, which accumulate over time into increasingly complex and finely tuned morphologies. Through this progressive rearrangement, the material adapts its interface to the optical stimulus, reinforcing structural configurations that locally maximize energy absorption. In this sense, the surface does not passively deform but actively "learns" to enhance its interaction with light, encoding in its morphology the history of its exposure.

The resulting patterns represent optimal configurations for light–matter coupling, where the balance between morphological complexity and absorption efficiency is most favorable. This trajectory closely parallels adaptive processes in biology, such as reinforcement learning in plants, where structures evolve under environmental constraints to optimize light capture. Just as plants gradually reorient and reshape themselves through photomorphogenesis, laser-irradiated surfaces follow an iterative pathway of response, adaptation, and memory stabilization.

Importantly, this analogy also highlights the limits of adaptation. While the learning regime fosters a strong correlation between complexity and absorption, prolonged irradiation can push the system beyond optimality into a chaotic regime, where over-specialized features such as hotspots or excessive curvature degrade performance. Thus, the same mechanisms that promote adaptation and sensitization can, under sustained exposure, lead to fragility and eventual decline.

Taken together, these findings suggest that matter under repeated irradiation exhibits a history-dependent, selective adaptation process, akin to biological forms of learning. The surface progressively encodes the conditions of its interaction with light, develops condition-specific sensitivity, and consolidates its response into stable structural patterns. By framing self-organization as a learning process, we not only gain a unifying perspective on laser–matter interactions but also establish a conceptual bridge between physical and biological systems, where both adapt through cumulative feedback to optimize their performance under repeated stimuli.

Chapter 5

Unrolled-SINDy: Differential Equation Discovery under Sparse Data and Large Time-step Sampling

Discovering governing partial differential equations (PDEs) from experimental or simulated data is a central challenge in physics-based modeling, particularly when observations are sparse, noisy, or separated by large temporal gaps. Traditional approaches often struggle under these conditions, limiting their applicability to real-world systems. As the second main contribution, this chapter introduces Unrolled-SINDy, a novel extension of the Sparse Identification of Nonlinear Dynamics (SINDy) framework designed to reconstruct system dynamics from widely spaced observations in time, enabling the discovery of post-irradiation processes that drive the self-organization of matter without the need for additional measurements.

5.1 Introduction

Embedding physical knowledge into learning algorithms is at the core of Physics-informed Machine Learning (PIML) [60], a new line of research that has recently attracted much attention from both physics and ML communities. Beyond addressing issues related to ill-posed problems, data scarcity and solution consistency, PIML can be applied for (i) solving Differential Equations (ODEs or PDEs) (*e.g.* PINNs [102], FNOs [69]), (ii) leveraging physical priors to accelerate the learning process in hybrid (knowledge+data) modeling (*e.g.* PINO [70]) or augment incomplete physical knowledge (*e.g.* APHYNITY [141] or hybrid PINNs [35]) when physics is only partially understood or where it is derived under ideal conditions that do not hold exactly in real applications, and (iii) learning governing differential equations directly from data measurements (thus solving an inverse problem) in domains where the theory remains elusive. In this chapter, we focus on this latter scenario which has been shown to be of great help for discovering knowledge in various domains, including engineering, climate science, finance, medicine, biology and chemistry.

One of the most important breakthroughs in discovering equations from data came with the SINDy algorithm (see [20] for ODEs and [109] for its extension to PDEs with PDE-FIND) which envisions the problem from the perspective of sparse regression performed from a library of functions (typically including partial derivatives, trigonometric or polynomial terms). SINDy leverages the realistic assumption that in most equations, only a few important terms govern the underlying dynamics, prompting us to promote sparsity. The pioneered version of SINDy relies on an *explicit numerical method*, meaning that it calculates the state of the system at $t + h_t$

using known values from the current time step t. Several extensions have been introduced since then, including SINDYc [21] to leverage external inputs and feedback control, Reactive SINDy [55] to deal with vector-valued ansatz functions, Ensemble-SINDy [41] enabling uncertainty quantification, SINDy-PI [58] which uses rational functions to discover equations, an extension of SINDy to stochastic dynamical systems [11] or WSINDy [87] which eliminates pointwise derivative approximations with a weak formulation providing better robustness to noise.

Despite remarkable performances, SINDy-like explicit methods face a major limitation when deployed on real applications: they rely on accurate time derivative approximations along the identification process, typically using numerical methods like Euler, imposing constraints on the sampling time step sizes h_t of the data. As illustrated in Fig. 5.3.b, this can constitute a serious obstacle in scenarios where data is scarce, leading to large local truncation errors. To overcome this limitation, a significant improvement has been proposed with RK4-SINDy [51], which leverages the *explicit* fourth-order Runge-Kutta method and its advantageous convergence properties to better recover the underlying equations. However, even though RK4 pushes the limits of Euler to some extent, its small *absolute stability region* still limits RK4-SINDy, particularly for identifying stiff equations, unless very small step sizes are used. From an optimization perspective, as for both numerical methods, the local truncation error worsens with the growth of h_t, **the parameters of the equation underlying the sparse data no longer represent a minimizer** of the considered optimization problem.

To address this major issue, we introduce a new PDE discovery method, namely Unrolled-SINDy, which **decorrelates the numerical time step size from the available data sampling rate**. It consists in unrolling K-times each numerical step, as illustrated in Fig. 5.4.b, **without needing additional data**. To do so, Unrolled-SINDy evaluates the library terms at K iterative estimates of the solution between two successive observations $u(t)$ and $u(t + h_t)$. We show that this way, Unrolled-SINDy leverages an intrinsic smaller time step and thus benefits of a lower local truncation error. If at first sight, our unrolled scheme may seem close to the principle of Runge–Kutta methods, it is important to note that the latter are difficult to generalize to many stages associated with a large number of Butcher order conditions, preventing non-unique and closed-form solutions for the coefficients, and leading sometimes to numerical instability.

We position this original contribution within a framework where parsimonious models are promoted, and the identification of the governing equations can be solved in closed-form (*i.e.*, no parameter tuning aside a sparsity threshold) and relies on an explicit numerical technique to approximate the time derivatives. This concerns the wide range of SINDy-like methods that have flourished in the literature during the past few years and that belong to the state of the art in ODE/PDE discovery. However, our proposed unrolling strategy can also be applied to neural network-based PDE discovery methods, including PDE-Net [76], DeepMoD [14], PDE-LEARN [119], PDE-READ [120], PINN [102], ICNET [26] or iNeural-SINDy [48]. Broadly speaking, any neural method that implements a discretization scheme can directly benefit from our unrolling approach. We illustrate this nice feature by applying the unrolling inside the state-of-the-art noise-robust iNeural-SINDy.

To recap, the contribution of the current chapter is four-fold:

- We propose an unrolling scheme for SINDy-like methods, which decorrelates the integration time step from the data inter-observation time step h.

- Based on this methodology, we propose Unrolled-SINDy, a new explicit ODE/PDE discovery method. Unrolled-SINDy is simple to implement, relatively fast and comes with a closed-form. It is versatile as it can be adapted to any explicit method based on a finite-difference technique, such as forward Euler, central difference or Runge-Kutta. Importantly, the main contribution is "unrolling" approach itself not the unrolling a specific integration scheme. The approach is applicable to any numerical scheme in order to identify a governing equation in situation where the observations are largely spaced in time.
- Given a time step size h, we show that K Unrolled-SINDy benefits of a local truncation error in the order of $\mathcal{O}\left((\frac{h}{K})^p\right)$ that holds for any Runge-Kutta method of order p.
- We carry out a comprehensive experimental study on different equations and show that, with both Euler and RK4, our Unrolled-SINDy is more robust at recovering the underlying physics than SINDy and is able to solve problems with large time steps that are inaccessible to current methods. We also demonstrate that unrolling benefits neural PDE discovery methods: on state-of-the-art noise-robust iNeural-SINDy, unrolling allows to tackle problems with scarcer observations.

5.2 Background and preliminaries

5.2.1 Differential Equations in data-driven modeling

Differential equations lie at the heart of mathematical modeling because they formalize how systems evolve when influenced by intrinsic dynamics and external forces. They provide a bridge between observed physical phenomena and their underlying governing laws. To motivate the Sparse Identification of Nonlinear Dynamics (SINDy) framework and its extensions to partial differential equations, it is essential to distinguish between ordinary differential equations (ODEs) and partial differential equations (PDEs).

Ordinary Differential Equations (ODEs)

An ordinary differential equation (ODE) describes the rate of change of a system whose state depends only on a single independent variable, typically time. If the system state is represented by a vector $u(t) \in \mathbb{R}^n$, where n is the number of state variables, the ODE can be written as

$$\frac{du}{dt}(t) = f(u(t), t)$$

where f specifies the underlying law of motion. For instance:

- **Newton's law for a falling object with linear fluid resistance:**

  $$\frac{du}{dt}(t) = -g - \gamma u(t)$$

  where $u(t)$ is the velocity of the object, g is the acceleration due to gravity, and γ is the linear drag coefficient that models resistance from air or fluid.

- **Lotka–Volterra model for predator–prey interactions:** The state vector can be written as

$$u(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix}$$

where $x(t)$ represents the prey population and $y(t)$ represents the predator population. The dynamics are then compactly expressed as

$$\frac{du}{dt}(t) = \begin{bmatrix} \alpha x(t) - \beta x(t)y(t) \\ \delta x(t)y(t) - \gamma y(t) \end{bmatrix}$$

$$\frac{dx}{dt}(t) = \alpha x(t) - \beta x(t)y(t), \quad \frac{dy}{dt}(t) = \delta x(t)y(t) - \gamma y(t).$$

where α is the prey reproduction rate, β is the predation rate coefficient, δ is the rate at which consumed prey is converted into predator growth, and γ is the predator mortality rate.

The defining feature of ODEs is that they describe systems evolving only in time, with no explicit dependence on space. ODEs are widely used in physics, biology, and engineering to model population dynamics, mechanical systems, electrical circuits, and chemical kinetics.

Partial Differential Equations

Many natural phenomena are not purely temporal but evolve across both time and space, requiring partial differential equations (PDEs) for their description. While ODEs capture dynamics that depend only on time, PDEs extend this idea by incorporating spatial variation. In this sense, PDEs can be seen as a generalization of ODEs. ODEs describe the temporal evolution at individual points in space, while PDEs describe the collective dynamics across space and time. Conversely, a PDE describes how the state of a system evolves simultaneously in time and across space, capturing interactions and couplings that ODEs cannot represent.

Formally, if the system state is denoted by $u(t, \mathbf{x}) \in \mathbb{R}^n$, depending on time $t \in [0, T]$ and spatial coordinates $\mathbf{x} = (x_1, x_2, \ldots, x_d) \in \Omega \subset \mathbb{R}^d$, a PDE can generally be written as

$$\frac{\partial u}{\partial t}(t, \mathbf{x}) = \mathcal{N}[u(t, \mathbf{x})]$$

where $\mathcal{N}$ is a differential operator involving spatial derivatives of u, which encodes the governing physical laws. In the absence of spatial dependence, $\mathcal{N}$ reduces to a function of u alone, recovering the standard ODE form. Examples include:

- **Heat equation (diffusion of temperature):**

$$\frac{\partial u}{\partial t}(t, \mathbf{x}) = \kappa \Delta u(t, \mathbf{x})$$

where $u(t, \mathbf{x})$ is the temperature field, κ is the thermal diffusivity, and Δ is the Laplace operator representing spatial diffusion. In d dimensions, the Laplacian is defined as

$$\Delta u(t, \mathbf{x}) = \frac{\partial^2 u}{\partial x_1^2}(t, \mathbf{x}) + \frac{\partial^2 u}{\partial x_2^2}(t, \mathbf{x}) + \cdots + \frac{\partial^2 u}{\partial x_d^2}(t, \mathbf{x})$$

- **Wave equation (propagation of vibrations):**

$$\frac{\partial^2 u}{\partial t^2}(t, \mathbf{x}) = c^2 \Delta u(t, \mathbf{x})$$

where $u(t, \mathbf{x})$ is the displacement field, c is the wave speed, and Δ is the Laplace operator.

When higher-order derivatives appear in time, as in the wave equation, one can systematically reduce the PDE into a first-order system by introducing new variables corresponding to the time derivatives of $u(t, \mathbf{x})$. For example, defining

$$v(t, \mathbf{x}) = \frac{\partial u}{\partial t}(t, \mathbf{x})$$

the wave equation can be rewritten as a first-order system:

$$\frac{\partial u}{\partial t}(t, \mathbf{x}) = v(t, \mathbf{x}), \quad \frac{\partial v}{\partial t}(t, \mathbf{x}) = c^2 \Delta u(t, \mathbf{x}).$$

The velocity field $v(t, \mathbf{x})$ can be obtained either directly from measurements or numerically by differentiating the observed displacement field $u(t, \mathbf{x})$ with respect to time. This reformulation is particularly useful for numerical integration, stability analysis, and for applying data-driven discovery algorithms such as SINDy, as it allows the system to be expressed entirely in terms of first-order time derivatives.

PDEs are ubiquitous in real-world applications: heat conduction in solids, fluid dynamics described by the Navier–Stokes equations, chemical reaction-diffusion systems, electromagnetic wave propagation, and even financial mathematics. These applications highlight that PDEs not only model continuous processes over space and time but also encode the essential physics needed to predict system behavior.

5.2.2 The Sparse Identification of Nonlinear Dynamics (SINDy)

Main ideas of SINDy

In traditional science and engineering, the dynamics of physical systems are described by differential equations derived from first principles, such as Newton's laws, conservation of mass and energy. These approaches are highly effective when the underlying physics is fully understood, but modern systems are often far more complex. They may involve multiple interacting processes, partially observed variables or unknown influences, making it difficult or even impossible to write down an accurate governing equation from first principles alone. At the same time, advances in experimental techniques, sensors and high-fidelity simulations provide access to large amounts of high-resolution data, capturing the system's behavior in time and space. This combination of incomplete theoretical knowledge and abundant data raises a compelling question: instead of assuming a governing equation and solving it to predict behavior, can we reverse the process and infer the underlying laws directly from observations? In other words, can we "discover" the differential equations that govern the system by analyzing how it evolves?

Discovering governing equations from data involves two complementary tasks. First, it requires identifying the functional structure of the equation, which means determining which mathematical terms and interactions are present in the dynamics, for instance, whether the system exhibits linear or nonlinear behavior, includes diffusion, advection, or reaction terms, or involves products, powers, or derivatives

of the state variables. Second, it requires estimating the numerical values of the coefficients associated with each term, which quantify the strength or rate of each process. Together, these steps define the governing equation that best describes the observed behavior.

It is important to distinguish this task from solving a PDE. When solving a PDE, the form of the equation and its coefficients are assumed known, and the goal is to compute the system state $u(t, \mathbf{x})$ that satisfies the equation for given initial and boundary conditions. In contrast, identifying a PDE means discovering the mathematical form of the equation and the coefficients associated with each term.

For example, consider a classical ODE for a freely falling object with linear air resistance:

$$\frac{du}{dt}(t) = -g - \gamma u(t)$$

where $u(t)$ is the velocity, g is gravitational acceleration, and γ is a drag coefficient modeling resistance from air or fluid. Solving this ODE means computing $u(t)$ for given g, γ, and initial conditions. Identifying the ODE from measured velocity data, on the other hand, involves inferring both the functional form and the numerical values of g and γ that best reproduce the observed motion.

In more complex systems, such as fluid flows or reaction-diffusion processes, identifying a PDE similarly involves selecting the relevant terms in the operator, estimating the coefficients for each term, and uncovering the governing rules. Once identified, the PDE can then be solved to predict the evolution of the system state $u(t, \mathbf{x})$. This distinction makes clear that PDE identification is about learning the governing laws, while solving a PDE is about computing their consequences.

In this work, we focus on the PDE identification task and leverage the Sparse Identification of Nonlinear Dynamics (SINDy) framework to systematically discover governing equations directly from data.

ODE formulation of SINDy

Building on this foundation, the Sparse Identification of Nonlinear Dynamics (SINDy) framework provides a systematic methodology for discovering governing equations directly from data. SINDy leverages the insight that, although the system's behavior may appear complex, it is often driven by a small number of dominant interactions among a large pool of possible nonlinear terms. By enforcing sparsity in the selection of these terms, SINDy identifies the essential components of the governing equations, allowing us to reconstruct the underlying dynamics directly from observations without requiring complete prior knowledge of the physics.

To identify the function f in an ordinary differential equation

$$\frac{du}{dt}(t) = f(u(t), t)$$

we first collect a time history of the system state $u(t) \in \mathbb{R}^n$ at m discrete time points $t_1, t_2, \ldots, t_m$. Each vector $u(t_i) = [u_1(t_i), u_2(t_i), \ldots, u_n(t_i)]$ represents the state of all n variables at time t_i. These measurements are arranged into the state matrix

$$\mathbf{U} = \begin{bmatrix} u(t_1) \\ u(t_2) \\ \vdots \\ u(t_m) \end{bmatrix} = \begin{bmatrix} u_1(t_1) & u_2(t_1) & \cdots & u_n(t_1) \\ u_1(t_2) & u_2(t_2) & \cdots & u_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ u_1(t_m) & u_2(t_m) & \cdots & u_n(t_m) \end{bmatrix} \in \mathbb{R}^{m \times n}$$

The corresponding derivatives $\dot{u}(t_j)$ are either measured directly or approximated numerically and arranged into

$$\dot{\mathbf{U}} = \begin{bmatrix} \dot{u}(t_1) \\ \dot{u}(t_2) \\ \vdots \\ \dot{u}(t_m) \end{bmatrix} \in \mathbb{R}^{m \times n}$$

Before constructing the library matrix, we define a set of candidate functions to include in the dictionary. These functions are assumed to potentially appear in the right-hand side of the ODE. Common choices include constants, linear terms (u_i), higher-order polynomial terms ($u_i^2, u_i u_j$), and trigonometric functions ($\sin(u_i), \cos(u_i)$). This dictionary encodes our prior belief about the types of nonlinear interactions that may govern the system.

Once the dictionary is defined, we construct the library matrix $\Theta(\mathbf{U}) \in \mathbb{R}^{m \times p}$ in a compact form as

$$\Theta(\mathbf{U}) = \left[1, \mathbf{U}, \mathbf{U}^2, \sin(\mathbf{U}), \cos(\mathbf{U}), \dots\right]$$

where for example $(sin(U))_{i,j} = sin(u_i(t_j))$

The matrix $\Theta(\mathbf{U})$ encodes the functional structure of the ODE, with candidate terms including constants, linear terms $\mathbf{U}$, quadratic terms $\mathbf{U}^2$, and trigonometric terms $\sin(\mathbf{U})$ and $\cos(\mathbf{U})$.

The identification problem is then formulated as a regression problem where we seek a coefficient matrix $\alpha \in \mathbb{R}^{p \times n}$ such that

$$\dot{\mathbf{U}} \approx \Theta(\mathbf{U})\, \alpha$$

Sparse regression methods such as LASSO or sequential thresholding are used to identify the nonzero entries of each column α_i, revealing both the functional structure of the ODE (through $\Theta(\mathbf{U})$) and the numerical values of the coefficients (through α) that best reproduce the observed dynamics. In particular, the Sequential Thresholding Ridge (STR) method is often preferred over LASSO in the context of SINDy. STR combines ridge regression with iterative thresholding: at each iteration, a ridge regression is performed to compute the coefficients α, which are then thresholded by magnitude to remove small, likely insignificant terms. The procedure is repeated until convergence.

The thresholding parameter, typically denoted by α_{th}, controls the magnitude below which coefficients are set to zero at each iteration. A higher threshold enforces more sparsity by eliminating weaker contributions, while a lower threshold preserves more terms, potentially capturing subtle interactions. Choosing an appropriate threshold is crucial: if it is too high, important dynamics may be lost but if too low, the model may retain false terms and overfit noise.

The advantage of STR over LASSO lies in its ability to handle highly correlated features in the candidate library $\Theta(\mathbf{U})$, which frequently occurs when using polynomial and trigonometric terms. While LASSO tends to arbitrarily select one among correlated terms and can produce biased estimates of the coefficients due to its ℓ_1 penalty, STR mitigates this issue by using ridge regression for coefficient estimation, providing more stable and accurate identification of the active terms. Additionally, STR allows for fine-grained control over sparsity through the threshold parameter α_{th}, making it particularly effective in recovering the true sparse dynamics of the system from noisy or partially observed data.

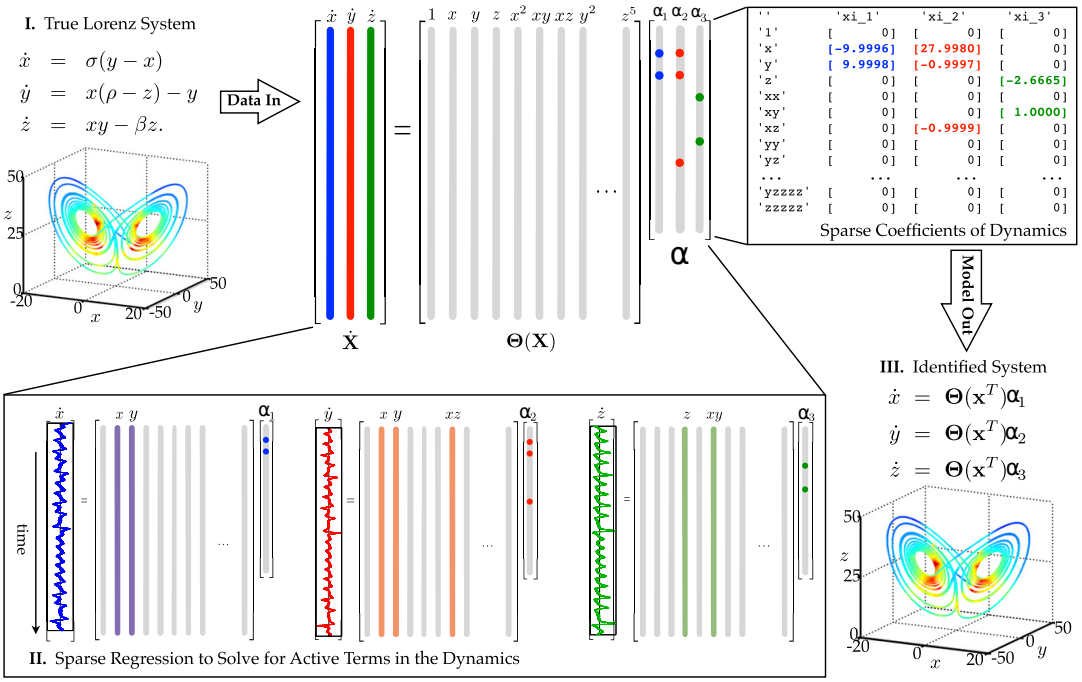


FIGURE 5.1: Schematic illustration of the SINDy algorithm applied to the Lorenz system. Adapted from [20].

Figure 5.1 provides a visual summary of the SINDy workflow. Starting from time-series data, a library of candidate nonlinear functions is constructed and evaluated on the observations. Sparse regression is then applied to select only the most relevant terms from this library, yielding a parsimonious set of governing equations. Although the algorithm has already been described in detail, this schematic highlights the essential steps and illustrates how SINDy recovers the Lorenz system from data.

PDE formulation of SINDy

Having first introduced the SINDy framework in the simpler ODE setting for clarity, we now extend the formulation to PDEs and introduce the notation that will be used throughout the rest of this work.

We consider equations of the general form

$$\frac{\partial u}{\partial t} = \mathcal{N}[u] \tag{5.1}$$

where $\mathcal{N}[\cdot]$ denotes a (possibly nonlinear) differential operator involving spatial derivatives, and $u(t, \mathbf{x}) \in \mathbb{R}^{d_2}$ is the latent, supposedly unique hidden solution, with $t \in [0, T]$ the temporal variable and $\mathbf{x} \in \Omega \subset \mathbb{R}^d$ the spatial coordinates. When higher-order time derivatives appear on the left-hand side, the system can be equivalently rewritten in first-order form by introducing additional variables corresponding to u and its time derivatives.

To identify such equations, we define a library Θ of size $|\Theta|$, typically consisting of constants, exponentials, trigonometric terms, as well as spatial derivatives and polynomial terms up to order r. We denote by

$$\Theta_u : [0, T] \times \mathbb{R}^d \to \mathbb{R}^{|\Theta|} \tag{5.2}$$

the function that maps
$(t, \mathbf{x}) \mapsto \left(\frac{\partial u_1}{\partial \mathbf{x}_1}(t,\mathbf{x}), \frac{\partial u_1}{\partial \mathbf{x}_2}(t,\mathbf{x}), \frac{\partial^2 u_1}{\partial \mathbf{x}_1 \partial \mathbf{x}_2}(t,\mathbf{x}), \ldots, \frac{\partial^2 u_1}{\partial \mathbf{x}_d^2}(t,\mathbf{x}), \ldots, \frac{\partial^r u_{d_2}}{\partial \mathbf{x}_d^r}(t,\mathbf{x}), \ldots\right)$, i.e., the evaluation of the candidate library at $(t, \mathbf{x})$.

For each identification task, we build a dataset S as follows: we select M spatial locations $\mathbf{x}_m \in \Omega$, $m = 1, \ldots, M$ and collect (through simulation or observation) the corresponding solutions $u(t_j, \mathbf{x}_m)$ at time instances $\{t_0, \ldots, t_J\}$, where $h_j = t_{j+1} - t_j$ is the time step and $t_0 = 0$. This results in the dataset

$$S = \{u(t_j, \mathbf{x}_m)\}_{j=0..J,\, m=1..M} \tag{5.3}$$

which consists of $(J+1) \times M$ observations and provides $N = J \times M$ training pairs $(u(t_j, \mathbf{x}_m), u(t_{j+1}, \mathbf{x}_m))$ for learning the underlying dynamics.

We denote by $\mathbf{U}_{prev}$ (resp. $\mathbf{U}_{next}$) the $N \times d_2$ matrix containing the first (resp. second) elements of these pairs. In addition, we define the $N \times |\Theta|$ matrix

$$\boldsymbol{\Theta}_u = \left(\Theta_u(t_j, \mathbf{x}_m)\right)_{j=0..J-1,\, m=1..M} \tag{5.4}$$

and the $N \times N$ matrix $\mathbf{H}$ containing $N \times M$ copies of the time-step vector $\mathbf{h} = (h_0, \ldots, h_{J-1})$. These notations are illustrated in Fig. 5.5. Finally, we denote by $\mathcal{H}$ the $N \times d_2$ matrix formed by $M \times d_2$ repetitions of $\mathbf{h}$.

5.2.3 Integration method

Euler-SINDy

Euler-SINDy relies on a forward (explicit) Euler one-step update to approximate temporal evolution. For an ODE

$$\frac{du}{dt}(t) = f(t, u(t))$$

the forward Euler step reads

$$u(t+h) \approx u(t) + h\, f(t, u(t))$$

which follows from a first-order Taylor expansion. The method has local truncation error of order $\mathcal{O}(h^2)$ and global error of order $\mathcal{O}(h)$. Despite its low order, the explicit Euler scheme is attractive because of its simplicity and low computational cost.

In Euler-SINDy, a single Euler step is used as the prediction operator inside the SINDy identification framework. For possibly irregular observation times t_j (with $h_j = t_{j+1} - t_j$), the state update at spatial location $\mathbf{x}_m$ is approximated by

$$u(t_{j+1}, \mathbf{x}_m) \approx u(t_j, \mathbf{x}_m) + h_j\, \Theta_u(t_j, \mathbf{x}_m)\, \boldsymbol{\alpha}$$

where $\Theta_u(t_j, \mathbf{x}_m) \in \mathbb{R}^{1 \times |\Theta|}$ evaluates the library of candidate functions at the current state and $\boldsymbol{\alpha} \in \mathbb{R}^{|\Theta| \times d_2}$ contains the coefficients for the d_2 governing equations. The identification problem then seeks the sparsest coefficient matrix that best matches observed transitions:

$$\min_{\boldsymbol{\alpha}} \sum_{j=0}^{J-1} \sum_{m=1}^{M} \left\| u(t_{j+1}, \mathbf{x}_m) - \left(u(t_j, \mathbf{x}_m) + h_j\, \Theta_u(t_j, \mathbf{x}_m)\boldsymbol{\alpha}\right) \right\|_2^2 \; + \; \lambda \|\boldsymbol{\alpha}\|_{\mathcal{F}}^2 \tag{5.5}$$

or equivalently in matrix form

$$\min_{\boldsymbol{\alpha}} \left\| \mathbf{U}_{\text{next}} - (\mathbf{U}_{\text{prev}} + \mathbf{H}\,\boldsymbol{\Theta}_{\boldsymbol{u}}\,\boldsymbol{\alpha}) \right\|_{\mathcal{F}}^2 + \lambda \|\boldsymbol{\alpha}\|_{\mathcal{F}}^2 \tag{5.6}$$

where $\|\cdot\|_{\mathcal{F}}$ is the Frobenius norm and λ is a ridge regularization parameter. While L_1 penalties (LASSO) are commonly used to promote sparsity, we employ Sequential Thresholded Ridge Regression (STRidge) to mitigate issues arising from highly correlated library terms and noisy measurements. Note that because forward Euler is only first-order accurate and can be unstable for large timesteps or stiff dynamics, its single-step approximation may be poor when observation gaps h_j are large; this limitation motivates methods that unroll multiple integration steps or use higher-order integrators.

RK4-SINDy

The Runge-Kutta family of methods improves upon the Euler method by taking several intermediate evaluations of the function within a single integration step, combining them to achieve higher accuracy. Notably, the explicit Euler method is a special case of this family, known as the first-order Runge-Kutta method (RK1), which uses only a single stage and provides first-order accuracy. In general, an s-stage Runge–Kutta method approximates

$$u(t+h) \approx u(t) + h \sum_{i=1}^{s} b_i k_i$$

where each stage k_i is computed as

$$\begin{aligned} k_1 &= f(t, u(t)) \\ k_2 &= f(t + c_2 h, u(t) + h a_{21} k_1) \\ &\vdots \\ k_s &= f\Big(t + c_s h, u(t) + h \sum_{j=1}^{s-1} a_{sj} k_j\Big) \end{aligned}$$

The coefficients a_{ij}, b_i, and c_i determine the specific Runge-Kutta scheme. The most widely used instance is the classical 4-stage Runge–Kutta method (RK4), which has local truncation error $\mathcal{O}(h^5)$ and global accuracy $\mathcal{O}(h^4)$. In RK4, the stages are defined as

$$\begin{aligned} k_1 &= f(t, u) \\ k_2 &= f\left(t + \frac{h}{2}, u + \frac{h}{2} k_1\right) \\ k_3 &= f\left(t + \frac{h}{2}, u + \frac{h}{2} k_2\right) \\ k_4 &= f\left(t + h, u + h k_3\right) \end{aligned}$$

and the update is

$$u(t+h) \approx u(t) + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

In RK4-SINDy [51], this RK4 approximation is used as the prediction step inside the sparse identification framework. Instead of updating the state with a single Euler

step (as in Euler-SINDy), RK4-SINDy replaces that prediction with the 4-stage Runge–Kutta approximation. This substitution reduces local truncation errors and makes the regression problem more accurate when the time step between observations is not too large.

In practice, the dictionary Θ is evaluated not only at the current observation but also at the intermediate RK4 stages. The coefficients in the linear combination are then adjusted according to the RK4 weights. This means that RK4-SINDy uses the same optimization procedure as Euler-SINDy but relies on a more accurate numerical integration scheme, which improves robustness when the sampling rate is relatively coarse.

For RK4-SINDy, the equivalent update replaces the Euler estimate with the classical 4-stage Runge–Kutta approximation. Let $h_j = t_{j+1} - t_j$ be the time step, and define the intermediate stages k_1, k_2, k_3, k_4 as

$$\begin{aligned} k_1 &= \Theta_u(t_j, \mathbf{x}_m)\,\boldsymbol{\alpha} \\ k_2 &= \Theta_u\Big(t_j + \frac{h_j}{2}, \mathbf{x}_m\Big)\,\boldsymbol{\alpha} \\ k_3 &= \Theta_u\Big(t_j + \frac{h_j}{2}, \mathbf{x}_m\Big)\,\boldsymbol{\alpha} \\ k_4 &= \Theta_u(t_j + h_j, \mathbf{x}_m)\,\boldsymbol{\alpha} \end{aligned}$$

The RK4 estimate of $u(t_{j+1}, \mathbf{x}_m)$ is then

$$u(t_{j+1}, \mathbf{x}_m) \approx u(t_j, \mathbf{x}_m) + \frac{h_j}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Using this, the RK4-SINDy optimization problem becomes

$$\min_{\boldsymbol{\alpha}} \sum_{j=0}^{J-1} \sum_{m=1}^{M} \left\| u(t_{j+1}, \mathbf{x}_m) - \Big(u(t_j, \mathbf{x}_m) + \frac{h_j}{6}(k_1 + 2k_2 + 2k_3 + k_4)\Big) \right\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_{\mathcal{F}}^2$$

In matrix form, letting $\mathbf{k}_i$ denote the concatenation of all stages for all spatial points, this reads

$$\min_{\boldsymbol{\alpha}} \left\| \mathbf{U}_{next} - \Big(\mathbf{U}_{prev} + \frac{H}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4)\Big) \right\|_{\mathcal{F}}^2 + \lambda \|\boldsymbol{\alpha}\|_{\mathcal{F}}^2$$

This is the RK4 analogue of the Euler-SINDy formula, with the key difference that the update now accounts for the four intermediate RK4 stages to improve accuracy and reduce truncation error.

5.2.4 Local and Global Truncation Errors

When analyzing numerical methods, it is important to distinguish between local truncation error (LTE) and global truncation error (GTE), as both measure different aspects of the accuracy of the method.

Local Truncation Error

Local truncation error refers to the error introduced in a single step of the numerical method, assuming that the solution at the previous step is exact. Formally, for a

method advancing from t_n to $t_{n+1} = t_n + h$, the LTE is defined as

$$\tau_{n+1} = \frac{u(t_{n+1}) - \phi(h, t_n, u(t_n))}{h},$$

where $u(t_{n+1})$ is the exact solution and $\phi(h, t_n, u(t_n))$ is the numerical update.

For the explicit Euler method, which updates the solution via

$$u_{n+1} = u_n + hf(t_n, u_n),$$

the exact solution can be expanded via a Taylor series:

$$u(t_{n+1}) = u(t_n) + hu'(t_n) + \frac{h^2}{2}u''(\xi_n),$$

for some $\xi_n \in [t_n, t_{n+1}]$. Subtracting the Euler update gives

$$u(t_{n+1}) - u_{n+1} = \frac{h^2}{2}u''(\xi_n),$$

so the LTE is $\tau_{n+1} = \mathcal{O}(h^2)$.

For RK4, the update involves four intermediate stages to approximate a Taylor expansion to fourth order:

$$u_{n+1} = u_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

and by comparing with the exact Taylor expansion, the LTE is $\mathcal{O}(h^5)$, which comes from the fifth-order term being the first neglected term in the expansion.

Global Truncation Error

Global truncation error accumulates the LTE over all $N = (T - t_0)/h$ steps taken to reach a final time T. If the LTE is $\mathcal{O}(h^{p+1})$ and the method is applied over $N = O(1/h)$ steps, the GTE scales as

$$\text{GTE} = N \cdot \mathcal{O}(h^{p+1}) = \mathcal{O}(h^p).$$

For Euler, $p = 1$, so the GTE is $\mathcal{O}(h)$, while for RK4, $p = 4$, so the GTE is $\mathcal{O}(h^4)$. This explains why RK4 maintains much higher accuracy over long time intervals compared to Euler for the same step size.

In summary, local truncation error measures the per-step accuracy of the method, derived from the first neglected term in the Taylor expansion, while global truncation error measures the accumulated error over the entire interval. Euler is simple but first-order accurate globally, while RK4 achieves fourth-order global accuracy, at the cost of more function evaluations per step.

5.2.5 Stability of Euler and RK4 Methods

The stability of a numerical solver for ordinary differential equations (ODEs) refers to its ability to produce accurate and reliable solutions over a long time interval, without the computed values of the solution becoming unbounded or oscillatory. In simple terms, a stable numerical solver prevents spurious growth of errors and ensures that the numerical solution behaves similarly to the true solution. A necessary

condition for stability is that the local truncation errors introduced at each step do not grow uncontrollably from one step to the next. This leads to the concept of absolute stability.

A numerical integration method is considered absolutely stable if, when applied to the linear test equation

$$u'(t) = \lambda u(t)$$

the numerical solution does not grow unbounded for a given time step h. Mathematically, this condition is expressed in terms of the method's amplification factor $R(z)$ as

$$|R(z)| \leq 1, \quad z = h\lambda$$

h is the time step and λ is the eigenvalue of the system's Jacobian matrix (the matrix of the partial derivatives of the system's with respect to the state variables). Here λ is represents the rate at which the true solution grows or decays. $R(z)$ characterizes how errors propagate between steps.

The region of absolute stability of a numerical method is defined as the set

$$\{z \in \mathbb{C} : |R(z)| \leq 1\}$$

consisting of all scaled eigenvalues $z = h\lambda$ for which the method produces a stable solution. This region is important because it determines the range of time steps that can be safely used without causing the numerical solution to grow unbounded.

Explicit Euler Method

When applied to the linear test equation $u'(t) = \lambda u(t)$, the explicit Euler method yields

$$u_{n+1} = u_n + h\lambda u_n = (1 + z)u_n$$

so that the stability function is

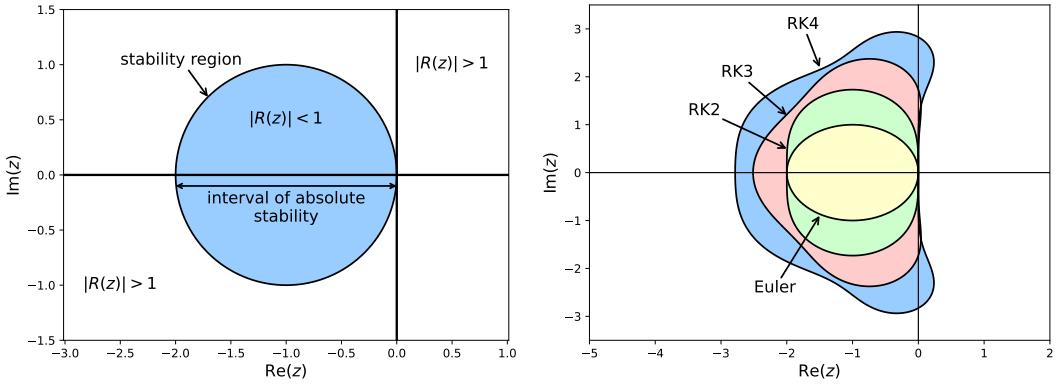
$$R(z) = 1 + z$$

The method is stable if $|1 + z| \leq 1$, which corresponds to a circular region in the complex plane centered at -1 with radius 1 (see Fig. 5.2.a). This indicates that explicit Euler remains stable only for eigenvalues with sufficiently negative real parts and only if the time step h is small enough to keep $z = h\lambda$ within the circle. For stiff problems, where the eigenvalues can be large and negative, the stability region is extremely limited, making the explicit Euler method unstable unless very small time steps are used.

Explicit Runge-Kutta 4 (RK4) Method

For the classical 4-stage Runge–Kutta method (RK4), applied to the linear test equation $u'(t) = \lambda u(t)$, the method computes four intermediate stages:

$$\begin{aligned}
k_1 &= \lambda u_n \\
k_2 &= \lambda \left(u_n + \frac{h}{2} k_1 \right) \\
k_3 &= \lambda \left(u_n + \frac{h}{2} k_2 \right) \\
k_4 &= \lambda \left(u_n + h k_3 \right)
\end{aligned}$$



(a) (b)

FIGURE 5.2: Illustration of stability regions for: (a) Euler method and (b) Runge-Kutta method. Adapted from [113].

and updates the solution as

$$u_{n+1} = u_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

By substituting the intermediate stages recursively and factoring out u_n, the stability function can be expressed as

$$R(z) = 1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \frac{z^4}{4!}$$

where $z = h\lambda$. This corresponds to the fourth-order Taylor expansion of e^z, reflecting the method's fourth-order accuracy. The region of absolute stability is defined by $|R(z)| \leq 1$ (see Fig. 5.2.b), which is considerably larger than that of the explicit Euler method. As a result, RK4 allows for larger time steps while maintaining stability, making it more robust for problems with moderate stiffness or larger eigenvalues. Nonetheless, since RK4 is still an explicit method, its stability region remains bounded, and it may be inadequate for highly stiff problems.

When comparing the two methods in terms of stability, the explicit Euler method exhibits a very limited stability region, which severely restricts the allowable time step and makes it highly susceptible to instability for stiff systems. In contrast, RK4 significantly enlarges the stability region, allowing for larger time steps while also providing higher-order accuracy. Both methods are explicit and therefore inherently bounded in their stability regions, meaning that for highly stiff problems, implicit methods are generally more appropriate. Additionally, RK4 benefits from lower truncation errors and improved error propagation, making it a more robust and accurate choice than Euler in most practical applications where stability and precision must be balanced.

5.2.6 Metric for evaluating the quality of a learned equation

As suggested in Weak SINDy [87], we evaluate the quality of a learned equation using two complementary metrics. The first is the so-called "shape" accuracy, defined as

$$Q(\boldsymbol{\alpha}) = \frac{TP}{TP + FN + FP}$$

where $\boldsymbol{\alpha}$ denotes the vector of coefficients associated with the candidate functions in the SINDy library. In other words, each entry of $\boldsymbol{\alpha}$ corresponds to the weight of a particular nonlinear term that may or may not be active in the governing equation. Here, TP is the number of correctly nonzero coefficients of $\boldsymbol{\alpha}$, FN is the number of coefficients that should be nonzero but were mistakenly set to zero, and FP is the number of coefficients incorrectly identified as nonzero. This metric evaluates whether the structure of the discovered equation matches the true one, independently of the precise coefficient values.

For example, consider the ground-truth system

$$\frac{du}{dt} = -0.5\,u + 0.1\,u^3$$

If the algorithm identifies exactly the same two active terms, u and u^3, then $TP = 2$, $FN = 0$, and $FP = 0$, so $Q(\boldsymbol{\alpha}) = 1$. In this case, one can then assess the numerical quality of the coefficients by computing the Mean Absolute Error (MAE) between the true solutions U and the simulated solutions $\tilde{U}$ generated from the learned equation.

If the algorithm instead recovers only one correct term, for example

$$\frac{du}{dt} = -0.5\,u$$

then $TP = 1$, $FN = 1$, $FP = 0$, yielding $Q(\boldsymbol{\alpha}) = 0.5$. This indicates partial structural recovery of the dynamics.

In another case, if the algorithm identifies the right terms but with incorrect coefficients, for example

$$\frac{du}{dt} = -1.2\,u + 0.3\,u^3$$

then $TP = 2$, $FN = 0$, $FP = 0$, so $Q(\boldsymbol{\alpha}) = 1$. However, the MAE will reveal significant quantitative errors in reproducing the trajectories, showing that both shape and coefficient accuracy are important for reliable discovery.

Finally, in the worst-case scenario where no relevant terms are discovered, such as

$$\frac{du}{dt} = 0.2\,\sin(u) + 0.05\,\cos(u)$$

we have $TP = 0$, $FN = 2$, $FP = 2$, resulting in $Q(\boldsymbol{\alpha}) = 0$. This signals a complete structural mismatch between the discovered model and the true dynamics.

Altogether, the metrics $Q(\boldsymbol{\alpha})$ and MAE together provide a robust evaluation framework: the first checks whether the discovered equation has the correct structure, while the second quantifies how accurately the learned dynamics reproduce the true system once the structure is correct.

5.3 Unrolled SINDy

Euler-SINDy relies on the observations $u(t_j, .)$ and the evaluation of the dictionary $\Theta_u(t_j, \mathbf{x}_m)$ at time t_j to predict the solution at time t_{j+1}. This approach works well when the data is densely sampled, meaning that the inter-observation time step $h_j = t_{j+1} - t_j$ is small relative to the characteristic time scales of the system dynamics, and when the dictionary Θ contains the correct terms needed to describe the dynamics, as illustrated in Fig. 5.3.a.

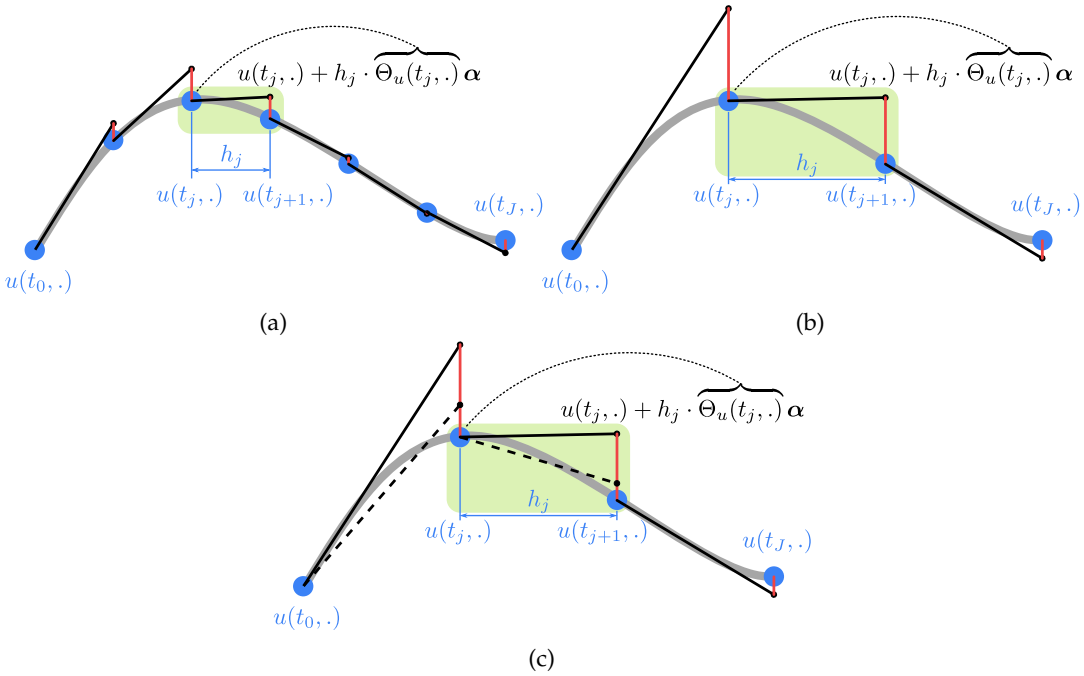


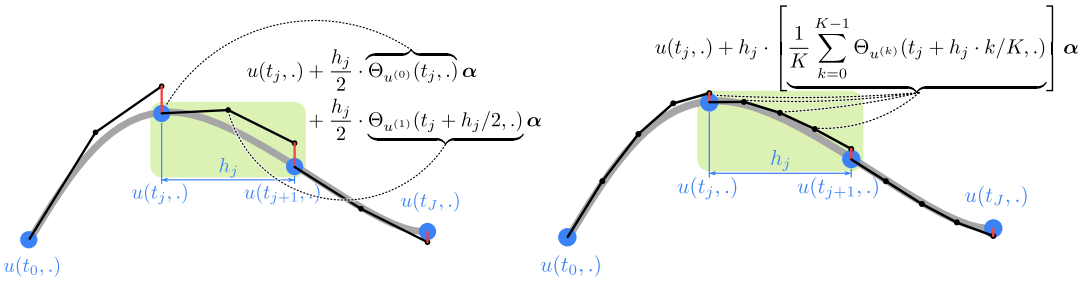
FIGURE 5.3: Illustration of Euler-SINDy with (a) dense observations, (b) sparse observations where the fitted line is tangent to the observation but errors accumulate, and (c) sparse observations where the dashed line is no longer using the true coefficients (no longer tangent to the trajectory) but the overall error is reduced.

However, when observations are scarce as shown in Fig. 5.3.b , even with a proper dictionary and the true coefficients, the residual error, the loss in Euler-SINDy, remains high. This high residual arises because a single-step Euler method cannot accurately capture the evolution of the system over a large time step. The optimization process compensates by assigning non-zero coefficients to extra terms in the dictionary that do not belong to the true underlying dynamics, in order to approximate the real system more closely, as illustrated in Fig. 5.3.c. The solution thus overfits: it reproduces the observed transitions in the dataset but does not correspond to the true governing equation and fails to generalize to new initial conditions or unseen time steps. This limitation is inherent not only to SINDy but also to its recent extensions such as iNeural-SINDy, which combines SINDy with physics-informed neural networks for noise robustness.

By design of these methods, the true coefficients are no longer minimizers of the loss when observations are separated by large time gaps as represented in Fig. 5.3.c. In contrast, our unrolled approach will explicitly incorporate multiple integration steps into the optimization, making the true coefficients minimizers of the loss even under sparse temporal sampling.

Changing the numerical method used to predict the next observation, as done in RK4-SINDy, improves the approximation for a single time step, since higher-order methods reduce local truncation errors, but this approach still remains fundamentally limited by the size of the inter-observation time step h_j. When h_j is large, even a fourth-order Runge-Kutta method may not capture the true dynamics between two consecutive measurements.

Our **Unrolled-SINDy** consists in unrolling the numerical method used to compute the prediction of the next observation. The main idea is to **decouple the numerical integration time step from the inter-observation time step** h_j. By introducing



(a) (b)

FIGURE 5.4: Illustration of the benefit of unrolling with K-step Unrolled Euler-SINDy: (a) $K = 2$ and (b) $K = 4$. Increasing K reduces the local truncation error at each numerical step.

intermediate integration steps, the method can maintain high accuracy even when the data is sparsely sampled, allowing the identification of the true coefficients.

Intuitively, when unrolled K times, the numerical method uses K intermediate steps of size h_j/K to compute the next observation, instead of relying on a single step in standard Euler-SINDy. This reduces the effective local truncation error while keeping the total integration interval fixed, enabling a more precise representation of the continuous-time dynamics. For simplicity of exposition, in Fig. 5.4.a we show the case of 2-Unrolled Euler-SINDy, which applies two steps of the Euler method, but the same principle can be generalized to RK4 or any explicit numerical method.

Similarly to Euler-SINDy, the first step computes a prediction using the current state $u(t_j,.)$ and the dictionary evaluated at the current time (denoted $\Theta_{u^{(0)}}$) but with the time step divided by 2. The second step then computes the final prediction using the first-step prediction and the dictionary evaluated at this intermediate state and time $t_j + h_j/2$ (denoted $\Theta_{u^{(1)}}$).

More generally, as depicted in Fig. 5.4.b K-Unrolled Euler-SINDy works by iterating K steps of size h_j/K. At each step $k = 0, \ldots, K-1$, the dictionary $\Theta_{u^{(k)}}$ is evaluated at the corresponding intermediate state and time $t_j + k \cdot h_j/K$, capturing how the dynamics evolve over the full interval in a piecewise manner. The final prediction $u^{(K)}$ can be expressed recursively as

$$u^{(k+1)} = u^{(k)} + \frac{h_j}{K} \cdot \Theta_{u^{(k)}}(t_j + k \cdot {}^{h_j}/_{K}, .) \cdot \boldsymbol{\alpha} \quad \text{and} \quad u^{(0)} = u(t_j, \cdot) \tag{5.7}$$

and, by developing the recursion and factorizing:

$$u^{(K)}(t_j + h_j, .) = \underbrace{u(t_j,.) + h_j \cdot \left[\frac{1}{K}\sum_{k=0}^{K-1} \Theta_{u^{(k)}}(t_j + k \cdot {}^{h_j}/_{K}, .)\right] \cdot \boldsymbol{\alpha}}_{\text{K-Unrolled Euler estimate}} \tag{5.8}$$

This formula resembles the one used in Euler-SINDy, but the dictionary is now effectively averaged over K intermediate time steps, improving the approximation of the temporal derivative over larger intervals. The same idea applies to RK4-SINDy, though the effective dictionary contains four times more terms with non-uniform weights due to the Runge-Kutta coefficients.

As we will show in the section 5.4, unrolling K times improves the numerical approximation: the Euler estimation in the factorized form benefits from a truncation error of order $\mathcal{O}(h^2/K)$. This is comparable to performing a single Euler step with a smaller time step of h/K while maintaining the same total interval h. Consequently,

Unrolled-SINDy allows accurate recovery of the true coefficients even when the inter-observation step h_j is large, reducing overfitting and improving the generalization of the discovered equations.

In addition, unrolling provides a natural way to handle non-uniform or adaptive time steps: the intermediate steps can be adjusted according to local dynamics or error estimates, making the approach flexible for irregularly sampled experimental data. Overall, Unrolled-SINDy extends the applicability of sparse identification methods to more challenging regimes, including sparse measurements, stiff dynamics, and high-dimensional systems.

5.3.1 Unrolled Euler-SINDy Algorithm

Similarly to what we have shown for Euler-SINDy (Eq. (5.6)), based on the prediction illustrated in Fig. 5.5.a), we can express the K-Unrolled Euler-SINDy algorithm as a penalized linear regression:

$$\min_{\boldsymbol{\alpha}} \left\| \mathbf{U}_{next} - \left(\mathbf{U}_{prev} + \mathbf{H} \cdot \left[\frac{1}{K} \sum_{k=0}^{K-1} \boldsymbol{\Theta}_{u^{(k)}} \right] \cdot \boldsymbol{\alpha} \right) \right\|_{\mathcal{F}}^2 + \lambda \|\boldsymbol{\alpha}\|_{\mathcal{F}}^2. \tag{5.9}$$

Fig. 5.5.b shows the prediction formula without the sum, for $K = 4$. The highlight in red emphasizes that the dictionary $\boldsymbol{\Theta}_{u^{(k)}}$, evaluated at intermediate steps $u^{(k)}$, has a dependency on $\boldsymbol{\alpha}$, the coefficients to be learned. K-Unrolled Euler-SINDy is a first solution, fast and simple to implement based on an **iterative closed-form solution** of the linear regression problem that corresponds to the core contribution presented in this chapter. Within one iteration, this effectively discards the dependency of the dictionary on $\boldsymbol{\alpha}$. The pseudo-code of K-Unrolled Euler-SINDy, implemented using `PyTorch` and `sklearn`, is presented in Algorithm 1. Notably, setting $K = 1$ recovers the original Euler-SINDy formulation.

Algorithm 1 K-Unrolled Euler-SINDy (iterative closed form)

Input: Training pairs $\{(u(t_j, \mathbf{x}_m), u(t_{j+1}, \mathbf{x}_m))\}_{j=0..J-1, m=1..M}$; time steps $\mathbf{t} = \{t_j\}_{j=0..J}$
K: nb of unrolling steps; λ: regularization parameter; $\mathcal{I}$: nb of iterations; α_{th}: threshold
$\mathcal{D}ict \in \mathbb{R}^{N \times d_2} \times \mathbb{R}^J \to \mathbb{R}^{N \times |\Theta|}$: a function to evaluate the dictionary
Initialize coefficient matrix $\boldsymbol{\alpha} \leftarrow \mathbf{0}$; $\mathbf{h}$, $\mathbf{H}$ and $\mathcal{H}$ using $\{t_j\}_j$ ▷ $\boldsymbol{\alpha} \in \mathbb{R}^{|\Theta| \times d_2}$
repeat $\mathcal{I}$ **times (or until stabilization of** $\boldsymbol{\alpha}$**)**
    Initialize $\tilde{\boldsymbol{\Theta}} \leftarrow \mathbf{0}$ ▷ $\tilde{\boldsymbol{\Theta}} \in \mathbb{R}^{N \times |\Theta|}$
    $\tilde{\mathbf{U}} \leftarrow \mathbf{U}_{prev}$ ▷ $\mathbf{U}_{prev} = \left(u(t_j, \mathbf{x}_m)\right)_{j=0..J-1, m=1..M} \in \mathbb{R}^{N \times d_2}$
    for $k = 0$ to $K - 1$ **do**
        $\boldsymbol{\Theta}_{\tilde{u}} \leftarrow \mathcal{D}ict(\tilde{\mathbf{U}}, \mathbf{t} + \frac{k}{K}\mathbf{h})$ ▷ $\boldsymbol{\Theta}_{\tilde{u}} = (\Theta_{u^{(k)}}(t_j + \frac{k}{K}h_j, \mathbf{x}_m))_{j,m} \in \mathbb{R}^{N \times |\Theta|}$
        $\tilde{\mathbf{U}} \leftarrow \tilde{\mathbf{U}} + \frac{1}{K} \cdot \mathbf{H} \cdot \boldsymbol{\Theta}_{\tilde{u}} \cdot \boldsymbol{\alpha}$
        $\tilde{\boldsymbol{\Theta}} \leftarrow \tilde{\boldsymbol{\Theta}} + \frac{1}{K} \cdot \boldsymbol{\Theta}_{\tilde{u}}$
    end for
    $\dot{\mathbf{U}} = (\mathbf{U}_{next} - \mathbf{U}_{prev}) \oslash \mathcal{H}$ ▷ $\dot{\mathbf{U}} \in \mathbb{R}^{N \times d_2}$; $\mathcal{H} \in \mathbb{R}^{N \times d_2}$; $\oslash$: element-wise division
    $\boldsymbol{\alpha} \leftarrow \left(\tilde{\boldsymbol{\Theta}}^\top \tilde{\boldsymbol{\Theta}} + \lambda \mathbf{I}_{|\Theta| \times |\Theta|}\right)^{-1} \tilde{\boldsymbol{\Theta}}^\top \dot{\mathbf{U}}$
    Hard thresholding [109]: $\alpha_{ij} = 0$ if $|\alpha_{ij}| < \alpha_{th}$
end
Output: Final sparse coefficient matrix $\boldsymbol{\alpha}$

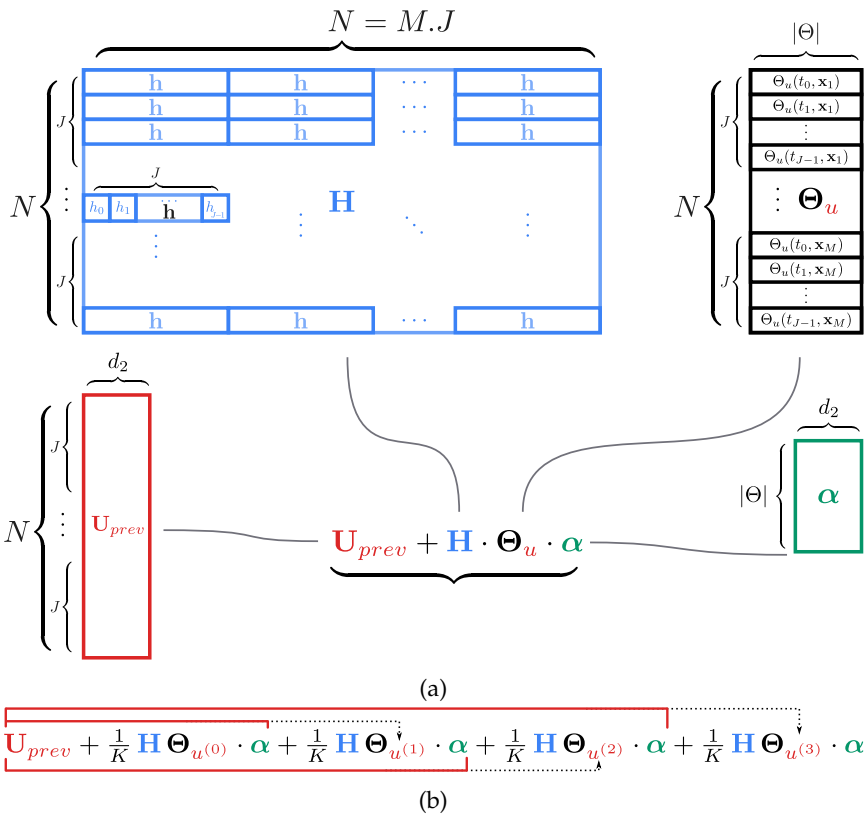


FIGURE 5.5: Illustration of the Euler-SINDy prediction process: (a) prediction with a single Euler step, and (b) prediction with four unrolled Euler-SINDy steps.

5.3.2 Unrolled RK4-SINDy Algorithm

The proposed unrolling approach could be applied to other integration scheme such as Runge-Kutta 4 (RK4). The equivalent of Eq. 5.8 for the factorized expression of our unrolled scheme for RK4 is defined as follows:

$$u^{(K)}(t_j + h_j, .) = \underbrace{u(t_j, .) + h_j \cdot \left[\frac{1}{K}\sum_{k=0}^{K-1} \Theta^{(k)}(t_j + k \cdot h_j/K, .)\right] \cdot \boldsymbol{\alpha}}_{\text{K-Unrolled RK4 estimate}} \tag{5.10}$$

where $\Theta^{(k)}$ is the weighted average estimate as computed with the RK4 method and used (with matrix notations) in line 13 in the following pseudo-code of our Unrolled RK4-SINDy Algorithm 2.

5.3.3 Unrolled *-SINDy-SGD

Instead of using the closed-form solution as used, another strategy consists in resorting to a stochastic gradient descent (SGD) approach.
To design the Unrolled Euler-SINDy-SGD (resp. Unrolled RK4-SINDy-SGD) stochastic gradient descent version of the algorithm, one can initialize the vector $\boldsymbol{\alpha}$ randomly, choose a learning rate η and repeatedly do stochastic gradient descent by running the algorithm where lines 13-14 in Algorithm 1 (resp. lines 17-18 in Algorithm 2) are replaced by a gradient step using the gradient obtained through auto-differentiation:

Algorithm 2 Unrolled RK4-SINDy

Input: Training pairs $\{(u(t_j, \mathbf{x}_m), u(t_{j+1}, \mathbf{x}_m))\}_{j=0..J-1, m=1..M}$; time steps $\mathbf{t} = \{t_j\}_{j=0..J}$
K: nb of unrolling steps; λ: regularization parameter; $\mathcal{I}$: nb of iterations; α_{th}: threshold
$\mathcal{D}ict \in \mathbb{R}^{N \times d_2} \times \mathbb{R}^J \to \mathbb{R}^{N \times |\Theta|}$: a function to evaluate the dictionary
Initialize coefficient matrix $\boldsymbol{\alpha} \leftarrow \mathbf{0}$; $\mathbf{h}$, $\mathbf{H}$ and $\mathcal{H}$ using $\{t_j\}_j$ ▷ $\boldsymbol{\alpha} \in \mathbb{R}^{|\Theta| \times d_2}$
repeat $\mathcal{I}$ **times (or until stabilization of** $\boldsymbol{\alpha}$**)**
    Initialize $\tilde{\boldsymbol{\Theta}} \leftarrow \mathbf{0}$ ▷ $\tilde{\boldsymbol{\Theta}} \in \mathbb{R}^{N \times |\Theta|}$
    $\tilde{\mathbf{U}} \leftarrow \mathbf{U}_{prev}$
    for $k = 0$ to $K - 1$ **do**
        $\boldsymbol{\Theta}_1 \leftarrow \mathcal{D}ict(\tilde{\mathbf{U}}, \mathbf{t} + \frac{k}{K}\mathbf{h})$
        $\boldsymbol{\Theta}_2 \leftarrow \mathcal{D}ict(\tilde{\mathbf{U}} + \frac{1}{2K} \cdot \mathbf{H} \cdot \boldsymbol{\Theta}_1 \cdot \boldsymbol{\alpha}, \mathbf{t} + \frac{k+1/2}{K}\mathbf{h})$
        $\boldsymbol{\Theta}_3 \leftarrow \mathcal{D}ict(\tilde{\mathbf{U}} + \frac{1}{2K} \cdot \mathbf{H} \cdot \boldsymbol{\Theta}_2 \cdot \boldsymbol{\alpha}, \mathbf{t} + \frac{k+1/2}{K}\mathbf{h})$
        $\boldsymbol{\Theta}_4 \leftarrow \mathcal{D}ict(\tilde{\mathbf{U}} + \frac{1}{K} \cdot \mathbf{H} \cdot \boldsymbol{\Theta}_3 \cdot \boldsymbol{\alpha}, \mathbf{t} + \frac{k+1}{K}\mathbf{h})$
        $\boldsymbol{\Theta} \leftarrow \frac{1}{6}(\boldsymbol{\Theta}_1 + 2\boldsymbol{\Theta}_2 + 2\boldsymbol{\Theta}_3 + \boldsymbol{\Theta}_4)$
        $\tilde{\mathbf{U}} \leftarrow \tilde{\mathbf{U}} + \frac{1}{K} \cdot \mathbf{H} \cdot \boldsymbol{\Theta} \cdot \boldsymbol{\alpha}$
        $\tilde{\boldsymbol{\Theta}} \leftarrow \tilde{\boldsymbol{\Theta}} + \frac{1}{K} \cdot \boldsymbol{\Theta}$
    end for
    $\dot{\mathbf{U}} = (\mathbf{U}_{next} - \mathbf{U}_{prev}) \oslash \mathcal{H}$ ▷ $\dot{\mathbf{U}} \in \mathbb{R}^{N \times d_2}$; $\mathcal{H} \in \mathbb{R}^{N \times d_2}$; $\oslash$: element-wise division
    $\boldsymbol{\alpha} \leftarrow \left(\tilde{\boldsymbol{\Theta}}^\top \tilde{\boldsymbol{\Theta}} + \lambda \mathbf{I}_{|\Theta| \times |\Theta|}\right)^{-1} \tilde{\boldsymbol{\Theta}}^\top \dot{\mathbf{U}}$
    Hard thresholding [109]: $\alpha_{ij} = 0$ if $|\alpha_{ij}| < \alpha_{th}$
end
Output: Final sparse coefficient matrix $\boldsymbol{\alpha}$

$$\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha} - \eta \cdot \nabla_{\boldsymbol{\alpha}} ||\mathbf{U}_{next} - \tilde{\mathbf{U}}||_{\mathcal{F}}^2$$

It then becomes unnecessary in the algorithm to keep track of $\tilde{\boldsymbol{\Theta}}$. If no unrolling is applied, using stochastic gradient descent with the Euler scheme recovers the standard Euler-SINDy algorithm, while using it with RK4 reduces to the standard RK4-SINDy algorithm.

In this work, we focus on the closed-form solution, which is computationally faster. However, we demonstrate later the unrolled approach on a neural-network-based SINDy variant (see section 5.4 for K Unrolled iNeural SINDy) to illustrate the versatility of the unrolling strategy.

5.4 Theoretical results: truncations error of Unrolled SINDy

The local truncation error of a numerical method is defined as the error induced for each approximation step. In the following, we derive an upper bound of this error when a K-unrolled scheme is applied on Euler, and then, more generally, on a s-stage Runge-Kutta method.

The purpose of these results is to quantify how unrolling the integration step affects the local truncation error. By splitting a full time step h into K smaller sub-steps, the effective error introduced at each step is reduced. This reduction in truncation error is a key factor behind the ability of unrolled *-SINDy methods to recover the true governing equations even when the sampling interval is relatively large.

Theorem 1. *The local truncation error suffered by the unrolled Euler estimate of Eq. (5.8) (assuming that $\forall j, h_j = h$) is on the order $\mathcal{O}\left(\frac{h^2}{K}\right)$ such that:*

$$\epsilon = \left(\frac{h^2}{2K}\right) \cdot \left| \frac{1}{K} \sum_{i=0}^{K-1} u''\left(t + \frac{hi}{K}\right) \right| \leq \left(\frac{h^2}{2K}\right) .M, \tag{5.11}$$

with the constant $M = \max_{t' \in [t, t+h]} |u''(t')|$.

Proof. By applying recursively K times Taylor's theorem, we get:

$$u(t+h) = u(t) + \frac{h}{K} \sum_{i=0}^{K-1} u'(t + \frac{hi}{K}) + \left(\frac{h}{K}\right)^2 \frac{\sum_{i=0}^{K-1} u''(t + \frac{hi}{K})}{2}. \tag{5.12}$$

Then applying recursively K times Euler's approximation, we get:

$$\tilde{u}(t+h) = u(t) + \frac{h}{K} \sum_{i=0}^{K-1} u'(t + \frac{hi}{K}). \tag{5.13}$$

From Eq. (5.12) and (5.13), we deduce that the local truncation error is equal to:

$$\epsilon = |u(t+h) - \tilde{u}(t+h)| = \left(\frac{h}{K}\right)^2 \cdot \left| \frac{1}{2} \sum_{i=0}^{K-1} u''(t + \frac{hi}{K}) \right|$$

□

Theorem. 1 states that as K tends to infinity, ϵ converges towards 0. This means that, provided that the dictionary Θ contains the correct terms, there exists an $\boldsymbol{\alpha}$ in Eq. (5.8) that corresponds to the governing equation and which allows a correct

prediction of the next observation. The method for finding these coefficients is covered in the following section. Before that, and since the unrolling can be applied on other numerical methods, the next theorem generalizes this result when the unrolling is embedded in an s-stage RK method.

Theorem 2. *Assume that* $|\frac{\partial f}{\partial u}| \leq L$. *Then there exists* $C \in \mathbb{R}^+$ *such that the error* ϵ *of a* K*-unrolled one-step of an* s*-stage Runge-Kutta method of order* p*, with a time-step* h *satisfies:*

$$\epsilon \leq \left(\frac{h}{K}\right)^p \frac{C}{L}\left(e^{Lh} - 1\right) \tag{5.14}$$

Proof. Consider the ODE

$$\frac{du}{dt} = f(u,t), \quad u(t) = u_0,$$

and let $u(t+h)$ be approximated using a K-unrolled Runge-Kutta method.

We denote by $\widetilde{u}_i$ the solution of the ODE starting at each intermediate stage such that

$$\widetilde{u}_0(t) = u_0, \quad \widetilde{u}_1\left(t + \frac{h}{K}\right) = u_1, \quad \widetilde{u}_2\left(t + 2\frac{h}{K}\right) = u_2, \ \ldots\ , \widetilde{u}_{K-1}\left(t + (K-1)\frac{h}{K}\right) = u_{K-1}.$$

After K steps, the approximation of the solution at $t+h$ is

$$u_K \approx \widetilde{u}_{K-1}(t+h).$$

The local error at each stage satisfies

$$e_i = \left|\widetilde{u}_{i-1}\left(t + i\frac{h}{K}\right) - u_i\right| \leq C\left(\frac{h}{K}\right)^{p+1},$$

where C is a constant depending on the method and the solution's smoothness [53, Thm. 3.1]. These local errors propagate to the final point, and we denote the propagated difference between consecutive intermediate solutions at $t+h$ as

$$E_i = \left|\widetilde{u}_i(t+h) - \widetilde{u}_{i-1}(t+h)\right|.$$

The global error of the K-unrolled method is then

$$\epsilon = \left|u(t+h) - u_K\right| = \left|\widetilde{u}_0(t+h) - u_K\right|,$$

and by standard Runge-Kutta error analysis [53, Thm. 3.4], it is bounded as

$$\epsilon \leq \left(\frac{h}{K}\right)^p \frac{C}{L}(e^{Lh} - 1),$$

where L is the Lipschitz constant of f and p is the order of the method.

For Euler's method, i.e., $p = 1$, Theorem 2 holds since the right-hand side of Eq. (5.11) is upper-bounded by that of Eq. (5.14). □

Together, Theorems 1 and 2 show that the local error decreases as the number of unrolled steps K increases, and that this effect holds for general Runge-Kutta methods of order p. This explains why unrolled Euler-SINDy or RK4-SINDy can recover the correct coefficients more reliably: by effectively reducing the integration step,

unrolling mitigates discretization errors that would otherwise corrupt the regression and lead to spurious terms or inaccurate predictions. In practice, this makes unrolled methods robust to coarser temporal sampling and improves the accuracy of derivative approximations used in the sparse regression.

Moreover, unrolling provides a systematic way to bridge the gap between the discrete observations and the underlying continuous dynamics, which is particularly important when dealing with real experimental data where measurements may be sparse or unevenly spaced in time. By refining the effective time resolution without requiring additional data, unrolled methods enable SINDy to operate in regimes that would normally compromise the identification of governing equations. However, it is worth noting that increasing K also increases the computational cost of the forward integration, and in noisy settings, very large K may amplify measurement noise if not properly regularized. Overall, unrolling offers a flexible and practical tool to enhance the reliability and accuracy of data-driven discovery of dynamical systems, making it applicable to both synthetic benchmarks and complex real-world problems.

5.5 Numerical experiments results

5.5.1 Comprehensive Study of the 2D-Cubic damped oscillator ODE

The goal of these experiments on the 2-dimensional cubic damped oscillator is to demonstrate how unrolling the numerical step can effectively delay the onset of instability in sparse regression methods, thereby allowing them to recover the correct governing equations for a longer range of step sizes. In classical SINDy with Euler discretization, when the step size becomes too large, numerical instabilities quickly accumulate. These instabilities manifest as errors in the regression stage, which either obscure the true active terms or introduce spurious ones. By unrolling, the solver approximates the dynamics across multiple smaller sub-steps within each integration step, which reduces discretization error and postpones the point at which instabilities dominate.

The system under consideration is the cubic damped oscillator, a two-dimensional nonlinear ODE given by:

$$\begin{cases} \dot{x}(t) = -0.1\, x^3(t) + 2.0\, y^3(t), \\ \dot{y}(t) = -2.0\, x^3(t) - 0.1\, y^3(t). \end{cases} \tag{5.15}$$

Here, $\dot{x}(t) = \frac{dx}{dt}(t)$ and $\dot{y}(t) = \frac{dy}{dt}(t)$ denote the time derivatives of the state variables $x(t)$ and $y(t)$, respectively. The cubic nonlinearities $x^3(t)$ and $y^3(t)$ capture the fact that the restoring and damping forces do not scale linearly with displacement but grow with the cube of the amplitude. The coefficients in front of these terms, such as -0.1 or 2.0, determine both the strength and the direction of each contribution: negative coefficients correspond to damping-like or stabilizing effects, while positive coefficients correspond to forcing or destabilizing effects. Together, the combination of cubic terms and their coefficients defines the overall nonlinear dynamics and the coupling between the x and y components.

This oscillator represents a prototypical example of coupled nonlinear systems with damping. Unlike a linear damped oscillator, where the restoring force is proportional to displacement, here the damping and restoring terms scale cubically. This results in richer dynamics, including nonlinear oscillations and strong coupling between the x and y components. While not a model of a specific physical system, the

cubic damped oscillator is often used as a benchmark for testing numerical methods and system identification algorithms. Its nonlinear structure makes it an excellent candidate to evaluate whether a method can correctly isolate relevant cubic terms without introducing spurious ones.

Although the equations appear simple at first glance, their nonlinear coupling makes the system highly sensitive: even small numerical or modeling errors can quickly amplify, causing the solution to deviate significantly from the true trajectory. This sensitivity highlights the importance of robust numerical solvers and sparse regression techniques, making the cubic damped oscillator a valuable testbed for evaluating stability, accuracy, and term selection in system identification. In applied contexts, similar nonlinear forms arise in systems such as mechanical vibrations, coupled electrical circuits, and interacting populations with feedback, illustrating the broad relevance of this benchmark.

Data generation. To generate training data, we simulate trajectories using the adaptive solver `solve_ivp` from the `SciPy` library with its default Runge-Kutta (RK45) integrator. The adaptive solver ensures that numerical integration errors remain negligible compared to the errors introduced later by discretization in SINDy. We collect a dataset S consisting of 50000 points over the time domain $[0, 10]$, using a very fine integration step of $h = 2 \times 10^{-4}$. These finely resolved trajectories can then be downsampled to emulate coarser step sizes, which allows us to systematically test the stability and accuracy of Euler-SINDy and Unrolled-SINDy under varying integration granularities. The choice of initial conditions, $x(0) = -0.488$ and $y(0) = 1.096$, is made so that the trajectory explores nontrivial regions of the phase space, rather than quickly decaying to a trivial equilibrium. This ensures that the regression problem is non-degenerate and that the full cubic structure of the dynamics is excited in the observed data.

Training protocol. For the regression stage, the sparsity threshold α_{th} is set to 0.05 and the regularization parameter λ to 10^{-2}. The threshold α_{th} controls the minimum coefficient magnitude that is retained in the sparse model, effectively filtering out spurious small coefficients that may arise from noise or discretization. The regularization λ prevents overfitting and stabilizes the regression process by penalizing large coefficients. The dictionary of candidate terms considered in the regression corresponds to all monomials up to degree four in the variables x and y:

$$\{1,\ x,\ y,\ x^2,\ xy,\ y^2,\ x^3,\ x^2y,\ xy^2,\ y^3,\ x^4,\ x^3y,\ x^2y^2,\ xy^3,\ y^4\}.$$

This dictionary includes both the cubic monomials present in the true system, namely x^3 and y^3 in Eq. (5.15), as well as higher-order and mixed terms that should ideally be eliminated by the sparsity-promoting regression. Both the parameters and the dictionary are chosen to balance accuracy and robustness while allowing the method to recover the governing cubic interactions of the oscillator.

This experimental setup provides a controlled environment where the benefits of unrolling can be clearly observed. The cubic damped oscillator challenges the methods with strongly nonlinear terms, while the use of fine-resolution data ensures that observed instabilities in recovery are attributable to the discretization and identification procedure, rather than numerical simulation errors. At the same time, the oscillator itself provides a meaningful benchmark system that captures the essential features of nonlinear coupled dynamics encountered in many applied settings.

TABLE 5.1: Robustness of Euler-SINDy, RK4-SINDy and their unrolled versions on cubic damped oscillator (Eq. 5.15), with an increasing time step h and decreasing number of learning pairs N. When the method fails to recover the governing equations (*i.e.* $Q(\boldsymbol{\alpha}) < 1$), only the number of wrong additional terms is indicated.

h (N)	Euler-SINDy	50 Unrolled Euler-SINDy	RK4-SINDy	10 Unrolled RK4-SINDy
$2{\cdot}10^{-4}$ ($N = 5{\cdot}10^4$)	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$
$2{\cdot}10^{-3}$ ($N = 5{\cdot}10^3$)	$-0.101x^3 + 1.994y^3$ $-1.995x^3 - 0.101y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$
$2{\cdot}10^{-2}$ ($N = 500$)	$-0.124x^3 + 1.986y^3$ $-1.993x^3 - 0.121y^3$	$-0.098x^3 + 1.995y^3$ $-1.995x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.995x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.995x^3 - 0.099y^3$
$4{\cdot}10^{-2}$ ($N = 249$)	$-0.124x^3 + 1.985y^3$ +1 $-1.983x^3 - 0.122y^3$ +1	$-0.099x^3 + 1.995y^3$ $-1.995x^3 - 0.100y^3$	$-0.098x^3 + 1.995y^3$ $-1.995x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.995x^3 - 0.099y^3$
$1{\cdot}10^{-1}$ ($N = 100$)	$-0.252x^3 + 1.900y^3$ +3 $-1.923x^3 - 0.230y^3$ +2	$-0.100x^3 + 1.993y^3$ $-1.994x^3 - 0.101y^3$	$-0.098x^3 + 1.994y^3$ $-1.994x^3 - 0.099y^3$	$-0.098x^3 + 1.994y^3$ $-1.994x^3 - 0.099y^3$
$4{\cdot}10^{-1}$ ($N = 24$)	$-0.388x^3 + 1.313y^3$ +9 $-1.420x^3 - 0.595y^3$ +7	$-0.108x^3 + 1.989y^3$ $-1.984x^3 - 0.107y^3$	$-0.099x^3 + 1.987y^3$ +3 $-1.997x^3 - 0.108y^3$ +1	$-0.098x^3 + 1.993y^3$ $-1.985x^3 - 0.100y^3$
$5{\cdot}10^{-1}$ ($N = 19$)	$-0.525x^3 + 1.260y^3$ +9 $-1.446x^3 - 0.561y^3$ +9	$-0.114x^3 + 1.990y^3$ $-1.985x^3 - 0.110y^3$	$-0.119x^3 + 1.961y^3$ +1 $-1.942x^3 - 0.113y^3$ +2	$-0.100x^3 + 1.994y^3$ $-1.989x^3 - 0.101y^3$
$6{\cdot}10^{-1}$ ($N = 16$)	$-0.681x^3 + 1.067y^3$ +11 $-1.330x^3 - 0.461y^3$ +11	$-0.114x^3 + 1.989y^3$ $-1.978x^3 - 0.116y^3$	$-0.123x^3 + 1.952y^3$ +4 $-1.872x^3 - 0.169y^3$ +4	$-0.100x^3 + 1.989y^3$ $-1.988x^3 - 0.103y^3$

Governing equation recovery under reduced temporal resolution

From the original dataset of 50000 points evenly spaced by $2 \cdot 10^{-4}$, we construct several sub-problems by systematically increasing the time step h from $2 \cdot 10^{-4}$ up to $6 \cdot 10^{-1}$. As the time step grows, the number of available training pairs $(u(t), u(t+h))$ decreases, and each pair represents a larger temporal jump. This creates a more challenging scenario for regression-based identification methods: the dynamics between $u(t)$ and $u(t+h)$ are less directly observable, and a single step of standard numerical approximation may no longer be sufficient to capture the true evolution of the system.

We evaluate Euler-SINDy, RK4-SINDy, and their unrolled variants on these sub-problems, reporting the results in Table 5.1 and in Table 5.2. For compactness, we present the results associated with the unrolling value K yielding the best training error as the behavior with respect to K is stable as illustrated in Table 5.3. Due to their volume, the full tables have been moved to the appendix B.2. The numbers shown in red in the table correspond to additional terms that are absent in the true equation.

The results illustrate the limitations of standard SINDy approaches as the time step increases. Euler-SINDy quickly fails to recover the correct equation as soon as h reaches 4.10^{-2}. RK4-SINDy is more robust, delaying failure until $h = 4 \cdot 10^{-1}$, but eventually also suffers from instabilities and introduces spurious terms when the number of training pairs becomes too small or the time step too large.

In contrast, both Unrolled Euler-SINDy and Unrolled RK4-SINDy exhibit remarkable robustness across all time steps. By subdividing each integration step into K smaller sub-steps of size h/K, these methods effectively reduce local truncation errors, yielding more accurate approximations of the derivative even for widely spaced observations. This allows the unrolled methods to recover the underlying equations almost perfectly, even from as few as 16 training pairs when $h = 6 \cdot 10^{-1}$. The unrolled scheme compensates for the sparsity of data by effectively "filling in" the missing temporal information between successive observations, which standard Euler or RK4 approaches cannot do.

Fig. 5.6.a and Fig. 5.6.b visually illustrate this effect by comparing the equations learned by Euler-SINDy, RK4-SINDy and their unrolled variants to the ground truth defined in Eq. (5.15). The differences are striking: Euler-SINDy fails to capture the

TABLE 5.2: Robustness of Euler-SINDy, RK4-SINDy, and their unrolled variants on cubic damped oscillator (Eq. 5.15), evaluated with increasing time step h and decreasing number of training pairs N. The figure shows the ℓ_1 loss measuring the difference between the true and predicted coefficients.

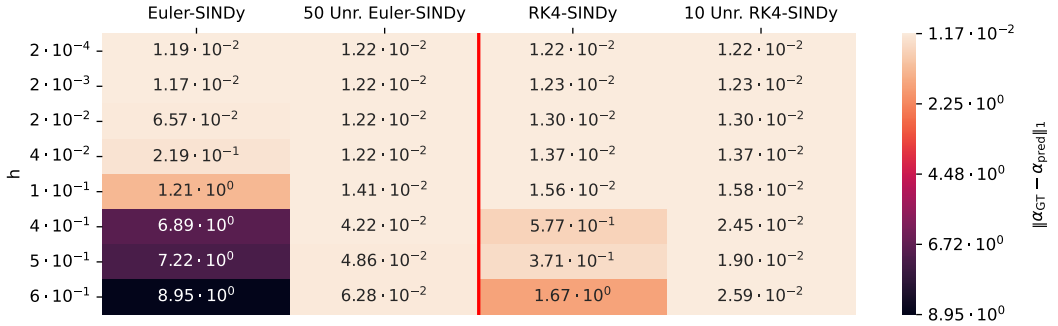
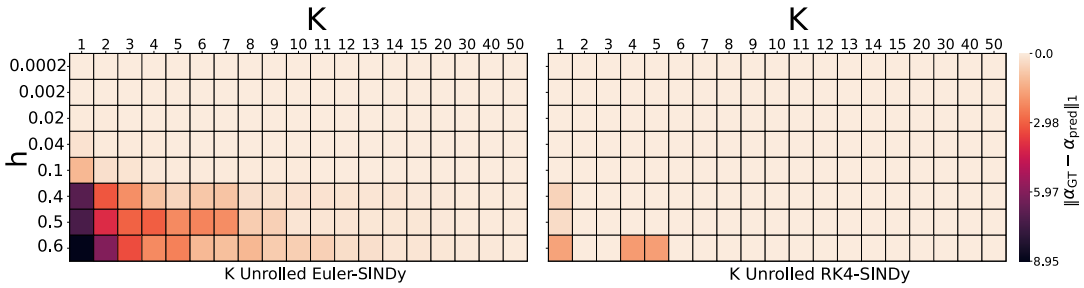


TABLE 5.3: Accuracy of K-Unrolled SINDy with different unrolling depths K and observation step sizes h. Lighter colors indicate more accurate recovery (details and exact values in B.2).

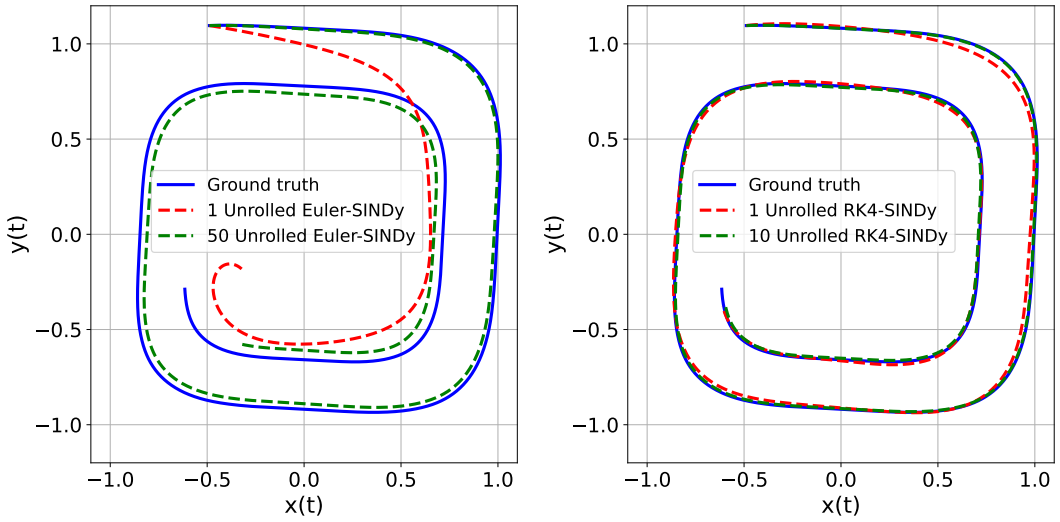


cubic interactions entirely at large time steps, producing false coefficients or missing key terms. RK4-SINDy shows a better ability to recover some of the terms correctly but still introduces multiple incorrect terms. It demonstrates that higher-order integration alone cannot fully overcome the issues caused by sparse observations. In contrast, the unrolled variants maintain high fidelity to the true equations, clearly showing that unrolling is a highly effective strategy for mitigating both the loss of information due to large time steps and the reduction in the number of available training pairs.

Regarding Fig. 5.6.b, it is worth noting that visually RK4-SINDy solution is similar to the ground truth although the equation recovered is incorrect as shown in Tab. 5.1. This illustrates that when observations are scarce, the true coefficients are no longer minimizers of the loss. The model then tries reduce the error by assigning non-zero coefficients to extra terms in the dictionary that do not belong to the true underlying dynamics to be able to approximate the real equation more closely, as displayed in Fig. 5.3.c.

Governing equation recovery under reduced temporal resolution with constant training data

We have seen in Table 5.1 and Table 5.2 that Euler-SINDy struggles to recover the true governing equation as h increases. However, in that earlier experiment, the number



(a) (b)

FIGURE 5.6: Comparison of the cubic damped oscillator solutions (Eq. (5.15)) with trajectories obtained from the learned ODEs using (a) Euler-SINDy and (b) RK4-SINDy, including their unrolled variants, at $h = 0.6$. RK4-SINDy produces a visually accurate solution despite recovering an incorrect equation (Tab. 5.1). When observations are scarce ($N = 16$), the model compensates by adding additional terms to match the true dynamics.

of training pairs N also decreased as h increased. One might think the inability to recover the true governing equation could come not only from integration errors but also from a lack of training data. In this section, the goal is to isolate the effect of the time step size itself by keeping N large and constant.

The data generation process follows the procedure described in the previous section, where a reference trajectory is obtained using `solve_ivp`. The SINDy configuration remains the same as previously described. What differs here is the construction of sub-trajectories for training designed to keep the number of observations approximately constant.

From the reference trajectory, we construct multiple sub-trajectories corresponding to larger effective step sizes h. Each sub-trajectory consists of pairs $(u(t), u(t+h))$, representing the system state at time t and after a time step h.

For example, suppose the reference trajectory contains points

$$u_0, u_1, u_2, u_3, u_4, u_5, \ldots$$

with a spacing h_0. To generate a dataset with effective step size $2h_0$, we could downsample by keeping every second point: $u_0, u_2, u_4, \ldots$, but this would ignore the remaining points. Instead, we create two interleaved sub-trajectories:

$$(u_0, u_2, u_4, \ldots), \quad (u_1, u_3, u_5, \ldots)$$

so that all points from the reference are reused.

This procedure generalizes to any desired step size h: multiple sub-trajectories are generated so that the entire reference trajectory is covered, and each dataset contains roughly the same number of training pairs. In this way, we increase the effective spacing between observations without reducing the dataset size or discarding

TABLE 5.4: Robustness of Euler-SINDy and its unrolled version on Eq. 5.15, with an increasing time step h and a constant number of learning pairs $N = 50000$. When Euler-SINDy fails to recover the governing equations (*i.e.* $Q(\boldsymbol{\alpha}) < 1$), only the number of wrong additional terms is indicated.

h	**Euler-SINDy**	**50 Unrolled Euler-SINDy**
$h = 2e-04$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$
$h = 4e-04$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$
$h = 1e-03$	$-0.099x^3 + 1.995y^3$ $-1.996x^3 - 0.100y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$
$h = 2e-03$	$-0.101x^3 + 1.994y^3$ $-1.995x^3 - 0.101y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$
$h = 2e-02$	$-0.124x^3 + 1.986y^3$ $-1.993x^3 - 0.121y^3$	$-0.098x^3 + 1.995y^3$ $-1.996x^3 - 0.099y^3$
$h = 3e-02$	$-0.135x^3 + 1.989y^3$ $-1.988x^3 - 0.117y^3{}_{+1}$	$-0.099x^3 + 1.995y^3$ $-1.995x^3 - 0.099y^3$
$h = 5e-02$	$-0.129x^3 + 1.981y^3{}_{+1}$ $-1.979x^3 - 0.126y^3{}_{+1}$	$-0.099x^3 + 1.995y^3$ $-1.995x^3 - 0.100y^3$
$h = 1e-01$	$-0.247x^3 + 1.948y^3{}_{+8}$ $-1.983x^3 - 0.223y^3{}_{+10}$	$-0.100x^3 + 1.994y^3$ $-1.995x^3 - 0.101y^3$
$h = 2e-01$	$-0.371x^3 + 1.834y^3{}_{+11}$ $-1.913x^3 - 0.350y^3{}_{+12}$	$-0.103x^3 + 1.993y^3$ $-1.994x^3 - 0.103y^3$
$h = 3e-01$	$-0.465x^3 + 1.648y^3{}_{+11}$ $-1.786x^3 - 0.478y^3{}_{+12}$	$-0.105x^3 + 1.991y^3$ $-1.993x^3 - 0.105y^3$
$h = 4e-01$	$-0.525x^3 + 1.406y^3{}_{+9}$ $-1.601x^3 - 0.598y^3{}_{+13}$	$-0.108x^3 + 1.990y^3$ $-1.992x^3 - 0.107y^3$

information.

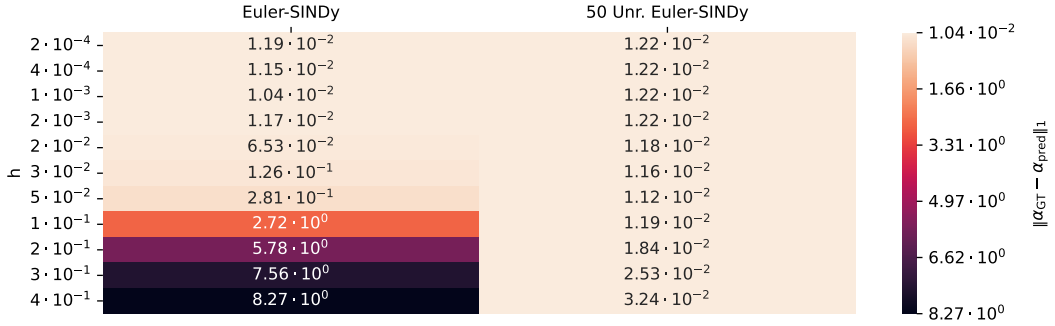
Both Euler-SINDy and Unrolled Euler-SINDy are then trained on these sub-trajectories, and the results are presented in Table 5.4 and in Table 5.5, where the best number K of unrolling steps is reported for each h. The results highlight several important aspects:

First, for Euler-SINDy without unrolling, the ℓ_1 between the true and predicted coefficients deteriorates rapidly as h increases. In particular, for $h = 5 \cdot 10^{-2}$, it fails to recover the correct governing equations. This failure is directly linked to the local truncation error of the Euler method, which grows as $\mathcal{O}(h^2)$, meaning that discretization errors accumulate quickly when the step size is too large.

Second, the unrolled variant remains robust under much larger time steps. For example, 50-Unrolled Euler-SINDy can still identify the correct dynamics where standard Euler-SINDy fails. The reason is that unrolling effectively simulates integration with a finer internal step h/K, reducing the local truncation error. This implicit reduction in the effective time step size allows the model to retain accuracy while still training on pairs separated by a large h in the observed data.

This confirms that the failure of Euler-SINDy to recover the true governing equation at large h is not simply a consequence of having fewer training pairs. Even when the number of pairs is held constant, Euler-SINDy deteriorates rapidly as h increases, showing that the main bottleneck lies in the discretization error inherent to the Euler scheme. RK4-SINDy alleviates this issue through its higher-order accuracy, but without unrolling, it still fails to cope with very large time steps. In contrast, the unrolled RK4-SINDy formulation nearly recovers the governing equations exactly, as

TABLE 5.5: Robustness of Euler-SINDy and its unrolled variant on Eq. 5.15, evaluated with increasing time step h and constant number of learning pairs $N = 50000$. The figure reports the ℓ_1 loss between the true coefficients and the predicted coefficients.



it effectively reduces error by averaging intermediate dictionary evaluations across K sub-steps. This demonstrates how unrolling not only stabilizes the regression but also preserves the true dynamics.

These results highlight the central role of unrolling in making SINDy robust to large sampling interval. In real-world scenarios, high-frequency measurements are often unattainable due to hardware constraints, acquisition costs, or experimental limitations. Unrolled formulations provide a practical solution by internally reconstructing finer-scale dynamics from sparse data, thereby reducing reliance on dense temporal sampling. This makes unrolled SINDy a powerful framework for modeling complex systems where only limited temporal resolution can be achieved.

The Role of Numerical Stability in Discovering Governing Equations

Both Euler-SINDy and RK4-SINDy are explicit schemes that embed a numerical integrator into the iterations of equation discovery. As discussed in Section 5.2.4, their truncation errors arise solely from the approximation of time derivatives and are independent of the underlying physics. However, beyond truncation error, the stability properties of these methods also play a decisive role. Following the analysis of Section 5.2.5, we assess stability through the eigenvalues of the Jacobian, which define the complex points $z = h\lambda$ whose location relative to the absolute stability region determines whether the numerical solution remains bounded.

From Eq. (5.15), we obtain the two eigenvalues $\lambda_1 = -0.2159 + 3.206i = \overline{\lambda_2}$. As they are conjugate of each other, it follows that $|R(h\lambda_1)| = |R(h\lambda_2)|$ for any $h \in \mathbb{R}$. We report in the two stability regions of Fig. 5.7 the complex number $z_j = h\lambda_j$ for the different step sizes h used in Table 5.1. We use green dots when the method succeeds in recovering the equations and red dots otherwise. By "succeeds", we mean that the ratio of true positive terms $Q(\boldsymbol{\alpha}) = 1$ (as defined in Section 5.2.6) and the MAE is close to 0.

From Eq. (5.15), the system has two eigenvalues

$$\lambda_1 = -0.2159 + 3.206\, i = \overline{\lambda_2},$$

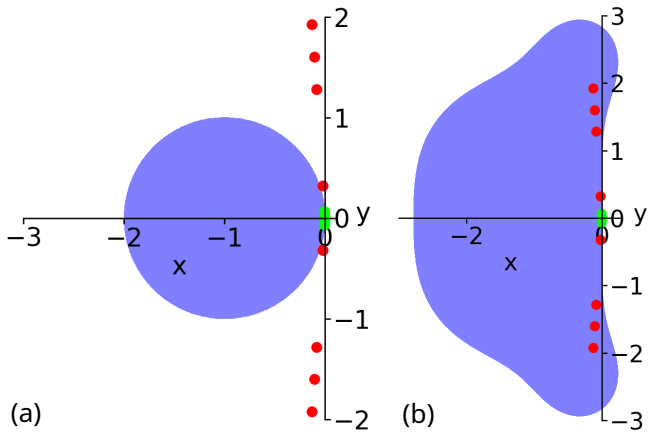


FIGURE 5.7: Stability zones (in blue) for (a) Euler and (b) RK4 methods. The points $z_j = h\lambda_j$ are shown in green when the governing equation is successfully recovered, and in red when it is not. The dots represents different value of unrolling depth K

where i denotes the imaginary unit. Since the eigenvalues are complex conjugates, their corresponding amplification factors satisfy

$$|R(h\lambda_1)| = |R(h\lambda_2)|, \quad \forall h \in \mathbb{R}.$$

In Fig. 5.7, we plot the scaled eigenvalues

$$z_j = h\lambda_j$$

for the various time steps h listed in Table 5.1, overlaying them on the respective stability regions of the numerical methods.

We can note that all the green dots are inside the stability regions. This means that for the equation discovery method to perform well, a **necessary condition** is that it is stable for the considered step size. This explains why Euler-SINDy fails dramatically at $h = 0.6$ on Fig. 5.6.a, its corresponding red dot being the farthest from the blue region. However, this **condition is not sufficient** as it does not guarantee that the regression within SINDy will recover the governing equation. This is illustrated with the red dots inside the region for RK4-SINDy.

Because our unrolled versions benefit of an implicit smaller step size h/K, the corresponding complex numbers satisfy the constraint $|R(z)| \leq 1$ much more easily, justifying why they work much better. The shapes of the stability regions hint that choosing between $4K$-Unrolled Euleur-SINDy and K-Unrolled RK4-SINDy (which are using the same number of dictionary evaluations) might depend on the equation (none of the 4-times-bigger circle of Euler and the RK4 region is included in the other) that is to be recovered.

Robustness to corrupted data

It is important to note that Unrolled-SINDy, whether implemented with Euler or RK4 integration, was primarily developed to mitigate issues arising from large time steps and sparse temporal sampling, rather than to explicitly handle noisy measurements. To investigate its behavior under measurement noise, we perform experiments on the two-dimensional cubic damped oscillator by adding Gaussian noise of increasing amplitude to the simulated trajectories.

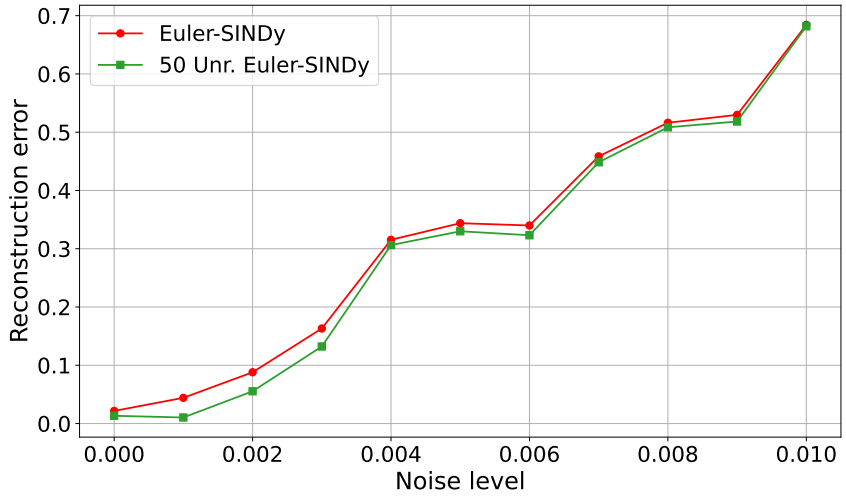


FIGURE 5.8: Evolution of the reconstruction error of Euler-SINDy and 50 Unrolled Euler-SINDy on the 2-dimensional cubic damped oscillator as the data is more and more corrupted with a gaussian noise.

Specifically, the noisy observations are generated as

$$\tilde{x}(t) = x(t) + \eta_x(t), \quad \tilde{y}(t) = y(t) + \eta_y(t),$$

where $\eta_x(t) \sim \mathcal{N}(0, \sigma_x^2)$ and $\eta_y(t) \sim \mathcal{N}(0, \sigma_y^2)$ are independent Gaussian random variables with zero mean and standard deviations

$$\sigma_x = \sigma \cdot \text{std}(x(t)), \quad \sigma_y = \sigma \cdot \text{std}(y(t)).$$

The parameter σ controls the relative noise amplitude, with larger values corresponding to more heavily corrupted data.

Figure 5.8 summarizes the ability of standard Euler-SINDy and Unrolled Euler-SINDy to recover the governing equation when the data is noisy. The plot reports the reconstruction error as a function of the noise level σ. To compute this error, we rerun the numerical solver used to generate the reference trajectory, but now using the coefficients $\boldsymbol{\alpha}$ identified by each SINDy variant to generate a predicted trajectory. The reconstruction error is then quantified as the root-mean-square error (RMSE) between the ground truth trajectory and the predicted trajectory. Several key observations can be made. When the noise level is relatively small, the unrolled variant maintains a clear advantage: the use of intermediate steps allows the method to leverage partial temporal information and reduce discretization errors, which helps in accurately estimating the derivatives and recovering the underlying governing equations. However, as the noise amplitude increases, this advantage diminishes. Intermediate steps themselves become corrupted, propagating the noise through the unrolled integration and reducing the accuracy of derivative approximations.

At a certain noise threshold (around $\sigma = 0.01$ in these experiments), both Euler-SINDy and Unrolled Euler-SINDy perform similarly, indicating that unrolling no longer provides a meaningful improvement. At this point, the magnitude of the noise overwhelms the benefit of using finer intermediate steps, and the regression procedure cannot reliably distinguish the true dynamics from random fluctuations.

This behavior highlights an important limitation of the unrolling approach. While

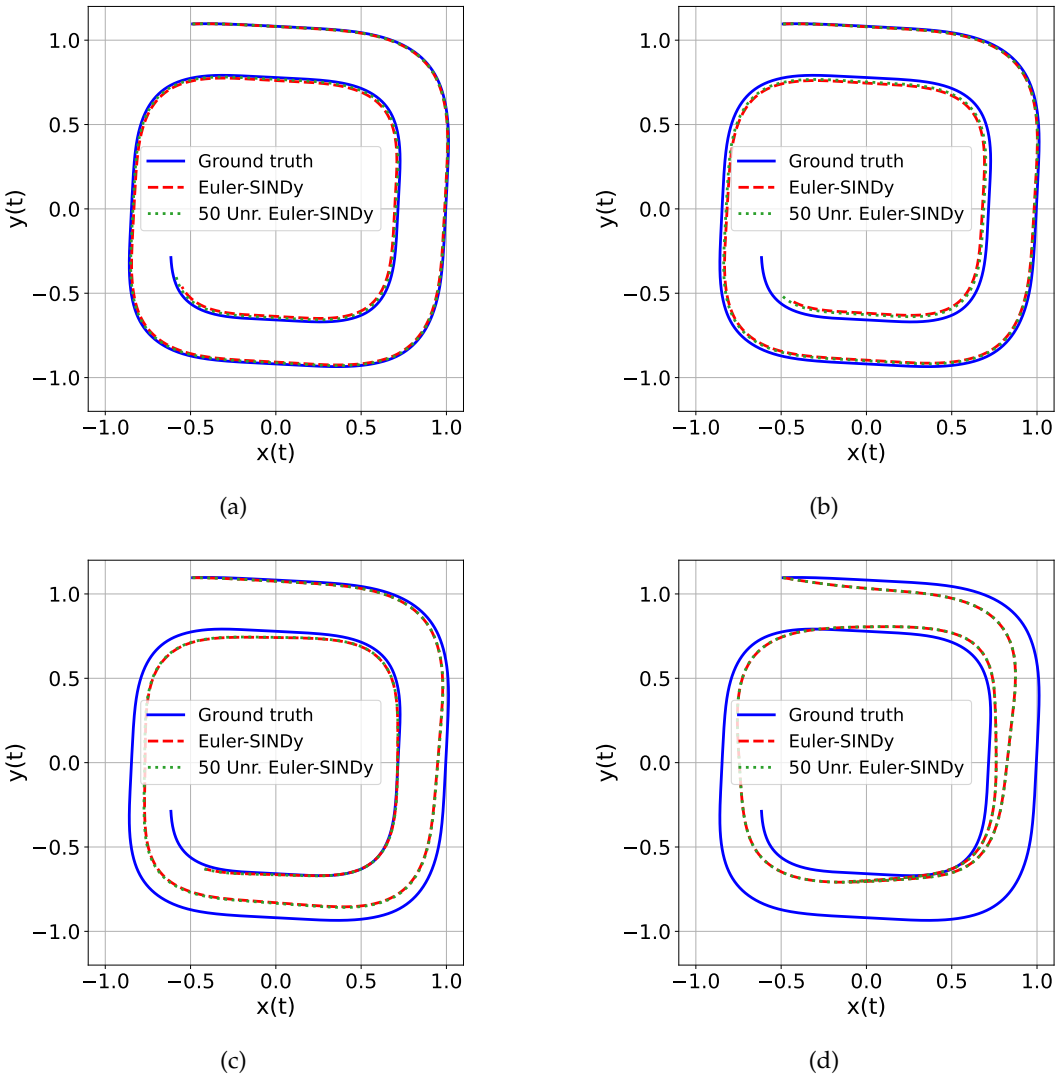


FIGURE 5.9: Impact of the noise on the capacity of the methods to recover the 2-dimensional cubic damped oscillator (with $h = 0.02$) for increasing noise rates: a) $\sigma = 2 \cdot 10^{-2}$, b) $\sigma = 3 \cdot 10^{-2}$, c) $\sigma = 5 \cdot 10^{-2}$, d) $\sigma = 1 \cdot 10^{-1}$.

it is highly effective for mitigating the effects of large time steps, it does not inherently provide robustness to noisy data. However, it remains a versatile approach that can be integrated into state-of-the-art PDE discovery methods to benefit from their noise robustness. In Section 5.5.5, we specifically demonstrate this by integrating the unrolling approach with iNeural-SINDy [48] in order to recover the true governing equation in noisy and scarce data cases.

Gradient descent execution time measurement.

Training protocol To gain insight into the computational costs of the proposed methods, we systematically measured the execution time of the SGD-based SINDy models. Specifically, training was conducted for 600 epochs with mini-batches of size 100. The optimization process was initialized using the RAdam optimizer, chosen for its ability to stabilize training in the early stages. A learning rate of 5×10^{-3} and an ℓ_2 regularization term with $\lambda = 10^{-2}$ were employed to balance

convergence speed and coefficient stability. To promote sparsity in the recovered model, a thresholding operation was performed every 200 epochs: coefficients in the matrix $\boldsymbol{\alpha}$ with magnitude smaller than $\alpha_{th} = 5 \times 10^{-2}$ were set to zero, and these entries were masked to prevent subsequent updates. This enforces the parsimony principle central to SINDy by progressively eliminating spurious terms. After each thresholding step, the learning rate was decreased by a factor of 10, and the optimizer was reinitialized to adapt to the smaller step size. Convergence was monitored by tracking the average change in $\boldsymbol{\alpha}$ over a sliding window of 5 epochs, with a tolerance of 10^{-6}. However, this convergence criterion was not met during the 600 epochs, highlighting the relatively slow progress of gradient-based optimization compared to closed-form sparse regression.

Figure 5.10 shows how the reconstruction error varies with execution time across the different methods, revealing three main insights. First, as expected, the closed-form implementations (dashed lines) are significantly faster than their gradient descent counterparts (solid lines). This efficiency gap arises because closed-form methods rely on direct sparse regression with thresholding, which converges in essentially one step, whereas gradient-based approaches require hundreds of iterations with repeated backpropagation, making them orders of magnitude slower. This difference justifies the emphasis placed throughout the chapter on closed-form, parameter-free approaches, as they achieve sparsity and accuracy with far less computational overhead.

Second, introducing unrolling into Euler-SINDy and RK4-SINDy in the closed-form setting does not result in noticeable additional costs. The unrolled scheme simply partitions the integration interval into smaller effective steps without increasing the algebraic complexity of the regression, so the computational burden remains essentially unchanged. At the same time, the reduced truncation error of unrolling improves robustness to large time steps, offering a clear numerical benefit without trade-offs in efficiency.

Third, when gradient descent is used, the additional burden of unrolling becomes more pronounced. Each unrolled step increases the depth of the computational graph, which not only lengthens backpropagation chains and raises memory usage, but also complicates the optimization landscape. Consequently, execution time grows more steeply compared to the non-unrolled methods, and convergence is slower, reflecting the limitations of gradient-based training when combined with unrolling.

These results highlight a fundamental trade-off between accuracy, robustness, and computational efficiency. Gradient-based methods, while flexible, come with significant computational costs. Their main advantage lies in their adaptability: they can incorporate sophisticated constraints such as adaptive regularization, stochastic optimization strategies, or non-standard loss functions that go beyond what closed-form regression can easily achieve. However, this flexibility comes at the price of much slower convergence and higher memory demands. The difficulty becomes even greater when unrolled schemes are employed, since each unrolled step increases both the computational depth and the complexity of the optimization process.

In contrast, closed-form approaches achieve an appealing balance. They are extremely efficient, often converging in a single regression step, and they scale favorably with system size. Importantly, they maintain their efficiency even when unrolling is applied, because unrolling in the closed-form setting only modifies the effective step size without altering the algebraic complexity of the regression. This combination of low cost and robustness makes unrolled closed-form methods particularly well suited for large-scale or real-time applications where both tractability and resilience to sparse or irregular data are essential.

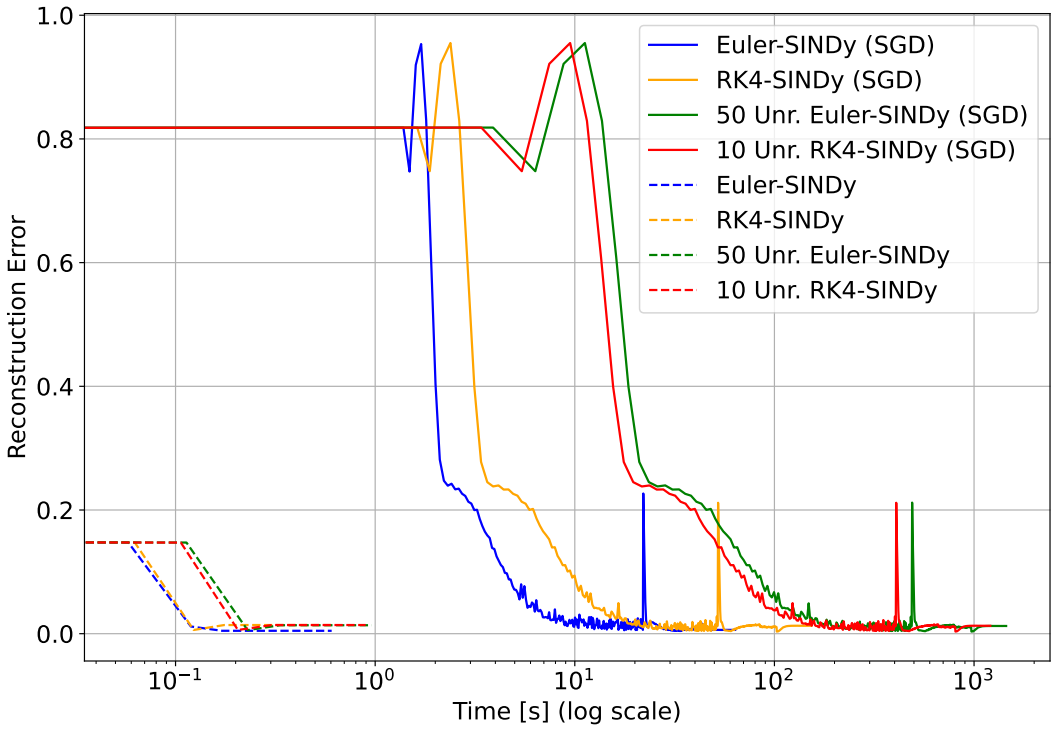


FIGURE 5.10: Comparison of the reconstruction errors as a function of the execution time (log scale) for gradient descent (SGD) and closed-form SINDy models on the 2-dimentional cubic damped oscillator (where $h = 10^{-3}$).

Taken together, these findings suggest a practical guideline: whenever possible, closed-form unrolled methods should be preferred as the default choice, since they provide both computational efficiency and robustness to coarse temporal sampling. Gradient descent variants may still be valuable, but their use is best reserved for situations where closed-form regression cannot be applied, for instance when the function library includes highly nonlinear, data-driven features or when additional modeling constraints must be enforced through the optimization process.

5.5.2 Advection equation

After validating the proposed methodology on the cubic oscillator ODE through multiple experiments, which demonstrated both its accuracy and robustness, we now turn to partial differential equations (PDEs). Unlike ODEs, PDEs describe the evolution of fields that depend on both space and time, and thus present additional challenges such as spatial discretization, numerical stability, and sensitivity to noise. To illustrate the applicability of our framework in this broader setting, we begin with one of the simplest yet most fundamental PDEs: the linear advection equation.

This partial differential equation (PDE) models the advection of a scalar field $u(x,t)$ under a constant velocity field. The governing equation is

$$u_t(x,t) = -0.4\, u_x(x,t), \tag{5.16}$$

where $u_t(x,t)$ and $u_x(x,t)$ are shorthand notations for the partial derivatives $\frac{\partial u}{\partial t}(x,t)$ and $\frac{\partial u}{\partial x}(x,t)$, respectively. More generally, higher-order derivatives with

respect to x will be written in the compact form

$$u_{xx} = \frac{\partial^2 u}{\partial x^2}, \quad u_{xxx} = \frac{\partial^3 u}{\partial x^3}, \quad u_{xxxx} = \frac{\partial^4 u}{\partial x^4},$$

and so forth. In this notation, the subscript indicates both the differentiation variable and its multiplicity. For brevity, we will consistently use this compact notation $(u_t, u_x, u_{xx}, \dots)$ instead of writing the full derivative expressions throughout the discussion.

Physically, this equation represents the classical linear advection process, which models the transport of a scalar quantity u (such as temperature, concentration, or density) through space at a constant velocity of 0.4. Although the model appears simple, it serves as a cornerstone in numerical analysis because it is highly sensitive to discretization errors, making it an effective test case for evaluating the stability, accuracy, and robustness of numerical integration schemes. In addition, the linear advection equation is frequently used in the field of data-driven discovery of dynamical systems, such as in SINDy [20] or PDE-FIND [109] frameworks, where its clear structure and sensitivity provide an excellent benchmark for testing the ability of these methods to recover governing equations from data.

Data generation. Using the `pde-bench` framework [125], we generate synthetic data comprising 20000 temporal snapshots over $t \in [0, 2]$ and a spatial domain $\Omega = [0, 1]$, discretized on a regular grid of size 20000×1024. The reference time step is set to $h = 5 \cdot 10^{-4}$, and the system is initialized with a sinusoidal profile

$$u(x, 0) = \sin(2\pi x),$$

which corresponds to a smooth wave transported by the velocity field without change in shape. Periodic boundary conditions are imposed at $x = 0$ and $x = 1$, ensuring continuity across the domain. This high-resolution dataset provides a reliable reference for testing the ability of SINDy and unrolled variants to accurately recover the underlying PDE dynamics.

Training protocol. The sparsity threshold is fixed at $\alpha_{th} = 0.01$, and the regularization parameter is set to $\lambda = 10^{-2}$. The dictionary of candidate terms considered is

$$\{1, u, u^2, u^3, u_x, u_{xx}, u_{xxx}\},$$

where spatial derivatives u_x, u_{xx}, and u_{xxx} are computed numerically using finite differences on the discrete grid, with centered differences in the interior and forward/backward differences at the boundaries. This dictionary contains both nonlinear terms in u and its derivatives. In the case of the linear advection equation, only the term u_x is active in the true governing equation, while all other terms should ideally be eliminated by the sparsity-promoting procedure.

Results analysis. Before discussing the results, we recall that our unrolled-SINDy method is applied to the linear advection equation on a high-resolution grid of size 20000×1024, corresponding to 20000 temporal snapshots over the interval $t \in [0, 2]$. The spatial grid is kept fixed while the temporal grid is varied to study the effect of coarser time sampling. Specifically, we consider time steps ranging from $h = 2 \cdot 10^{-4}$ (with $N = 10000$ observation pairs) up to $h = 4 \cdot 10^{-2}$ (with $N = 50$ observation

TABLE 5.6: Robustness of Euler-SINDy and its unrolled version on advection equation (Eq. 5.16), with an increasing time step h and a decreasing number of learning pairs.

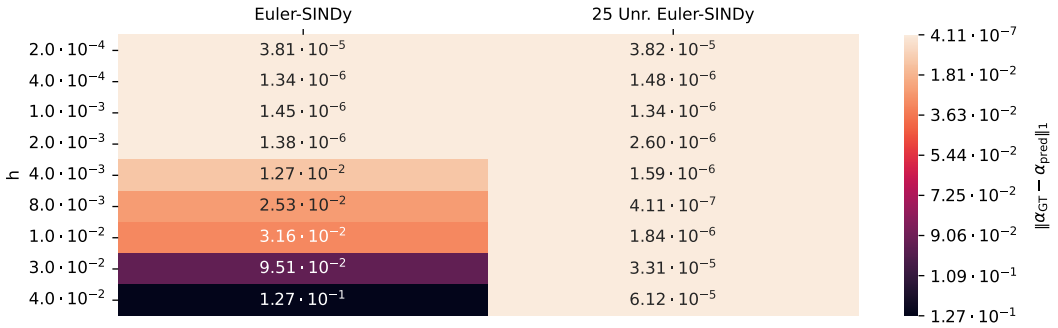
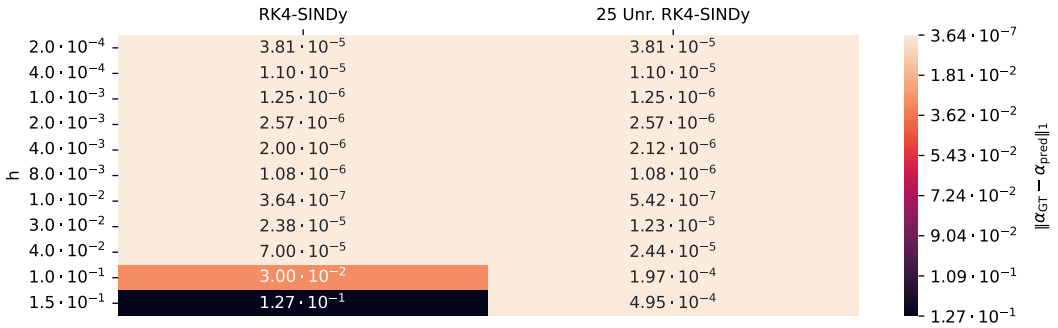


TABLE 5.7: Robustness of RK4-SINDy and its unrolled version on advection equation (Eq. 5.16), with an increasing time step h and a decreasing number of learning pairs.



pairs), enabling us to evaluate the ability of the methods to recover a governing equation under increasingly sparse temporal data.

From now on, we will use Table 5.6 to guide our analysis, while a full summary of the identified governing equations is available in Appendix B.2. Table 5.6 reports the ℓ_1 loss between the true coefficients and the predicted coefficients. When applying Euler-SINDy to this problem, we observe that the method becomes unstable as soon as the time step reaches $h = 4 \cdot 10^{-3}$. This instability stems from the fact that the explicit Euler scheme has a very restrictive stability region for PDEs like advection, where truncation and dispersion errors quickly dominate at larger step sizes. By contrast, its unrolled counterpart with $K = 25$ steps remains stable and accurate up to $h = 4 \cdot 10^{-2}$. The improvement arises from the unrolling mechanism, which effectively replaces one large step of size h by multiple smaller sub-steps of size h/K, thereby reducing the local truncation error while keeping the computational framework unchanged.

As illustrated in Table 5.7, RK4-SINDy performs better than Euler-SINDy and can reliably recover the governing PDE up to $h = 4 \cdot 10^{-2}$. However, beyond this point it too begins to degrade. In contrast, 25-Unrolled RK4-SINDy continues to deliver accurate results even when the step size is as large as $h = 0.15$. This striking difference highlights how unrolling leverages the intrinsic accuracy of higher-order schemes and extends their effective stability range far beyond the non-unrolled baseline.

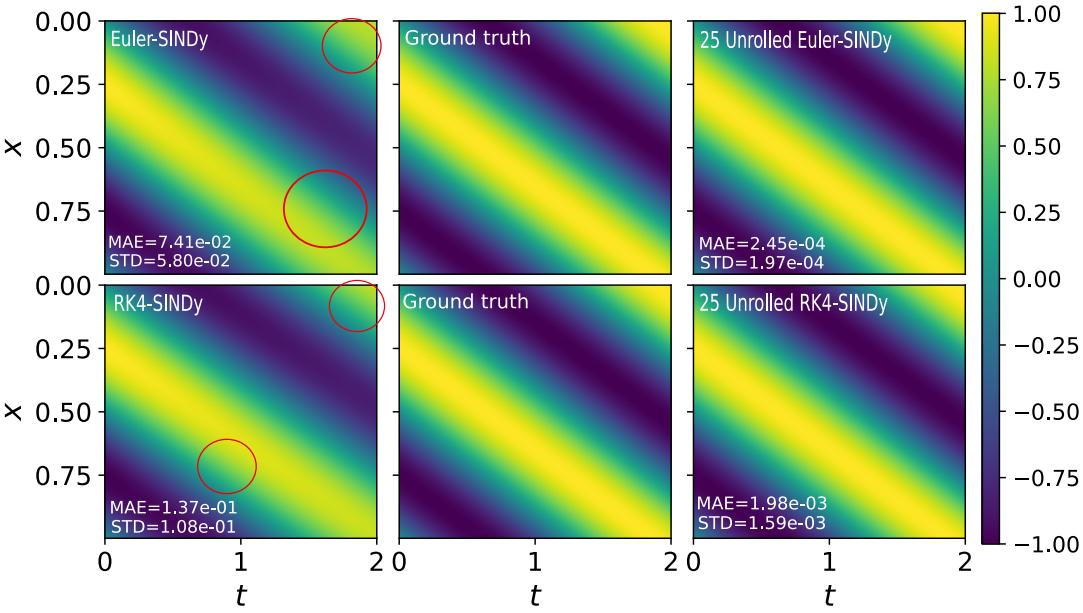


FIGURE 5.11: Solutions of the Advection PDE for different time steps. A time step of $h = 0.04$ is used for Euler-based methods and $h = 0.15$ for RK4-based methods. The first row (with $h = 0.04$) shows Euler-SINDy, the Ground Truth (GT), and 25 Unrolled Euler-SINDy. The second row (with $h = 0.15$) shows RK4-SINDy, GT, and 25 Unrolled RK4-SINDy. For each learned equation, the corresponding Mean Absolute Error (MAE) is indicated.

The equation solutions at the h values where Euler-SINDy and RK4-SINDy saturate, alongside their unrolled variants, are illustrated in Fig. 5.11. At first sight, the visual comparison of the reconstructed solutions might suggest that all methods perform similarly. However, closer inspection reveals significant qualitative differences: non-unrolled models produce visibly blurred structures in some regions (highlighted by red-circled areas in Fig. 5.11). These blurred regions are indicative of accumulated numerical dispersion and instability, typical when the time step is too large for the scheme to handle reliably.

In terms of Mean Absolute Error (MAE) shown in Fig. 5.11, the unrolled models consistently achieve errors that are two orders of magnitude lower than those of the original Euler-SINDy and RK4-SINDy algorithms.

5.5.3 Reaction diffusion equation

Reaction-diffusion partial differential equations (PDEs) are widely used to describe the spatiotemporal evolution of chemical or biological substances. These equations capture two fundamental processes: local reactions, which dictate how the concentrations of substances grow or decay due to nonlinear interactions, and diffusion, which governs how these substances spread across space. Reaction-diffusion systems are central in fields such as chemistry, physics, and biology, and they provide canonical models for phenomena like Turing patterns, chemical oscillations, and morphogenesis in biological tissues.

In our study, we consider a two-component reaction–diffusion system given by:

$$\begin{cases} u_t(t,x,y) &= 0.1\,\Delta u(t,x,y) + R_u\big(u(t,x,y), v(t,x,y)\big), \\ v_t(t,x,y) &= 0.1\,\Delta v(t,x,y) + R_v\big(u(t,x,y), v(t,x,y)\big), \end{cases} \tag{5.17}$$

where the Laplacian is defined as

$$\Delta u = u_{xx} + u_{yy}, \quad \Delta v = v_{xx} + v_{yy},$$

and the reaction terms are

$$R_u(u,v) = u - u^3 + v^3 + u^2 v - u v^2, \quad R_v(u,v) = v - u^3 - v^3 - u^2 v - u v^2.$$

$u(t,x,y)$ and $v(t,x,y)$ represent the concentrations of two interacting species. For compactness, we use the shorthand notation

$$u_{xx} = \frac{\partial^2 u}{\partial x^2}, \quad u_{yy} = \frac{\partial^2 u}{\partial y^2}, \quad v_{xx} = \frac{\partial^2 v}{\partial x^2}, \quad v_{yy} = \frac{\partial^2 v}{\partial y^2},$$

to denote second-order spatial derivatives along the x and y directions.

The nonlinear terms u^3, v^3, $u^2 v$, and $u v^2$ encode complex reaction interactions between the two fields, while the diffusion terms $u_{xx} + u_{yy}$ and $v_{xx} + v_{yy}$ describe the isotropic spreading of each substance in space.

Compared to the linear advection equation, the 2D reaction–diffusion system poses additional challenges for PDE discovery. First, the presence of second-order spatial derivatives increases the complexity of the terms that must be accurately estimated from data, requiring precise numerical differentiation. Second, the system evolves over two spatial dimensions, which enlarges the spatial domain and increases the number of data points needed to resolve the dynamics. Third, the reaction terms introduce nonlinear interactions between the two fields u and v, making it necessary for the discovery method to identify cross-coupled terms such as $u^2 v$ and $u v^2$. These features, namely higher-order derivatives, multiple spatial dimensions, and coupled nonlinear reactions, make the reaction–diffusion system a more stringent benchmark for evaluating the accuracy, sparsity, and robustness of data-driven PDE discovery methods such as SINDy.

Data generation For the reaction–diffusion dynamics, we generate synthetic data using a solver from `PDE-FIND` framework [109]. The dataset consists of 20000 temporal snapshots over $t \in [0, 10]$ on a two-dimensional spatial domain $\Omega = [-10, 10]^2$, discretized on a uniform grid. The integration is performed with a fine time step $h = 5 \cdot 10^{-4}$ to ensure numerical stability and accuracy. The system is initialized with Gaussian-shaped initial conditions,

$$u(x,y,0) = \exp\left(-\frac{x^2 + y^2}{2}\right), \quad v(x,y,0) = 0,$$

which provide a localized concentration profile that subsequently spreads and interacts through both reaction and diffusion mechanisms. Periodic boundary conditions are imposed along both spatial directions, ensuring continuity across the edges of the square domain.

Training protocol The sparsity threshold is fixed at $\alpha_{th} = 0.05$, and the regularization parameter is set to $\lambda = 10^{-1}$. The dictionary of candidate terms considered

is

$$\{1,\, u,\, v,\, u^2,\, v^2,\, u^3,\, v^3,\, u_x,\, v_x,\, u_y,\, v_y,\, u_{xx},\, v_{xx},\, u_{yy},\, v_{yy},\, u_{xy},\, v_{xy},\, uv,\, u^2v,\, uv^2\},$$

where spatial derivatives are approximated numerically using finite differences on the discrete grid, with centered schemes in the interior and forward/backward schemes at the boundaries. This dictionary captures both nonlinear reaction terms (e.g., u^3, uv^2) and diffusive contributions (e.g., u_{xx}, v_{yy}), as well as cross-derivative and cross-coupling terms that are essential for modeling the two-field reaction–diffusion system.

Results analysis As in the advection experiments, the spatial discretization is kept fixed while the resolution of the temporal grid is varied by adjusting the integration step size h, which also determines the number of training pairs available for model identification.

Table B.6 reports the recovered equation by Euler-SINDy across different time step sizes. These results, reported in the appendix tables, can also be followed visually through Table 5.8, which provides a complementary view of the ℓ_1 loss between the true and predicted coefficients. Euler-SINDy remains stable for small step sizes but begins to diverge once the step size reaches $h = 1.25 \cdot 10^{-1}$. By contrast, its unrolled variant with $K = 25$ significantly improves robustness, successfully recovering the governing equations even for step sizes as large as $h = 1$. A similar trend is observed for RK4-SINDy, whose results are summarized in Table B.7. While the standard RK4-SINDy formulation is able to correctly identify the dynamics up to $h = 0.625$, it fails when the step size is increased to $h = 1$. In this case, the unrolled version with $K = 25$ once again demonstrates superior stability and accuracy, maintaining perfect recovery of the equations. These findings highlight the key advantage of unrolled SINDy variants in handling larger integration steps, making them more practical for coarse-grained scenarios.

Figure 5.12 highlights the most demanding scenario, where the time step is increased to $h = 1$, yielding training data that are very sparsely distributed in time. In this setting, standard Euler-SINDy fails to reproduce the correct dynamics, with errors nearly an order of magnitude larger than the ground truth. By contrast, the 25-step unrolled Euler-SINDy succeeds in capturing the dominant structures and spatial patterns of the solution, reaching a mean absolute error (MAE) close to 10^{-2}.

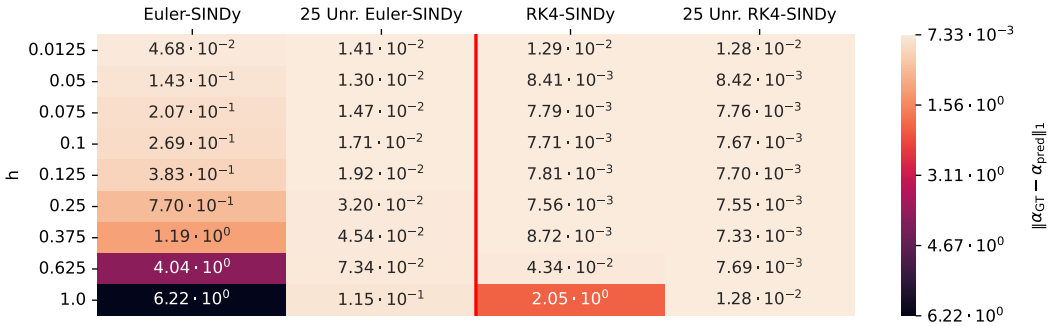
For RK4-based approaches, the advantage of unrolling is less striking in visual comparisons, since RK4 already delivers higher baseline accuracy than Euler. Nevertheless, quantitative analysis shows that unrolled RK4-SINDy reduces the error to about 10^{-3}, while the standard version remains near 10^{-2}. Although this improvement may not be obvious by inspection, it is statistically significant, as confirmed by a Student's t-test.

It is worth noting that unrolling comes with an additional computational cost: using $K = 25$ increases runtime by roughly a factor of 6 to 7, as detailed in Appendix B.3 (Table B.11). This extra cost is still justified, as it enables recovery of the underlying PDE in situations where traditional SINDy fails.

5.5.4 Kuramoto–Sivashinsky equation

The Kuramoto–Sivashinsky (KS) equation is a prototypical nonlinear partial differential equation (PDE) that exhibits rich spatiotemporal dynamics, ranging from pattern formation to deterministic chaos. It arises in a variety of physical contexts, including

TABLE 5.8: Robustness of Euler-SINDy and its unrolled version on reaction-diffusion equation (Eq. 5.17), with an increasing time step h and a decreasing number of learning pairs.



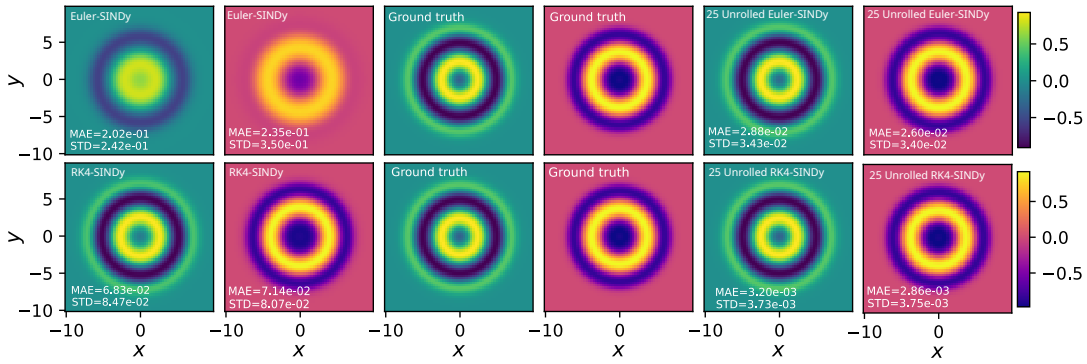


FIGURE 5.12: Solutions of the 2D Reaction-Diffusion PDEs with $h = 1$ and $K = 25$. The plots follow the same organization as in the 1D case, except that there are twice as many figures to represent the two variables: u in green and v in red.

instabilities of laminar flame fronts, thin liquid films, interfacial fluid dynamics, and certain reaction–diffusion systems where the interplay between diffusion, dispersion, and nonlinear advection is dominant. Its ability to generate both coherent structures and turbulent-like behavior makes it a canonical benchmark for testing data-driven PDE discovery methods.

In this work, we consider the one-dimensional KS equation in the form:

$$u_t(t,x) \quad = \quad -u_{xx}(t,x) - u_{xxxx}(t,x) - 5\,u(t,x)\,u_x(t,x), \tag{5.18}$$

where $u(t,x)$ is the evolving field defined over space and time.

The notation is as follows:

$$u_x = \frac{\partial u}{\partial x}, \quad u_{xx} = \frac{\partial^2 u}{\partial x^2}, \quad u_{xxxx} = \frac{\partial^4 u}{\partial x^4}.$$

Each term in this PDE carries a distinct physical role. The destabilizing term $-u_{xx}$ drives the growth of perturbations, while the stabilizing higher-order diffusion $-u_{xxxx}$ suppresses short-wavelength instabilities. The nonlinear advective interaction $-5uu_x$ couples modes across scales, enabling energy transfer and leading to the emergence of complex spatiotemporal patterns. The competition between these linear and nonlinear effects is what produces the chaotic yet structured dynamics that

characterize the KS system.

As with reaction–diffusion systems, the KS equation involves both linear operators and nonlinear couplings, but here the higher-order derivative u_{xxxx} makes numerical approximation more challenging, while the advective nonlinearity introduces strong mode interactions that complicate sparse regression and PDE identification.

Data generation For the Kuramoto–Sivashinsky system, we generate synthetic data by numerically solving the PDE using an Exponential Time Differencing (ETD) scheme implemented in JAX. The implementation of the solver was adapted from the GitHub repository [24]. The simulations are carried out on a one-dimensional spatial domain $\Omega = [0, L]$ with $L = 64$, discretized into 100 uniformly spaced grid points. The temporal domain is set to $t \in [0, 200]$, integrated with a fine time step $\Delta t = 0.001$ to ensure stability and accurate resolution of the rapid, multiscale dynamics inherent to the KS equation.

The system is initialized with a Gaussian-shaped profile,

$$u(x, 0) = 0.5 \exp\big(-100(x - L/2)^2\big),$$

which represents a localized perturbation centered at $x = L/2$. This localized bump evolves into a complex spatiotemporal pattern as instabilities amplify, nonlinear advection redistributes energy, and higher-order diffusion counteracts small-scale fluctuations. Periodic boundary conditions are imposed at the edges of the spatial domain, enforcing continuity of the solution and preventing boundary artifacts. This dataset provides an interesting benchmark for evaluating the accuracy and robustness of PDE discovery methods when applied to chaotic spatiotemporal dynamics.

Training protocol The sparsity threshold is set to $\alpha_{th} = 0.1$, and the regularization parameter to $\lambda = 10^{-6}$. The dictionary of candidate terms considered is

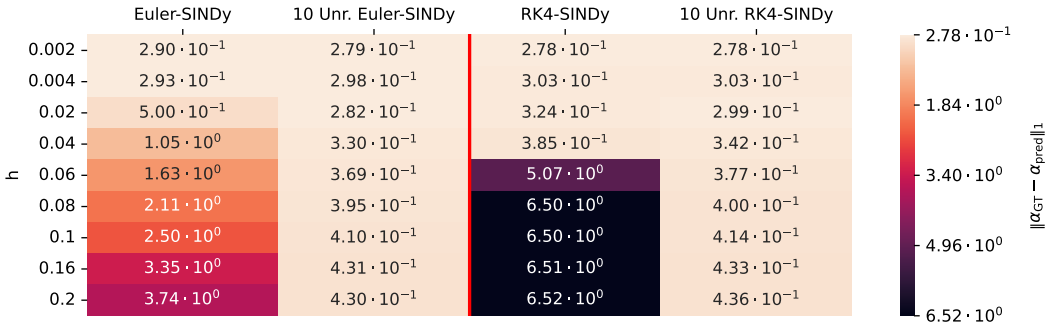
$$\{1, u_x, u_{xx}, u_{xxx}, u_{xxxx}, uu_x\},$$

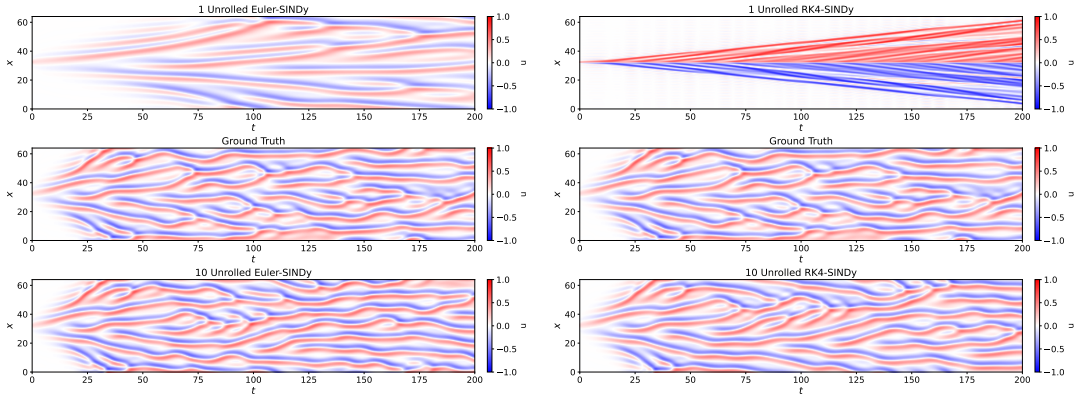
where spatial derivatives up to fourth order are approximated using finite differences on the discrete grid, with centered schemes in the interior and one-sided schemes at the boundaries. This dictionary is specifically tailored to the Kuramoto–Sivashinsky system: it includes linear instability contributions (u_{xx}), higher-order dissipation (u_{xxxx}), and nonlinear advective effects (uu_x).

Results analysis In all experiments, the spatial discretization is kept fixed to ensure that spatial derivatives are consistently resolved across trials. The variability instead comes from the temporal grid: we progressively increase the time step h while proportionally reducing the number of observation pairs available for training.

The results illustrate the limitations of Euler-SINDy when observations are sparse. As shown in Table 5.9, Euler-SINDy ability to recover the true dynamics of KS equation becomes unstable once the effective time step reaches $h = 0.02$, at which point it fails to identify the correct governing terms. In contrast, the unrolled Euler-SINDy, which replaces each large step by $K = 10$ smaller intermediate updates, remains accurate and stable up to $h = 0.12$. This improvement reflects the ability of unrolling to recoverer the underlying dynamics of a phenomenon even when observations are temporally sparse. While RK4-SINDy becomes unstable slower than Euler-SINDy, the overall behavior is similar. These trends can also be followed

TABLE 5.9: Robustness of Euler-SINDy and its unrolled version on Kuramoto–Sivashinsky equation Eq. 5.18, with an increasing time step h and a decreasing number of learning pairs.





(a) (b)

FIGURE 5.13: Solutions of the Kuramoto–Sivashinsky PDE with $h = 0.2$. a) From top to bottom: Euler-SINDy, ground truth, and 10 Unrolled Euler-SINDy. b) From top to bottom: RK4-SINDy, ground truth, and 10 Unrolled RK4-SINDy.

visually in Table B.9, which provides a complementary view of the ℓ_1 loss between true and predicted coefficients.

Figures 5.13.a and 5.13.b highlights the qualitative gap between the methods. Euler-SINDy and RK4-SINDy fail to reproduce the essential dynamics, with errors that accumulate rapidly and lead to noticeable departures from the ground truth. In contrast, the unrolled variants track the dominant trends and spatial structures much more accurately, yielding solutions that remain close to the true KS dynamics. Some discrepancies persist, however, reflecting the inherent difficulty of recovering a fourth-order nonlinear PDE from temporally sparse or noisy data.

5.5.5 Unrolled iNeural-SINDy

iNeural-SINDy

The iNeural-SINDy [48] framework is a robust extension of the SINDy method, designed to handle noisy and scarce data. It combines neural networks, sparse regression and an integral formulation to stabilize the discovery of governing equations. Figure 5.14 illustrates the methodology on the example of a cubic oscillator.

To explain and illustrate the iNeural-SINDy framework, let us consider the example of a two-dimensional cubic damped oscillator(see Fig. 5.14.a). In this case, the available data are corrupted by measurement noise, so that instead of directly observing the true trajectory but we only have access to noisy measurements. The objective of iNeural-SINDy is to recover both the clean dynamics and the governing equations from these noisy data.

To achieve this, iNeural SINDy introduces a deep neural network (DNN) that serves as an implicit representation of the trajectory as illustrated in Fig. 5.14.b. The DNN is trained to approximate the clean state from noisy observations. By doing so, it acts simultaneously as a denoiser and interpolator, producing smooth trajectories that remain faithful to the data. A crucial advantage of this representation is that derivatives can be computed reliably through automatic differentiation, which avoids the numerical instabilities associated with finite differences in noisy settings.

Similarly to SINDy, iNeural SINDy constructs a candidate function library as presented in Fig. 5.14.c. In this case, this library typically contains polynomial, interaction, and derivative terms that may appear in the governing equations Sparse regression is then applied to select a parsimonious set of terms by estimating the coefficient matrix. Sequential thresholding is used during training to prune away coefficients below a chosen threshold, ensuring that the resulting model remains both interpretable and sparse.

Similar to SINDy, iNeural SINDy builds a library of candidate functions, shown in Fig. 5.14.c. This library generally includes polynomial, interaction, and derivative terms that could appear in the underlying governing equations. Sparse regression is then employed to identify a minimal set of relevant terms by estimating the corresponding coefficient matrix. During training, sequential thresholding removes coefficients below a specified threshold, ensuring that the resulting model remains both interpretable and sparse.

The key innovation of iNeural-SINDy lies in its loss function, shown in Fig. 5.14.d. The total loss combines three complementary components: a data fidelity term that keeps the neural network outputs close to the noisy observations, a derivative matching term that ensures consistency between the neural network derivatives and the sparse model predictions, and an integral term, based on a Runge–Kutta scheme, that aligns the trajectories generated by the sparse model with the DNN outputs. In the original formulation, Runge–Kutta scheme is used but other integration methods could also be applied and unrolled. This combination stabilizes training by enforcing both local accuracy through derivative matching and global temporal consistency via the integral constraint.

Finally, Fig. 5.14.e highlights the effectiveness of the framework. The estimated coefficient matrix successfully captures the dominant terms of the cubic oscillator while eliminating spurious contributions. The reconstructed trajectories (purple diamonds) closely track the ground truth dynamics (green curve), despite the model being trained on noisy observations (black points).

In essence, iNeural-SINDy combines neural networks for smooth trajectory reconstruction, automatic differentiation for stable derivative estimation, an RK4-based integral constraint for dynamical consistency, and sparse regression for interpretability. By integrating these components, the method extends the classical SINDy framework into a powerful tool for data-driven discovery of nonlinear dynamical systems under realistic conditions of noise, sparsity, and limited temporal resolution.

iNeural-SINDy is robust to noise but struggles to recover equations when observations are widely spaced in time. Conversely, unrolled *-SINDy methods are highly efficient in scenarios with large temporal gaps but are sensitive to noise. To

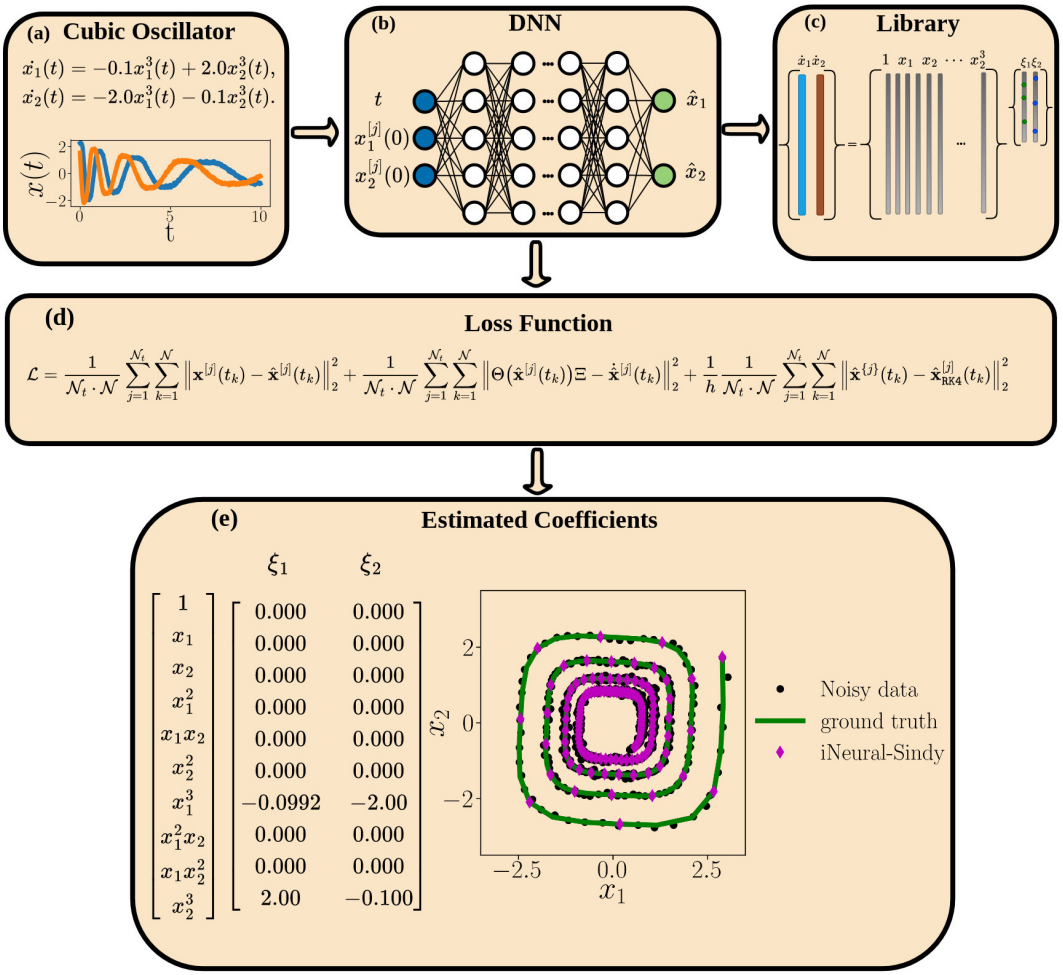


FIGURE 5.14: Schematic illustration of the iNeural-SINDy algorithm applied to the cubic damped oscillator. Adapted from the figure available on GitHub [47].

address both challenges, we propose Unrolled iNeural-SINDy, a hybrid approach that combines the noise robustness of iNeural-SINDy with the temporal efficiency of unrolled methods in order to identify governing equation even with sparse and noisy observations.

Unrolled iNeural SINDy

Unrolled iNeural-SINDy extends the principles of Unrolled-SINDy to neural-network-based PDE/ODE discovery frameworks. Analogous to Unrolled *-SINDy-SGD, it replaces a single large integration step with multiple smaller unrolled steps, thereby reducing local truncation errors while maintaining compatibility with noise-robust training procedures. This approach allows the model to leverage the inherent robustness of iNeural-SINDy to measurement noise while further improving the accuracy of coefficient recovery in scenarios with sparse or temporally coarse data. The following paragraphs illustrate this methodology on two representative systems: the cubic damped oscillator and the Fitz-Hugh Nagumo system.

Cubic oscillator equation We recall here the formula of the cubic oscillator, already presented in Section 5.5.1.

$$\begin{cases} \dot{x}(t) = -0.1\,x^3(t) + 2.0\,y^3(t), \\ \dot{y}(t) = -2.0\,x^3(t) - 0.1\,y^3(t). \end{cases} \tag{5.19}$$

Unlike the previous setting, the data generation and training protocol are adapted to follow the methodology introduced in the iNeural-SINDy paper [48].

Data generation. The dataset is generated synthetically using the `solve_ivp` function. We consider different noise levels σ ranging from 0 to 0.06 and time steps h ranging from 0.025 to 0.333. The training data consist of 800 samples collected over the interval $t \in [0, 10]$.

Training protocol. The training is performed with a maximum of 30000 epochs and a tolerance threshold of $\alpha_{th} = 0.05$. An initial training phase of 15000 iterations is carried out with a learning rate of 10^{-4} for the DNN parameters and 10^{-3} for the coefficient matrix α_{pred}. After this initialization, sequential thresholding is applied every 5000 iterations. Following each thresholding step, both the DNN parameters and the coefficient matrix α_{pred} are updated. The dynamical system is reconstructed within the polynomial feature space spanned by all monomials up to degree three:

$$\{1,\, x,\, y,\, x^2,\, xy,\, y^2,\, x^3,\, x^2y,\, xy^2,\, y^3\}.$$

Result analysis. For Euler-iNeural-SINDy, in the absence of noise (Table 5.10.A) as the time step h increases, the ℓ_1 loss between the true and predicted coefficients also increases. This confirms that Euler-iNeural-SINDy struggles to accurately recover the studied equation when observations are widely spaced since larger time steps amplify discretization errors.

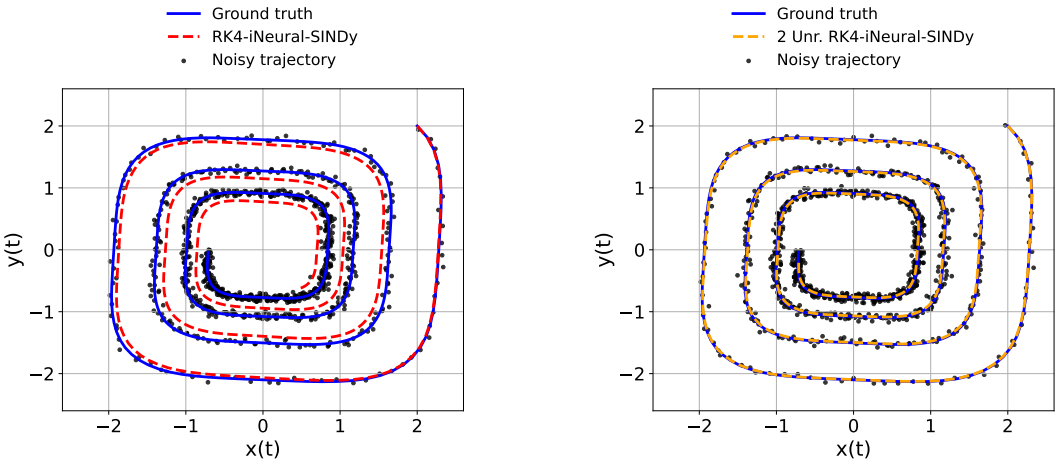
Then when the noise is introduced the increase in ℓ_1 loss remains moderate and the predictions stay relatively close to the no noise case. This indicates a certain level of robustness to noise, even though the method alone does not fully handle large temporal gaps.

A significant improvement is observed with the unrolled variant, 8 Euler-iNeural-SINDy (Table 5.10.B). By unrolling the dynamics across multiple steps, the model captures more of the underlying structure of the system, resulting in a global reduction in ℓ_1 loss. This demonstrates that unrolling effectively mitigates sensitivity to both larger time steps and higher noise levels.

A similar trend is found for RK4-iNeural-SINDy and its unrolled version, 2 RK4-iNeural-SINDy (Table 5.10.C–D). Both follow the same qualitative behavior as their Euler counterparts but achieve systematically lower ℓ_1 loss. This advantage stems from the higher accuracy of the RK4 scheme, which provides a more faithful numerical approximation of the continuous dynamics.

Fig. 5.15 illustrates the solutions reconstructed by RK4 iNeural-SINDy and its unrolled variant on the cubic damped oscillator with noisy data ($\sigma = 0.02$). While iNeural-SINDy is already designed to be robust to noise, the standard RK4 implementation still gradually deviates from the true dynamics as the simulation progresses. In contrast, the 2-Unrolled RK4-iNeural-SINDy further enhances this robustness, effectively mitigating noise and closely tracking the true solution throughout the interval.

In summary, unrolling consistently enhances the identification capability of iNeural-SINDy. While Euler-iNeural-SINDy benefits most clearly from unrolling, RK4-iNeural-SINDy already achieves higher accuracy due to its integration method, and unrolling further strengthens its robustness. Overall, unrolled iNeural-SINDy yields models that are simultaneously resilient to noise and capable of handling



(a) (b)

FIGURE 5.15: Solution of cubic damped oscillator with noisy data ($\sigma = 0.02$) using a) RK4 iNeural-SINDy and b) 2 Unrolled RK4-iNeural-SINDy

sparse temporal observations, making it a powerful tool for learning governing equations from challenging datasets.

Linear oscillator equation We now consider the linear oscillator, a simple two-dimensional dynamical system commonly used to model harmonic motion. Unlike the cubic damped oscillator, which exhibits nonlinear behavior due to the presence of a cubic term in the restoring force, the linear oscillator evolves according to purely linear dynamics. This linearity ensures that the system exhibits sinusoidal oscillations whose amplitude and frequency are determined solely by the system parameters.

The governing equations are given by:

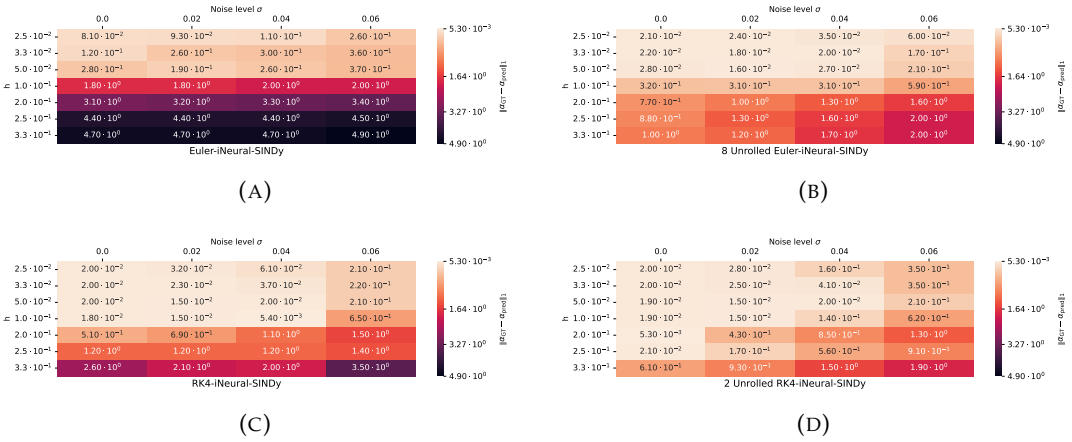
$$\begin{cases} \dot{x}(t) = -0.1\, x(t) + 2.0\, y(t), \\ \dot{y}(t) = -2.0\, x(t) - 0.1\, y(t). \end{cases} \tag{5.20}$$

Here, $\dot{x}(t)$ and $\dot{y}(t)$ denote the time derivatives of the state variables $x(t)$ and $y(t)$, respectively. The variable $x(t)$ represents the position of the oscillator, while $y(t)$ corresponds to its velocity. The off-diagonal coefficients 2.0 and -2.0 define the coupling between position and velocity, setting the natural oscillation frequency, whereas the diagonal terms -0.1 introduce weak damping, gradually reducing the amplitude of the oscillations.

Data generation. The dataset is generated synthetically using the `solve_ivp` function, with the parameter $\sigma \in [0, 0.06]$ controlling the relative noise amplitude. Experiments are conducted over a range of observation intervals $h \in [0.025, 0.333]$. The training data comprise 400 uniformly spaced samples collected over the time interval $t \in [0, 10]$.

Training protocol. The model is trained for up to 15000 epochs with a tolerance threshold of $\alpha_{\text{th}} = 0.05$. An initial training phase of 15000 iterations is performed using a learning rate of 10^{-4} for the DNN parameters and 10^{-3} for the coefficient matrix α_{pred}. After this pretraining stage, sequential thresholding is applied every 2000 iterations. Following each thresholding step, both the DNN parameters and the coefficient matrix α_{pred} are jointly updated. The dynamical system is reconstructed in

TABLE 5.10: Robustness of iNeural-SINDy on the cubic damped oscillator (Eq. 5.15), evaluated with increasing time step h and increasing noise level σ with A) Euler-iNeural-SINDy, B) 8 Unrolled Euler-iNeural-SINDy, C) RK4-iNeural-SINDy, and D) 2 Unrolled RK4-iNeural-SINDy.



(A) (B) (C) (D)

a polynomial feature space spanned by all monomials up to degree two:

$$\{1,\, x,\, y,\, x^2,\, xy,\, y^2\}.$$

Results analysis. We generate a series of subproblems from the original dataset by systematically varying the time step $h \in [0.025,\, 0.333]$ and the noise level $\sigma \in [0,\, 0.06]$.

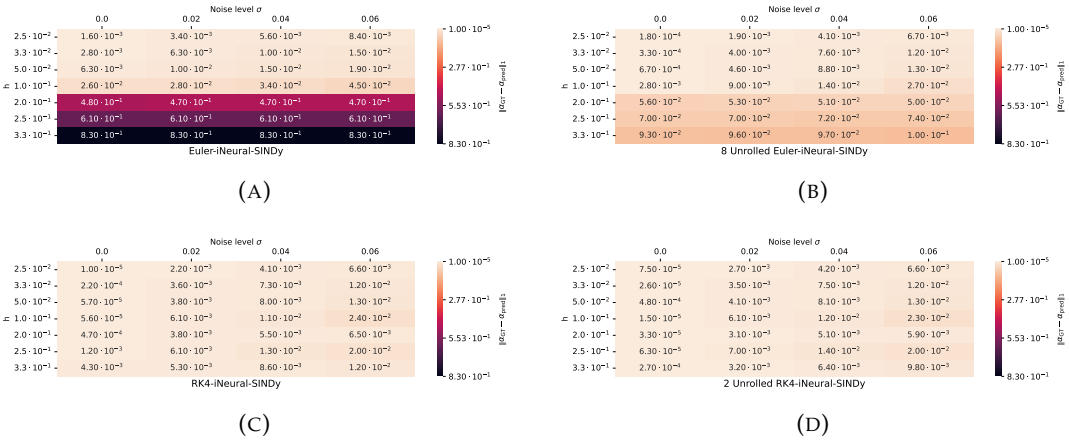
For the linear oscillator, the results exhibit trends similar with those observed for the cubic damped oscillator (Table 5.11.A–D). Increasing the time step h results in higher ℓ_1 losses, while adding noise has a moderate effect on the ℓ_1. But, unrolling allow to reduce the ℓ_1 despite the noise and large interval between observations. The 8-Unrolled Euler-iNeural-SINDy (Table 5.11.B) shows a clear reduction in error compared to the standard Euler-iNeural-SINDy (Table 5.11.A). A similar trend is observed for RK4-iNeural-SINDy (Table 5.11.C–D), which generally outperforms the Euler-based variant due to the higher accuracy of the RK4 integration scheme.

Fitz-Hugh Nagumo equation The system under consideration is the Fitz-Hugh Nagumo model, a two-dimensional nonlinear dynamical system originally introduced as a simplified version of the Hodgkin–Huxley equations for modeling the activation and deactivation dynamics of a spiking neuron. The governing equations are given by:

$$\begin{cases} \dot{x}(t) = 1.0\, x(t) - 1.0\, y(t) - \frac{1}{3}\, x^3(t) + 0.1, \\ \dot{y}(t) = 0.1\, x(t) - 0.1\, y(t). \end{cases} \tag{5.21}$$

Here, $\dot{x}(t)$ and $\dot{y}(t)$ denote the time derivatives of the state variables $x(t)$ and $y(t)$, respectively. The variable $x(t)$ represents the membrane potential of the neuron, while $y(t)$ acts as a recovery variable accounting for slower processes such as ion channel dynamics. The cubic nonlinearity $-\frac{1}{3}x^3(t)$ introduces a threshold-like behavior in the voltage dynamics, ensuring that small perturbations decay while sufficiently large ones trigger large excursions, mimicking neural spikes. The linear terms, such as $1.0\, x(t)$ and $-1.0\, y(t)$, determine the coupling between excitation and recovery, whereas the constant input 0.1 represents a baseline stimulus.

TABLE 5.11: Robustness of iNeural-SINDy on the linear oscillator (Eq. 5.20), evaluated with increasing time step h and noise level σ with A) Euler-iNeural-SINDy, B) 8 Unrolled Euler-iNeural-SINDy, C) RK4-iNeural-SINDy, and D) 2 Unrolled RK4-iNeural-SINDy.



(A) (B)

(C) (D)

Unlike purely oscillatory systems, the Fitz-Hugh Nagumo equations exhibit both rest states and spiking behavior depending on initial conditions and input strength. The interplay of cubic nonlinearity, linear coupling, and slow recovery dynamics makes the system a cornerstone in the study of nonlinear oscillations, bifurcations, and pattern formation.

Data generation. The dataset is generated synthetically using the `solve_ivp` function. We collect 400 samples over the time interval $t \in [0, 200]$. To assess robustness, measurement noise is added with levels σ ranging from 0 to 0.06, and the integration step size h is varied from 0.44 to 1.33.

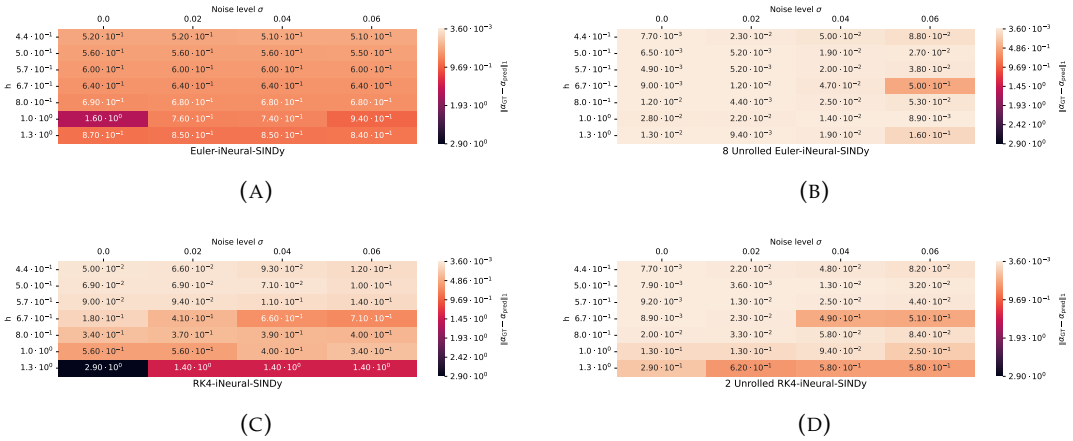
Training protocol. The employed DNN architecture consists of three hidden layers, each with 32 neurons. The maximum number of training epochs is set to 50000, with a tolerance threshold of $\alpha_{th} = 0.05$. The initial training phase is performed over 15000 iterations using a learning rate of 10^{-4} for the DNN parameters and 10^{-3} for the coefficient matrix α_{pred}. After this stage, sequential thresholding is applied every 5000 iterations, followed by an update of both the DNN parameters and α_{pred}. The governing equations are identified within the polynomial function space spanned by all monomials up to degree three:

$$\{1,\ x,\ y,\ x^2,\ xy,\ y^2,\ x^3,\ x^2y,\ xy^2,\ y^3\}.$$

Result analysis. We construct a series of sub-problems from the original dataset by systematically varying the time step h between 0.4 and 1.3 and the noise level σ from 0 to 0.06.

For the Fitz-Hugh Nagumo system, we observe the same general behavior as in the cubic damped oscillator case (Table 5.12.A–D). Increasing the time step h leads to larger ℓ_1 losses, while the addition of noise degrades ℓ_1 losses moderately. As before, unrolling improves robustness, with 8 Euler-iNeural-SINDy (Table 5.12.B) showing a clear reduction in error compared to the standard Euler-iNeural-SINDy (Table 5.12.A). A similar trend is seen for RK4-iNeural-SINDy (Table 5.12.C–D), which overall performs better than Euler due to the higher accuracy of the RK4 scheme.

TABLE 5.12: Robustness of iNeural-SINDy on the Fitz-Hugh Nagumo system (Eq. 5.21), evaluated with increasing time step h and increasing noise level σ. a) Euler-iNeural-SINDy, b) 8 Unrolled Euler-iNeural-SINDy, c) RK4-iNeural-SINDy, and d) 2 Unrolled RK4-iNeural-SINDy.



(A) (B)

(C) (D)

5.6 Limitations

Unrolled-SINDy primarily addresses the challenge of recovering accurate coefficients in sparse identification when the temporal resolution between observations is coarse. However, it does not inherently improve robustness to measurement noise. In noisy settings its ability to recover the true governing equation is similar to standard Euler-SINDy, as confirmed in the section 5.5.1. Since unrolling is not inherently robust to noisy measurements, the current solution is to integrate it with approaches that are already designed to handle noisy data. We demonstrated this integration with iNeural-SINDy in Section 5.5.5, showing that combining unrolling with a noise-robust framework can simultaneously reduce integration errors and improve recovery of sparse PDE representations.

Currently, Unrolled-SINDy focuses on learning linear combinations of predefined dictionary terms, which is sufficient for many standard PDEs but limits its expressivity. One could elaborate that extending the framework to handle nested or nonlinear combinations of dictionary terms, as in Nested SINDy [46], would allow capturing more complex interactions or hierarchically structured PDEs. Conversely, rational PDEs present a challenge because representing all possible numerator-denominator combinations in the dictionary Θ leads to combinatorial explosion. This limitation could be mitigated by integrating the unrolled scheme with methods like SINDy-PI [58], specifically designed for rational PDE discovery.

Another assumption in the current formulation is the time-invariance of coefficients $\boldsymbol{\alpha}$. Many physical systems exhibit parameters that vary with time or space, so future work could explore unrolled identification for varying-coefficient PDEs, requiring both adaptive dictionary evaluations and possibly time-dependent regression techniques.

Finally, a potential issue with the current closed-form unrolling approach is the existence of numerical instabilities, which can lead to NaN values in the predicted coefficient vector α_{pred}. These instabilities typically arise when intermediate predictions during unrolling diverge, often in challenging scenarios where the time step h is relatively large and the unrolling depth K is insufficient, making the matrix inversion ill-conditioned. However, this is not a fundamental limitation of the method.

Increasing K appropriately restores stability, eliminates NaN values and ensures accurate recovery of the system's coefficients. While problematic cases can also be handled by using the stochastic gradient descent (SGD), further improvements could involve incorporating regularization or adaptive step-size strategies during unrolling to reduce divergence risk.

5.7 Conclusion

In this work, we introduced Unrolled-SINDy, a principled extension of the Sparse Identification of Nonlinear Dynamics framework that leverages an unrolling mechanism to decouple the numerical integration time step from the observational sampling rate. This innovation allows explicit methods to operate effectively under coarse temporal resolutions, reducing integration errors without requiring additional data. The approach retains the advantages of closed-form regression, making it simple to implement, computationally efficient, and directly compatible with existing SINDy-like frameworks based on finite-difference schemes. It is important to clarify that the key contribution of this work is the "unrolling" approach itself not the unrolling a specific integration scheme like Euler or RK4. In fact, this approach could be applied any numerical scheme in order to allow the identification of a governing equation in situation where the observations are largely spaced in time.

We provided theoretical guarantees by showing that unrolling systematically reduces the local truncation error, and we highlighted the link between numerical stability and successful recovery of governing equations. This connection explains why unrolled variants of Euler-SINDy and RK4-SINDy achieve more reliable identification in regimes where standard explicit methods often fail. Our comprehensive experimental study across a wide variety of ODEs and PDEs validated these insights, demonstrating that unrolling substantially extends the operational regime of explicit methods while maintaining scalability and interpretability.

Beyond classical finite-difference formulations, we showed that unrolling is versatile and can be integrated with state-of-the-art neural network-based discovery approaches. In particular, we combined it with iNeural-SINDy [48], a method specifically designed to handle noise and scarce data. This hybridization improved overall ability of the method to recover the true governing equation when observations are simultaneously widely spaced in time and noisy. It demonstrates that unrolling is not limited to traditional numerical schemes but can also enhance the robustness and effectiveness of neural architectures for PDE discovery.

At the same time, several limitations of the present formulation open up promising avenues for future work. While unrolling mitigates integration error, it does not inherently improve robustness to measurement noise, motivating its integration with denoising techniques or neural-based methods, as illustrated here with iNeural-SINDy [48]. The reliance on linear combinations of predefined dictionary terms also constrains expressivity, suggesting that future extensions could incorporate nested or rational structures to capture richer dynamics. Furthermore, relaxing the assumption of time-invariant coefficients would enable applications to systems with spatially or temporally varying parameters. Finally, addressing occasional numerical instabilities in the closed-form solution could further strengthen the reliability of the approach in practice.

Chapter 6

Conclusion and perspectives

This thesis investigates the use of Physics-Informed Machine Learning to model the dynamics of self-organization in laser–matter interactions. Experimentally, the self-organization process is reproduced by irradiating metallic surfaces with an increasing number of laser double pulses. The initial objective was to design an end-to-end model that replicates this experimental process to predict the resulting surface morphology based on laser configuration and metallic surface initial state. From this objective emerged three main contributions.

The first is a fast, differentiable neural solver for laser energy absorption capturing the material response to laser irradiation by simulating accurately Maxwell's equations. This contribution was motivated by the need to model the interaction of the second pulse with a surface reshaped by the first pulse for which no microscopy scan is available.

The second contribution named Unrolled-SINDy is a PDE discovery method that handles cases where the measurements of the studied dynamics are distant in time. Standard approaches fail in such cases. This contribution addressed the lack of a governing equation describing the post-irradiation dynamics leading to self-organized patterns and the large temporal spacing between the scan of irradiated spots since the current devices cannot capture the surface evolution in real time.

The third contribution, while not part of the end-to-end module, leverages the fast, differentiable, and generalizable surrogate model for absorption developed in the first contribution to demonstrate that self-organized surface patterns resulting from laser irradiation act as a form of structural memory leading to the emergence of configurations that maximize energy absorption.

Although the modules of the end-to-end model were implemented, they were not combined into a complete predictive pipeline due to a remaining challenge: the scans of self-organized structures are not matched. It means even if two images appear consecutive within a series (as illustrated in Fig. 4.4), they do not belong to the same trajectory. Here, a trajectory is the temporal evolution of a single initial spot under successive laser pulses. The reason is that at the nanoscale, once a spot has been irradiated and scanned to obtain an image, we cannot reliably find and irradiate the same area again. This issue prevents repeated irradiations and measurements on the exact same spot. Furthermore, even on polished surfaces, slight variations in the initial conditions are unavoidable, meaning that every irradiated spot is inherently different. Consequently for a given series showing the evolution of self-organized patterns, the N scanned patterns are in reality from N distinct trajectories. Although PDE discovery method such as Unrolled-SINDy handle multiple trajectories, they still require at least an observation pair per trajectory, a requirement our experimental dataset cannot satisfy.

To address this limitation, two complementary directions could be explored. On one hand, a physical approach focusing on improving experimental setups to enable

matched observations. On the other hand, a machine learning approach aiming to infer the underlying dynamics directly from unmatched observations.

Concerning the physical approach, there are devices that can simultaneously irradiate and image a sample without moving it. However, they are expensive, making them not worth the investment for a single use. A more accessible approach consists in designing an experimental setup capable of producing matched observations of the same spot after successive irradiations. To get at least one pair of observations for each trajectory, we could imagine a different experimental setup. Instead of creating separate circular spots, the idea would be to perform N irradiations along a continuous line, which helps avoid interference between neighboring regions. After scanning these lines to obtain the first set of images, the experiment could then be repeated in a direction perpendicular to the initial one. Because the lines are continuous, a long enough line would ensure that some of the previously irradiated areas are hit again. By repeating this process several times, we would eventually obtain at least one pair of observations for each trajectory, making it possible to apply PDE discovery methods such as Unrolled-SINDy to find the governing equations.

About the machine learning approach, a possible solution could be to reformulate the PDE discovery method in a way that matched observation pairs are not explicitly required. Both the standard SINDy and Unrolled-SINDy optimization problems assume that consecutive observations pairs $(u(t_j, \mathbf{x}), u(t_{j+1}, \mathbf{x}))$ are from the same trajectory and this relation is true on the entire spatial domain $\mathbf{x} \in \Omega$ in order to approximate temporal derivative and identify a governing equation. Since the equation we try to identify is locally true it is not necessary to consider the whole spatial domain Ω. We could consider spatial subsets or patches of the observed patterns and search for those whose, if matched as pairs consistently explain the equation. The optimization problem will then consist in simultaneously determine the governing equation that best fits the data and find the pairs of spatial subsets that behaves like they are aligned in time and follows the same dynamics. However, the existence of such matching pairs is not always guaranteed. If they do exist, this would remove the main limitation that currently prevents the fully functional end-to-end model from predicting the formation of self-organized structures.

Overall, while certain constraints currently limit the implementation of the complete end-to-end model, the contributions of this work provide the foundation for a future realization. Together, these contributions still advance our understanding of self-organization in laser–matter interactions.

Appendix A

Learning to Learn in Laser–Matter Interactions: Linking Self-Organization to Learning Dynamics

A.1 Taylor complexity

How can we measure the diversity of shapes at a given point x? Take, for example, the shapes that we obtain from two real functions, $f(x) = a_1 x^2$ and $g(x) = sin(a_2 x) - sin(\sqrt{2} a_2 x)$, where $a_1, a_2 \in \mathbb{R}$ are sampled from the same discrete random variable A. Intuitively, the set of shapes that we obtain in the first case is a lot less diverse than the second one, even thought their source of diversity, A, is the same.

A variance-type measure of diversity of shapes could be obtained by adding the variances of the shape components, since for independent random variables X and Y:

$$\mathrm{Var}(X + Y) = \mathrm{Var}(X) + \mathrm{Var}(Y). \tag{A.1}$$

If x is small, in our f and g example above, the greatest contribution to the diversity of the two shapes comes from the g part. To see this, note that since $\sin x \sim x$ as $x \to 0$, we can approximate the variance as $\mathrm{Var}(xA) - \mathrm{Var}(\sqrt{2}xA) = (1 - \sqrt{2})^2 x^2 \mathrm{Var}(A)$, which is much greater than $\mathrm{Var}(Ax^2) = x^4 \mathrm{Var}(A)$ when x is small. Conversely, if x is large, then most of the diversity comes from the f part.

With entropy as a measure of diversity, again for independent X and Y, we obtain (applying the third Shannon-Khinchin axiom above to $\mathscr{H}(X + Y, Y)$, and noting that for independent X, Y we have $\mathscr{H}(X + Y|Y) = \mathscr{H}(Y)$) :

$$\mathscr{H}(X + Y) = \mathscr{H}(X) + \mathscr{H}(Y) - \mathscr{H}(Y|X + Y) \tag{A.2}$$

The last term $\mathscr{H}(Y|X + Y)$ is zero only if each sum $x + y \sim X + Y$ *uniquely* determines $y \sim Y$, which is generally not the case. Moreover, the diversity as measured by the entropy no longer has a clean dependency on the magnitude of x which means that we can no longer neatly divide the contributions of each of the parts to the entropy of $f + g$.

Hence, in general, the two measures of diversity thus correspond to different ways to combine component diversity. Similar arguments could be given for other statistics, which may have very different ways to be expressed in terms of statistics of given decompositions.

We can avoid these ambiguities by examining the decomposition of shape at a point $x + h$ into components that change at different rates with h, the distance to

x, specifically as a sum of terms proportional to powers of the distance h. Setting $u : \mathbb{R} \to \mathbb{R}$ for simplicity, we have:

$$u(x+h) = O(h^0) + O(h^1) + O(h^2) + \dots \tag{A.3}$$

When h is small and all constants multiplying the powers of h are of the same magnitude, most of the contribution to $u(x+h)$ comes from the first term $O(h^0)$; higher-order terms do not contribute much to the actual value of $u(x+t)$. To fix intuition, setting $h = 0.1$, one can think of the constant multiplying h^n as the n-th decimal digit in a decimal expansion, which becomes increasingly negligible for larger n.

Moreover, the provided u is sufficiently well-behaved since Taylor's Theorem allows us to compute the digits in this expansion using Taylor's expansion:

$$u(x+h) = \sum_{i=0}^{\infty} \frac{u^{(n)}(x)}{n!} h^n \tag{A.4}$$

Conversely, if h is large and all constants are of the same magnitude, then the least significant contribution to $u(x+h)$ comes from the first term.

In a physical setting "small" and "large" are given in terms of a length that is, in some way, fundamental to the problem: the *scale*. This brings us to the case where h is neither large nor small, that is, of order unity, $h = 1$, measured in that fundamental scale (specifically, in an image, fundamental length is given by the inter-pixel distance).

At the distance of one fundamental length, Taylor's formula thus reads:

$$u(x+1) = u(x) + u^{(1)}(x) + \frac{u^{(2)}(x)}{2!} + \dots, \tag{A.5}$$

which tells us how to weight each of the derivatives in order to compute u at a distance of 1 fundamental length from x. In other words, if we consider the value of the function and its derivatives at x, each contributes to the "shape" of the field at that scale, with higher-order derivatives being weighted by $1/n!$.

We use the reasoning above to introduce a weighting for shape statistics of a field in terms of the values of the field and k of its derivatives: that given by the terms of the Taylor expansion.

Specifically, given a statistic S of a field u (e.g. mean, variance, ...), we define its associated *Taylor-n statistic* as:

$$T^n S(u) = \sum_{k=0}^{n} \frac{S(u^{(k)})}{k!}. \tag{A.6}$$

Here, $u^{(k)}$ denotes the k-th derivative of u. If u is multi-dimensional (e.g. an image), an analogous expression can be written in terms of partial derivatives. Specifically, for $u : \mathbb{R}^p \to \mathbb{R}$ we have:

$$T^n S(u) = \sum_{|\alpha|=0}^{n} \frac{S(u^{(\alpha)})}{\alpha!}, \tag{A.7}$$

where α is a multi-index $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$ of length n, with $|\alpha| = \sum_{k=1}^{n} \alpha_k$, the factorial $\alpha!$ denoting $\alpha_1! \cdot \alpha_2! \cdots \alpha_n!$ and $u^{(\alpha)} := \frac{\partial^{\alpha_1 + \cdots \alpha_n}}{\partial x_1^{\alpha_1} \cdots \partial x_n^{\alpha_n}} u$.

We thus obtain a statistic that incorporates the value and the local shape, by weighting u and its spatial derivatives in the same proportion that they contribute to the change in u at a distance of one fundamental length from x, (namely $1/k!$).

With a satisfiable definition of pattern-statistics, we turn to the problem of computational efficiency. To do so, we turn to *permutation entropy* of Bandt and Pope[5], which we briefly recall. The problem is similar to ours, that of estimating dynamic complexity, but in the time-series setting. This can be done via traditional measures such as Lyapunov exponents or Kolmogorov-Sinai entropy, which are unfortunately intractable to compute and sensible to noise. To address these difficulties, the authors propose computing the Shannon entropy of *ordinal patterns* instead. Specifically, the ordinal pattern of length k at $u(t)$ is the permutation that arranges $u(t), u(t+1), u(t+2), \ldots, u(t+k-1)$ in increasing order. This measure is very simple and efficient to compute. Moreover, it can be shown that, in certain situations, it coincides with Kolmogorov-Sinai entropy, which is a measure of dynamical diversity in the sense of chaos, and which is generally intractable. Intuitively, this is possible because we replace the full time-series dynamics with a "collapsed" version that tracks local pattern dynamics. This collapse retains enough dynamic information to compute dynamic diversity, but is important enough that computation becomes possible.

Applying this measure to higher dimensions is not straightforward because there is no natural order in dimension $n \geq 1$. This motivates a more general definition of "pattern" that is applicable to higher dimensions. Suppose that the field u has been sampled at a number of grid points. Then the *sign-pattern field* associated to u is the mapping that assigns to each grid point x a binary vector of dimension equal to the number of neighbors of x and values 1 if u computed at that location is greater than the mean value of u and zero otherwise (the order of the components is arbitrary but fixed for all x). Specifically for a square grid in two dimensions, the sign pattern field of u at x^{ij}, which we denote by Pu^{ij} is the 9-component vector:

$$\begin{aligned} Pu^{ij} = (&u^{i-1,j-1} > 0, && u^{i-1,j} > 0, u^{i-1,j+1} > 0, \\ &u^{i,j} > 0, && u^{i,j-1} > 0, u^{i,j+1} > 0, \\ &u^{i+1,j-1} > 0, && u^{i+1,j} > 0, u^{i+1,j+1} > 0). \end{aligned}$$

Note that one can also see Pu^{ij} as a 9-bit binary number, and that computing a statistic on Pu, seen as a collection of independent samples, is straightforward.

Building on the concept of the sign-pattern field, we define the **Taylor-expanded LMC complexity** as a way to incorporate local geometric information from derivatives of the field. The central idea is to apply the Taylor-n statistic to the LMC complexity computed on the sign-pattern fields of the field itself and its derivatives.

Recall that the LMC complexity C of a discrete distribution $\{p_i\}_{i=1}^N$ is defined as

$$C = \mathscr{H} \times \mathscr{D}, \tag{A.8}$$

where $\mathscr{H}(p)$ is the Shannon entropy of the distribution:

$$\mathscr{H}(p) = -K \sum_{i=1}^{N} p_i \log p_i, \tag{A.9}$$

and $\mathscr{D}(p)$ is the disequilibrium:

$$\mathscr{D}(p) = \sum_{i=1}^{N} \left(p_i - \frac{1}{N} \right)^2 . \tag{A.10}$$

Here, N is the total number of possible configurations (for a 3×3 patch, $N = 2^9$), p_i is the probability of observing the i-th sign pattern, $\mathscr{H}(p)$ measures the disorder of the distribution, and $\mathscr{D}(p)$ measures the deviation from uniformity. The product $\mathscr{H}(p) \times \mathscr{D}(p)$ quantifies the structural complexity of the field in terms of both randomness and organization.

Specifically, let Pu denote the sign-pattern field of u as defined above. To capture local slope information, we compute sign-pattern fields Pu_x and Pu_y from the first-order partial derivatives of u, using finite differences on the grid. These fields encode whether the rate of change in each direction is positive or negative relative to the local neighborhood. To further capture curvature, we also compute second-order sign-pattern fields Pu_{xx}, Pu_{xy}, and Pu_{yy} corresponding to the second derivatives. The second-order derivatives encode concavity and convexity in each direction, as well as the local mixed curvature along diagonals.

The Taylor complexity of order 2 is then obtained by applying the LMC complexity function $C(\cdot)$ to each of these sign-pattern fields and combining them according to the weighting derived from the Taylor expansion:

$$\mathscr{C} = C(Pu) + C(Pu_x) + C(Pu_y) + \frac{1}{2}C(Pu_{xx}) + C(Pu_{xy}) + \frac{1}{2}C(Pu_{yy}).$$

Here, the factor $1/2$ applied to the pure second-order derivatives reflects the $1/2!$ weighting in the Taylor expansion, ensuring that each derivative contributes proportionally to its expected effect on the local shape at a distance of one fundamental length. In this formulation, first-order derivatives capture slope variations, while second-order derivatives capture curvature variations, resulting in a complexity measure that integrates both amplitude and shape information locally.

We adopt a second-order Taylor expansion because, in most self-organized surface patterns, the dominant contributions to local shape variations arise from the field values, the first-order derivatives (slopes), and the second-order derivatives (curvatures). Higher-order derivatives, such as third, fourth, or fifth order, contribute progressively less to the local shape at the scale of a single patch, due to the factorial weighting in the Taylor expansion ($1/k!$). Including these higher-order terms would significantly increase computational cost and the number of possible sign patterns, while adding minimal additional information about the relevant local structures. Consequently, second-order derivatives efficiently capture the essential geometric information needed to discriminate between flat, self-organized, and chaotic morphologies, ensuring both computational feasibility and statistical robustness.

Each term is computed as the LMC complexity over all 3×3 patches extracted from the corresponding field or derivative. For each patch, the sign pattern is treated as a discrete state, and the LMC complexity combines the Shannon entropy of the distribution of states with a disequilibrium factor, quantifying how far the distribution is from uniformity. The combination of entropy and disequilibrium ensures that $\mathscr{C}$ is maximal for highly structured, intermediate-order configurations, while it decreases for both completely ordered and completely random patterns.

The choice of 3×3 patches is critical for balancing spatial resolution and statistical robustness. Patches smaller than 3×3, such as 1×1, fail to capture any information about local slopes or curvature, since derivatives cannot be meaningfully computed

within a single pixel. Larger patches, such as 5×5, while potentially capturing more extended structures, lead to an exponential increase in the number of possible sign patterns (2^{25}), making the statistics sparse and reducing sensitivity to fine structural details. By using 3×3 patches, we ensure that first- and second-order derivatives are effectively represented while keeping the number of possible configurations manageable (2^9 per patch), preserving both local geometric information and computational feasibility.

The resulting Taylor complexity thus encodes the local diversity of shapes and amplitudes simultaneously. It provides a more informative measure than standard LMC complexity, which only considers local amplitude variations, because it also accounts for local slopes and curvatures, which are essential features of self-organized surface patterns. By summing contributions from derivatives of increasing order with appropriate Taylor weights, $\mathscr{C}$ gives a natural, scale-consistent measure of the complexity of local structures at a distance comparable to the fundamental length of the grid, i.e., the inter-pixel spacing in an image.

From a computational standpoint, the evaluation of $\mathscr{C}$ involves three main steps. First, all relevant derivatives of u are computed using finite differences on the discrete grid. Second, sign-pattern fields for each derivative and the original field are generated. Third, the LMC complexity is evaluated on each sign-pattern field, and the terms are combined according to the Taylor expansion weights. This approach is efficient because each patch provides an independent sample for the LMC calculation, and the total number of samples scales linearly with the number of grid points.

Figure A.1 illustrates how this approach quantifies surface entropy. The first row shows the sign distributions after mean subtraction for three representative cases: a flat surface, a well-organized pattern, and a chaotic morphology. The second row depicts the frequency of occurrence of each 3×3 sign pattern, sorted by index from least to most frequent. The inset highlights the four most common patterns, displayed in descending order of frequency. This visualization demonstrates how Taylor complexity discriminates between different morphologies: flat surfaces exhibit few dominant patterns, chaotic surfaces display a nearly uniform pattern distribution, and self-organized surfaces exhibit intermediate distributions that maximize complexity.

In summary, Taylor complexity extends the classical LMC measure by integrating information about local slopes and curvatures, providing a rich, multi-scale description of structural diversity. It is particularly well-suited for self-organized surface patterns, where subtle variations in local geometry are critical, and the combination of entropy and disequilibrium ensures sensitivity to intermediate-order structures that are characteristic of self-assembled morphologies. This formulation bridges the gap between purely amplitude-based measures and the need for geometric sensitivity, offering a systematic, computationally tractable framework for quantifying structural complexity in images.

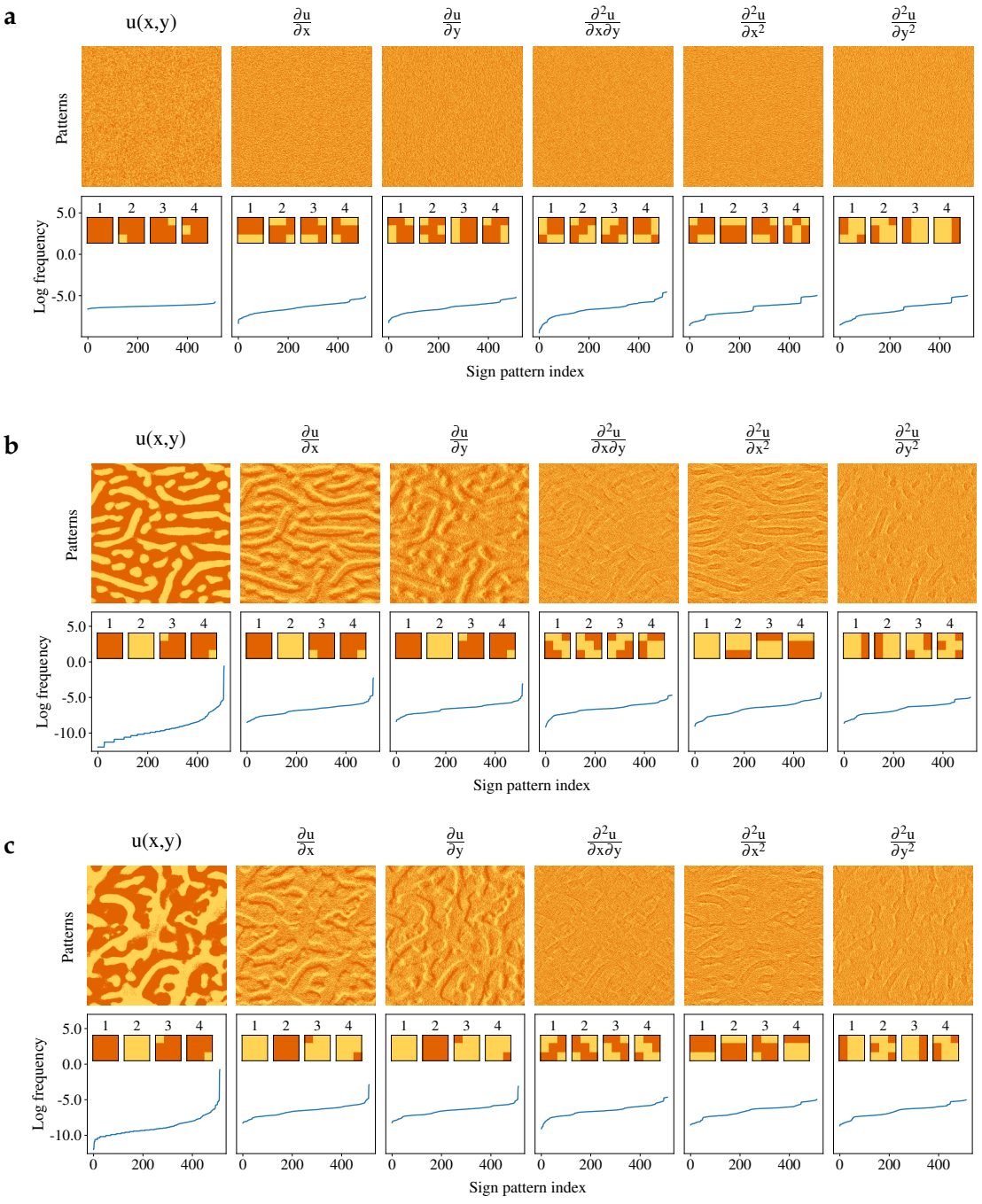


FIGURE A.1: **Illustration of entropy computation considering local information of a surface.** The plot represents three different patterns: (a) a flat pattern, (b) an organized pattern, and (c) a chaotic pattern. In each subfigure, the top row displays the sign of the indicated field after subtracting the mean: yellow for pixel values ≥ 0, orange for pixel values < 0. The second row shows the 3×3 sign distribution of the fields above. The small squares are the 4 most frequent patterns, with orange indicating pixel values < 0 and yellow indicating pixel values ≥ 0.

Appendix B

Unrolled-SINDy: Differential Equation Discovery under Sparse Data and Large Time-step Sampling

B.1 Locally Linearized Closed-Form Resolution

Even though we do not mention this third solution in the core of the manuscript, it is also possible to use a hybrid approach which iterates on a closed form (like the closed form version) but uses automatic differentiation (like the SGD version) to estimate a more accurate effective dictionary that takes into account the dependency of the effective dictionary on the parameters $\boldsymbol{\alpha}$.

We denote as $pred(\boldsymbol{\alpha})$ the prediction of the unrolled SINDy algorithm, based on a current estimate of $\boldsymbol{\alpha}$. For the sake of generality (to include K-Unrolled Euler-SINDy, K-Unrolled RK4-SINDy and RK4-SINDy), we assume the prediction is a function of the following form, involving an effective dictionary that is a function of $\boldsymbol{\alpha}$ (denoted $\Theta(\boldsymbol{\alpha})$ to insist on this dependence, and which corresponds to $\tilde{\boldsymbol{\Theta}}$ in Algorithms 1 and 2):

$$\forall j, m \quad pred(\boldsymbol{\alpha})(t_j, x_m) = u(t_j, x_m) + h_j \Theta(\boldsymbol{\alpha})(t_j, x_m) \cdot \boldsymbol{\alpha} \tag{B.1}$$

in more compact algorithmic/matrix form:

$$pred(\boldsymbol{\alpha}) = \mathbf{U}_{prev} + \mathbf{H} \cdot \Theta(\boldsymbol{\alpha}) \cdot \boldsymbol{\alpha} \tag{B.2}$$

reminding that $\mathbf{U}_{prev} \in \mathbb{R}^{N \times d_2}$, $\mathbf{H} \in \mathbb{R}^{N \times N}$, $\Theta(\boldsymbol{\alpha}) \in \mathbb{R}^{N \times |\Theta|}$ and $\boldsymbol{\alpha} \in \mathbb{R}^{|\Theta| \times d_2}$, and thus $pred \in (\mathbb{R}^{|\Theta| \times d_2} \to \mathbb{R}^{N \times d_2})$.

We can linearize the prediction using the Taylor expansion at the first order, around the current estimate $\boldsymbol{\alpha}$, with a variation $\delta\boldsymbol{\alpha} \in \mathbb{R}^{|\Theta| \times d_2}$:

$$\begin{aligned}
pred(\boldsymbol{\alpha} + \delta\boldsymbol{\alpha}) &\approx pred(\boldsymbol{\alpha}) + \frac{\partial pred}{\partial \boldsymbol{\alpha}}(\boldsymbol{\alpha}) \cdot \delta\boldsymbol{\alpha} && \text{(B.3)} \\
&= pred(\boldsymbol{\alpha}) + \mathbf{H} \cdot \left[\Theta(\boldsymbol{\alpha}) + \frac{\partial \Theta}{\partial \boldsymbol{\alpha}}(\boldsymbol{\alpha}) \cdot \boldsymbol{\alpha} \right] \cdot \delta\boldsymbol{\alpha} && \text{(B.4)} \\
&= pred(\boldsymbol{\alpha}) + \mathbf{H} \cdot [\Theta(\boldsymbol{\alpha}) + \boldsymbol{J}\boldsymbol{\alpha}] \cdot \delta\boldsymbol{\alpha} && \text{(B.5)} \\
&= \mathbf{U}_{prev} + \mathbf{H} \cdot \Theta(\boldsymbol{\alpha}) \cdot \boldsymbol{\alpha} + \mathbf{H} \cdot [\Theta(\boldsymbol{\alpha}) + \boldsymbol{J}\boldsymbol{\alpha}] \cdot \delta\boldsymbol{\alpha} && \text{(B.6)} \\
& && \text{(B.7)}
\end{aligned}$$

where $\boldsymbol{J} = \frac{\partial \Theta}{\partial \boldsymbol{\alpha}}(\boldsymbol{\alpha}) \in \mathbb{R}^{(N \times |\Theta|) \times (|\Theta| \times d_2)}$ is the Jacobian of the effective dictionary with respect to $\boldsymbol{\alpha}$, evaluated at the current estimate $\boldsymbol{\alpha}$, which can be obtained by

automatic differentiation. The tensor product of $\boldsymbol{J}$ by $\boldsymbol{\alpha}$ is to be understood as $(\boldsymbol{J}\boldsymbol{\alpha})_{ij} = \sum_{k,l} \boldsymbol{J}_{ijkl}\boldsymbol{\alpha}_{kl}$.

Neglecting higher order terms (in the Taylor expansion), we can iteratively update the estimate of $\boldsymbol{\alpha}$ by solving the following linear system, in $\delta\boldsymbol{\alpha}$:

$$\min_{\delta\boldsymbol{\alpha}} \left\|\mathbf{U}_{next} - \mathbf{U}_{prev} - \mathbf{H}\cdot\Theta(\boldsymbol{\alpha})\cdot\boldsymbol{\alpha} - \mathbf{H}\cdot\left[\Theta(\boldsymbol{\alpha}) + \boldsymbol{J}\boldsymbol{\alpha}\right]\cdot\delta\boldsymbol{\alpha}\right\|_{\mathcal{F}}^2 + \lambda\left\|\boldsymbol{\alpha}+\delta\boldsymbol{\alpha}\right\|_{\mathcal{F}}^2 \tag{B.8}$$

i.e.

$$\min_{\delta\boldsymbol{\alpha}} \left\|\left[\mathbf{U}_{next} - \mathbf{U}_{prev} - \mathbf{H}\cdot\Theta(\boldsymbol{\alpha})\cdot\boldsymbol{\alpha}\right] - \mathbf{H}\cdot\tilde{\tilde{\boldsymbol{\Theta}}}\cdot\delta\boldsymbol{\alpha}\right\|_{\mathcal{F}}^2 + \lambda\left\|[-\boldsymbol{\alpha}]-\delta\boldsymbol{\alpha}\right\|_{\mathcal{F}}^2 \tag{B.9}$$

with

$$\tilde{\tilde{\boldsymbol{\Theta}}} = \Theta(\boldsymbol{\alpha}) + \boldsymbol{J}\boldsymbol{\alpha} = \tilde{\boldsymbol{\Theta}} + \boldsymbol{J}\boldsymbol{\alpha} \tag{B.10}$$

In the end, we are using an updated effective dictionary $\tilde{\tilde{\boldsymbol{\Theta}}}$ that is the original effective dictionary $\tilde{\boldsymbol{\Theta}}$ plus $\boldsymbol{J}\boldsymbol{\alpha}$. This minimization can be achieved in closed form:

$$\delta\boldsymbol{\alpha} = \left[\tilde{\tilde{\boldsymbol{\Theta}}}^\top\tilde{\tilde{\boldsymbol{\Theta}}} + \lambda\mathbf{I}_{|\Theta|\times|\Theta|}\right]^{-1}\left[\tilde{\tilde{\boldsymbol{\Theta}}}^\top\dot{\boldsymbol{U}} - \lambda\boldsymbol{\alpha}\right] \tag{B.11}$$

with

$$\dot{\mathbf{U}} = \left(\mathbf{U}_{next} - \mathbf{U}_{prev} - \mathbf{H}\cdot\tilde{\boldsymbol{\Theta}}\cdot\boldsymbol{\alpha}\right)\oslash\mathcal{H} \tag{B.12}$$

$$= \left(\mathbf{U}_{next} - \mathbf{U}_{prev}\right)\oslash\mathcal{H} - \tilde{\boldsymbol{\Theta}}\cdot\boldsymbol{\alpha} \tag{B.13}$$

Then the estimate of $\boldsymbol{\alpha}$ can be updated as:

$$\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha} + \delta\boldsymbol{\alpha} \tag{B.14}$$

B.2 Identified governing equations and selection of unrolled value K

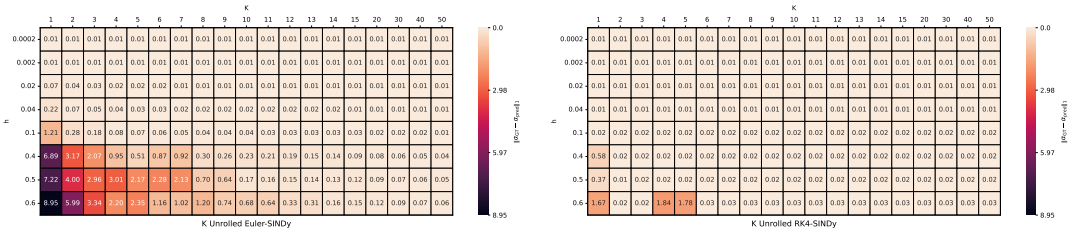
Tables B.1.A and B.1.B present the results and the choice of the unrolling depths K for cubic oscillator Eq. 5.15 when the number of pairs vary. Table B.2 correspond to the case where the number of training pairs is constant.

Tables B.6, B.7, B.5.A and B.5.B summarize the results for the 2D reaction-diffusion system Eq. 5.17

Tables B.3.A, B.3.B, B.4.A and B.4.B report similar results for the advection equation Eq. 5.16

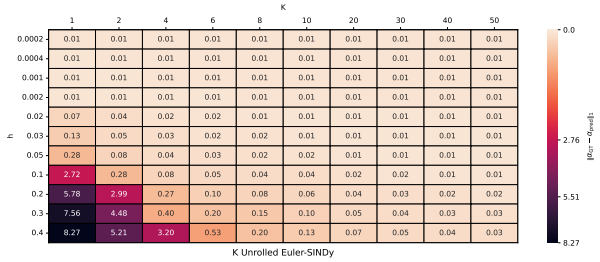
Finally, the Tables B.9 , B.8.A and B.8.B correspond to the Kuramoto–Sivashinsky equation Eq. 5.18

TABLE B.1: Accuracy of K-Unrolled SINDy methods with different unrolling depths K and observation step sizes h for the cubic damped oscillator (Eq. 5.15). (A) Euler-SINDy and (B) RK4-SINDy. Lighter colors indicate more accurate recovery.



(A) (B)

TABLE B.2: Accuracy of K-Unrolled Euler-SINDy with different unrolling depths K and observation step sizes h for the cubic damped oscillator (Eq. 5.15). While value h increases **the number of training pairs N is kept constant**.



B.3 Execution time results

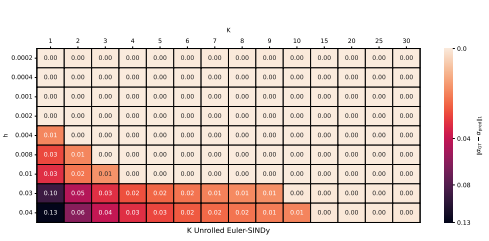
This section reports the computational cost of the different SINDy variants across the PDEs considered in this work. Tables B.10, B.11, and B.12 summarize the execution times for the advection, reaction-diffusion and Kuramoto–Sivashinsky equations, respectively. Unrolled methods have higher runtimes due to the multiple intermediate steps performed per integration interval, with the increase being more pronounced for higher-dimensional PDEs. These tables provide a reference for the trade-off between improved accuracy and additional computational effort.

Hardware. All experiments for unrolled SINDy were performed on a high-performance computing node, equipped with a Dell PowerEdge R7525 server. The system is configured with 512 GB of DDR4 memory and powered by two AMD EPYC 7F7 processors, each providing 24 cores per socket, for a total of 48 physical cores.

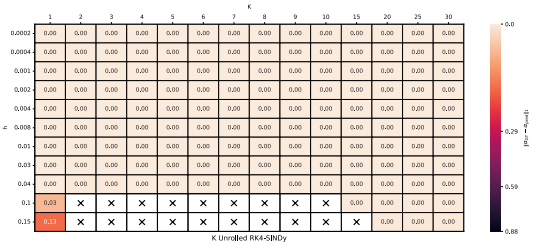
B.4 Dictionary terms used in the experiments

In Tables B.13 and B.14, we provide a comprehensive summary of the equations used in our experiments, along with the corresponding dictionaries of candidate terms employed during the discovery process. Table B.13 presents the PDEs and ODEs analyzed with Unrolled-SINDy, including the cubic damped oscillator, the advection equation, the reaction-diffusion system, and the Kuramoto–Sivashinsky equation. Table B.14 focuses on the equations used for Unrolled iNeural-SINDy experiments, specifically the cubic damped oscillator, linear oscillator and the Fitz-Hugh Nagumo

TABLE B.3: Accuracy of K-Unrolled SINDy with different unrolling depths K and observation step sizes h for the advection equation (Eq. 5.16). (a) Euler-SINDy and (b) RK4-SINDy. The marker X denotes entries where the results are NaN.



(A)



(B)

TABLE B.4: Robustness of SINDy methods on Eq. 5.16 with increasing time step h and decreasing number of learning pairs N. (A) Euler-SINDy and its 25-unrolled version. (B) RK4-SINDy and its 25-unrolled version. The number of additional wrong terms is indicated in red.

$h(N)$	**Euler-SINDy**	**25 Unrolled Euler-SINDy**
$h = 2e-04$ $(N = 10000)$	$-0.400u_x$	$-0.400u_x$
$h = 4e-04$ $(N = 5000)$	$-0.400u_x$	$-0.400u_x$
$h = 1e-03$ $(N = 2000)$	$-0.400u_x$	$-0.400u_x$
$h = 2e-03$ $(N = 1000)$	$-0.400u_x$	$-0.400u_x$
$h = 4e-03$ $(N = 500)$	$-0.400u_{x^{+1}}$	$-0.400u_x$
$h = 8e-03$ $(N = 250)$	$-0.400u_{x^{+1}}$	$-0.400u_x$
$h = 1e-02$ $(N = 200)$	$-0.400u_{x^{+1}}$	$-0.400u_x$
$h = 3e-02$ $(N = 67)$	$-0.400u_{x^{+1}}$	$-0.400u_x$
$h = 4e-02$ $(N = 50)$	$-0.399u_{x^{+1}}$	$-0.400u_x$

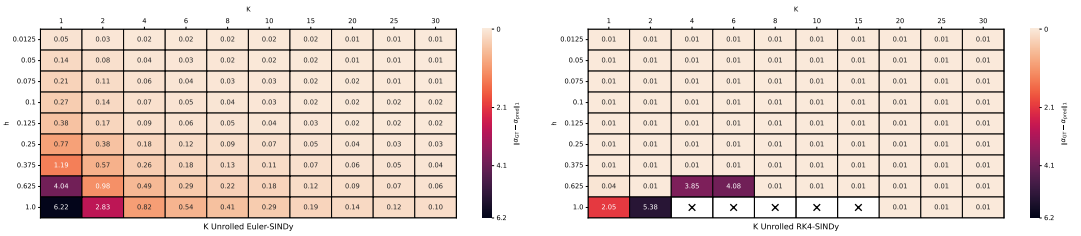
(A)

$h(N)$	**RK4-SINDy**	**25 Unrolled RK4-SINDy**
$h = 2e-04$ $(N = 10000)$	$-0.400u_x$	$-0.400u_x$
$h = 4e-04$ $(N = 5000)$	$-0.400u_x$	$-0.400u_x$
$h = 1e-03$ $(N = 2000)$	$-0.400u_x$	$-0.400u_x$
$h = 2e-03$ $(N = 1000)$	$-0.400u_x$	$-0.400u_x$
$h = 4e-03$ $(N = 500)$	$-0.400u_x$	$-0.400u_x$
$h = 8e-03$ $(N = 250)$	$-0.400u_x$	$-0.400u_x$
$h = 1e-02$ $(N = 200)$	$-0.400u_x$	$-0.400u_x$
$h = 3e-02$ $(N = 67)$	$-0.400u_x$	$-0.400u_x$
$h = 4e-02$ $(N = 50)$	$-0.400u_x$	$-0.400u_x$
$h = 1e-01$ $(N = 20)$	$-0.395u_{x^{+1}}$	$-0.400u_x$
$h = 1.5e-01$ $(N = 14)$	$-0.367u_{x^{+1}}$	$-0.400u_x$

(B)

system. Each row lists the name of the system, its analytical expression, and the full set of candidate terms considered in the sparse regression. These tables are presented in the appendix due to their size, but they provide essential details for reproducing the discovery experiments and understanding the structure of the dictionaries used in both classical and neural-network-based Unrolled-SINDy frameworks.

TABLE B.5: Accuracy of K-Unrolled SINDy with different unrolling depths K and observation step sizes h for the reaction-diffusion equation (Eq. 5.17). (A) K Unrolled Euler-SINDy and (B) K Unrolled RK4-SINDy. The marker X denotes entries where the results are NaN.



(A) (B)

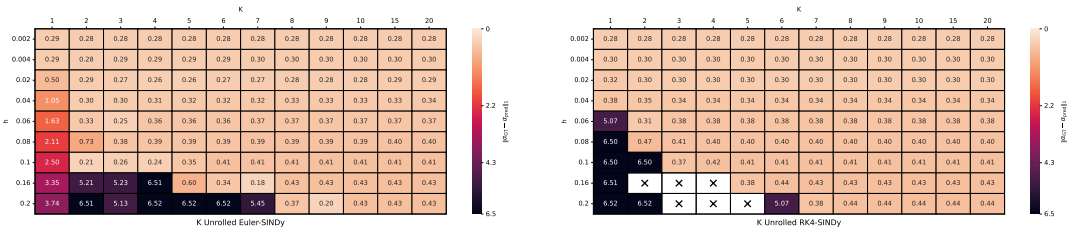
TABLE B.6: Robustness of Euler-SINDy and its unrolled version on Eq. 5.17, with an increasing time step h and a decreasing number of learning pairs. When Euler-SINDy fails to recover the governing equations (*i.e.* $Q(\boldsymbol{\alpha}) < 1$), only the number of wrong additional terms is indicated.

$h(N)$	**Euler-SINDy**	**25 Unr. Euler-SINDy**
$h = 1.250e-02$	$1.001u - 1.003u^3 + 1.001v^3 + 0.103u_{xx} + 0.102u_{yy} + 1.003u^2v - 1.004uv^2$	$1.000u - 1.000u^3 + 1.000v^3 + 0.101u_{xx} + 0.101u_{yy} + 1.001u^2v - 1.001uv^2$
$(N = 800)$	$1.005v - 1.001u^3 - 1.009v^3 + 0.102v_{xx} + 0.102v_{yy} - 1.008u^2v - 0.998uv^2$	$1.002v - 1.000u^3 - 1.002v^3 + 0.101v_{xx} + 0.100v_{yy} - 1.002u^2v - 0.997uv^2$
$h = 5.000e-02$	$1.002u - 1.010u^3 + 1.005v^3 + 0.108u_{xx} + 0.108u_{yy} + 1.009u^2v - 1.013uv^2$	$1.001u - 1.001u^3 + 1.000v^3 + 0.101u_{xx} + 0.101u_{yy} + 1.001u^2v - 1.001uv^2$
$(N = 200)$	$1.015v - 1.005u^3 - 1.028v^3 + 0.107v_{xx} + 0.107v_{yy} - 1.024u^2v - 1.001uv^2$	$1.001v - 1.001u^3 - 1.001v^3 + 0.101v_{xx} + 0.101v_{yy} - 1.001u^2v - 0.999uv^2$
$h = 7.500e-02$	$1.002u - 1.014u^3 + 1.008v^3 + 0.112u_{xx} + 0.112u_{yy} + 1.012u^2v - 1.019uv^2$	$1.001u - 1.001u^3 + 1.000v^3 + 0.101u_{xx} + 0.101u_{yy} + 1.001u^2v - 1.001uv^2$
$(N = 134)$	$1.022v - 1.008u^3 - 1.040v^3 + 0.111v_{xx} + 0.111v_{yy} - 1.035u^2v - 1.002uv^2$	$1.001v - 1.001u^3 - 1.002v^3 + 0.101v_{xx} + 0.101v_{yy} - 1.001u^2v - 0.999uv^2$
$h = 1.000e-01$	$1.002u - 1.018u^3 + 1.010v^3 + 0.115u_{xx} + 0.115u_{yy} + 1.016u^2v - 1.024uv^2$	$1.001u - 1.001u^3 + 1.000v^3 + 0.101u_{xx} + 0.101u_{yy} + 1.002u^2v - 1.002uv^2$
$(N = 100)$	$1.028v - 1.010u^3 - 1.053v^3 + 0.114v_{xx} + 0.114v_{yy} - 1.046u^2v - 1.003uv^2$	$1.002v - 1.001u^3 - 1.002v^3 + 0.101v_{xx} + 0.101v_{yy} - 1.001u^2v - 0.999uv^2$
$h = 1.250e-01$	$1.002u - 1.021u^3 + 1.012v^3 + 0.119u_{xx} + 0.119u_{yy} + 1.019u^2v - 1.029uv^2$	$1.001u - 1.001u^3 + 1.000v^3 + 0.102u_{xx} + 0.102u_{yy} + 1.002u^2v - 1.002uv^2$
$(N = 80)$	$0.981v - 0.933u^3 - 1.021v^3 + 0.105v_{xx} + 0.105v_{yy} - 1.014u^2v - 0.930uv^2{}_{+1}$	$1.002v - 1.001u^3 - 1.003v^3 + 0.101v_{xx} + 0.101v_{yy} - 1.002u^2v - 1.000uv^2$
$h = 2.500e-01$	$0.995u - 1.035u^3 + 1.020v^3 + 0.137u_{xx} + 0.137u_{yy} + 1.035u^2v - 1.048uv^2$	$1.001u - 1.002u^3 + 1.001v^3 + 0.102u_{xx} + 0.102u_{yy} + 1.002u^2v - 1.003uv^2$
$(N = 40)$	$0.941v - 0.846u^3 - 1.022v^3 + 0.106v_{xx} + 0.106v_{yy} - 1.010u^2v - 0.840uv^2{}_{+1}$	$1.003v - 1.001u^3 - 1.005v^3 + 0.102v_{xx} + 0.102v_{yy} - 1.004u^2v - 1.000uv^2$
$h = 3.750e-01$	$0.980u - 1.039u^3 + 1.023v^3 + 0.154u_{xx} + 0.154u_{yy} + 1.049u^2v - 1.056uv^2$	$1.001u - 1.003u^3 + 1.002v^3 + 0.103u_{xx} + 0.103u_{yy} + 1.003u^2v - 1.005uv^2$
$(N = 27)$	$0.881v - 0.740u^3 - 1.003v^3 + 0.105v_{xx} + 0.105v_{yy} - 0.987u^2v - 0.730uv^2{}_{+1}$	$1.004v - 1.002u^3 - 1.008v^3 + 0.103v_{xx} + 0.103v_{yy} - 1.006u^2v - 1.000uv^2$
$h = 6.250e-01$	$1.165u - 1.386u^3 + 0.691v^3 + 0.160u_{xx} + 0.160u_{yy} + 0.733u^2v - 1.383uv^2{}_{+3}$	$1.002u - 1.005u^3 + 1.003v^3 + 0.104u_{xx} + 0.104u_{yy} + 1.004u^2v - 1.008uv^2$
$(N = 16)$	$0.712v - 0.487u^3 - 0.914v^3 + 0.097v_{xx} + 0.097v_{yy} - 0.896u^2v - 0.467uv^2{}_{+1}$	$1.007v - 1.003u^3 - 1.013v^3 + 0.104v_{xx} + 0.104v_{yy} - 1.010u^2v - 1.001uv^2$
$h = 1.000e+00$	$1.112u - 1.469u^3 + 0.834v^3 + 0.209u_{xx} + 0.209u_{yy} + 0.878u^2v - 1.407uv^2{}_{+4}$	$1.002u - 1.008u^3 + 1.005v^3 + 0.106u_{xx} + 0.106u_{yy} + 1.005u^2v - 1.012uv^2$
$(N = 10)$	$0.331v - 0.641v^3 + 0.066v_{xx} + 0.066v_{yy} - 0.638u^2v_{+1}$	$1.012v - 1.005u^3 - 1.023v^3 + 0.106v_{xx} + 0.106v_{yy} - 1.015u^2v - 1.003uv^2$

TABLE B.7: Robustness of RK4-SINDy and its unrolled version on Eq. 5.17, with an increasing time step h and a decreasing number of learning pairs. When RK4-SINDy fails to recover the governing equations (*i.e.* $Q(\boldsymbol{\alpha}) < 1$), only the number of wrong additional terms is indicated.

$h(N)$	**RK4-SINDy**	**25 Unr. RK4-SINDy**
$h = 1.250e-02$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.100x_{yy} + 1.001x^2y - 1.001xy^2$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.100x_{yy} + 1.001x^2y - 1.001xy^2$
$(N = 800)$	$1.002y - 1.000x^3 - 1.002y^3 + 0.100y_{xx} + 0.100y_{yy} - 1.002x^2y - 0.997xy^2$	$1.002y - 1.000x^3 - 1.002y^3 + 0.100y_{xx} + 0.100y_{yy} - 1.002x^2y - 0.997xy^2$
$h = 5.000e-02$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$
$(N = 200)$	$1.001y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 1.000x^2y - 0.999xy^2$	$1.001y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 1.000x^2y - 0.999xy^2$
$h = 7.500e-02$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$
$(N = 134)$	$1.000y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 0.999xy^2$	$1.000y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 0.999xy^2$
$h = 1.000e-01$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$
$(N = 100)$	$1.000y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 0.999xy^2$	$1.000y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 0.999xy^2$
$h = 1.250e-01$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$
$(N = 80)$	$1.000y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 0.999xy^2$	$1.000y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 0.999xy^2$
$h = 2.500e-01$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$
$(N = 40)$	$1.000y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 0.999xy^2$	$1.000y - 1.000x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 0.999xy^2$
$h = 3.750e-01$	$1.001x - 1.001x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.001xy^2$	$1.001x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.001x^2y - 1.000xy^2$
$(N = 27)$	$1.000y - 1.001x^3 - 1.000y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 1.000xy^2$	$1.000y - 1.000x^3 - 0.999y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.999x^2y - 1.000xy^2$
$h = 6.250e-01$	$1.002x - 1.007x^3 + 1.002y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.005x^2y - 1.007xy^2$	$1.000x - 1.000x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 1.000x^2y - 1.000xy^2$
$(N = 16)$	$1.000y - 1.004x^3 - 1.005y^3 + 0.101y_{xx} + 0.101y_{yy} - 1.004x^2y - 1.003xy^2$	$1.000y - 1.000x^3 - 0.999y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.998x^2y - 1.000xy^2$
$h = 1.000e+00$	$0.950x - 0.950x^3 + 1.169y^3 + 0.114x_{xx} + 0.114x_{yy} + 1.056x^2y - 0.833xy^2{}_{+1}$	$1.000x - 0.999x^3 + 1.000y^3 + 0.101x_{xx} + 0.101x_{yy} + 0.999x^2y - 0.999xy^2$
$(N = 10)$	$0.890y - 0.611x^3 - 1.096y^3 - 1.029x^2y - 0.674xy^2{}_{+1}$	$0.999y - 1.000x^3 - 0.998y^3 + 0.101y_{xx} + 0.101y_{yy} - 0.997x^2y - 1.000xy^2$

TABLE B.8: Comparison of K-Unrolled SINDy methods with different unrolling depths K and observation step sizes h for the KS equation (Eq. 5.18). (A) Euler-SINDy, (B) RK4-SINDy. Marker X in (B) denotes NaN entries.



(A) (B)

TABLE B.9: Robustness of Euler-SINDy and its unrolled version on KS equation Eq. 5.18, with an increasing time step h and a decreasing number of learning pairs.

$h(N)$	Euler-SINDy	10 Unrolled Euler-SINDy	RK4-SINDy	10 Unrolled RK4-SINDy
$h = 2.000e-03$ ($N = 100000$)	$-1.081u_{xx} - 1.189u_{xxxx} - 4.979uu_x$	$-1.084u_{xx} - 1.193u_{xxxx} - 4.998uu_x$	$-1.085u_{xx} - 1.193u_{xxxx} - 5.000uu_x$	$-1.085u_{xx} - 1.193u_{xxxx} - 5.000uu_x$
$h = 4.000e-03$ ($N = 50000$)	$-1.079u_{xx} - 1.187u_{xxxx} - 4.974uu_x$	$-1.087u_{xx} - 1.196u_{xxxx} - 5.015uu_x$	$-1.088u_{xx} - 1.197u_{xxxx} - 5.018uu_x$	$-1.088u_{xx} - 1.197u_{xxxx} - 5.018uu_x$
$h = 2.000e-02$ ($N = 10000$)	$-0.983u_{xx} - 1.087u_{xxxx} - 4.603uu_x$	$-1.077u_{xx} - 1.188u_{xxxx} - 5.016uu_x$	$-1.084u_{xx} - 1.195u_{xxxx} - 5.046uu_x$	$-1.080u_{xx} - 1.191u_{xxxx} - 5.028uu_x$
$h = 4.000e-02$ ($N = 5000$)	$-0.861u_{xx} - 0.959u_{xxxx} - 4.131uu_x$	$-1.078u_{xx} - 1.192u_{xxxx} - 5.060uu_x$	$-0.990u_{xx} - 1.099u_{xxxx} - 4.724uu_x$	$-1.079u_{xx} - 1.193u_{xxxx} - 5.069uu_x$
$h = 6.000e-02$ ($N = 3334$)	$-0.761u_{xx} - 0.855u_{xxxx} - 3.750uu_x$	$-1.079u_{xx} - 1.194u_{xxxx} - 5.096uu_x$	$-0.180u_{xx} - 0.241u_{xxxx} - 1.512uu_x$	$-1.079u_{xx} - 1.195u_{xxxx} - 5.103uu_x$
$h = 8.000e-02$ ($N = 2500$)	$-0.681u_{xx} - 0.771u_{xxxx} - 3.440uu_x$	$-1.078u_{xx} - 1.195u_{xxxx} - 5.121uu_x$	$-0.500uu_x$	$-1.079u_{xx} - 1.195u_{xxxx} - 5.126uu_x$
$h = 1.000e-01$ ($N = 2000$)	$-0.614u_{xx} - 0.702u_{xxxx} - 3.185uu_x$	$-1.077u_{xx} - 1.195u_{xxxx} - 5.139uu_x$	$-0.497uu_x$	$-1.077u_{xx} - 1.195u_{xxxx} - 5.142uu_x$
$h = 1.600e-01$ ($N = 1250$)	$-0.468u_{xx} - 0.552u_{xxxx} - 2.631uu_x$	$-1.071u_{xx} - 1.192u_{xxxx} - 5.168uu_x$	$-0.489uu_x$	$-1.071u_{xx} - 1.192u_{xxxx} - 5.170uu_x$
$h = 2.000e-01$ ($N = 1000$)	$-0.400u_{xx} - 0.482u_{xxxx} - 2.375uu_x$	$-1.066u_{xx} - 1.188u_{xxxx} - 5.176uu_x$	$-0.484uu_x$	$-1.067u_{xx} - 1.189u_{xxxx} - 5.180uu_x$

TABLE B.10: Execution time (in seconds) for advection equation with $h = 0.002$ and $N = 1000$ pairs.

h=2e-03	**Euler-SINDy**	**25 Unr. Euler-SINDy**	**RK4-SINDy**	**25 Unr. RK4-SINDy**
$time[s]$	7.2 ± 0.5	12.0 ± 0.4	7.2 ± 0.4	27.9 ± 0.7

TABLE B.11: Execution time (in seconds) for reaction-diffusion equation with $h = 0.1$ and $N = 100$ pairs.

h=0.1	**Euler-SINDy**	**25 Unr. Euler-SINDy**	**RK4-SINDy**	**25 Unr. RK4-SINDy**
$time[s]$	94 ± 2	594 ± 12	298 ± 7	2001 ± 68

TABLE B.12: Execution time (in seconds) for Kuramoto-Sivashinsky equation with $h = 0.002$ and $N = 1000$ pairs.

h=0.02	**Euler-SINDy**	**10 Unr. Euler-SINDy**	**RK4-SINDy**	**10 Unr. RK4-SINDy**
$time[s]$	27	70	24	200

TABLE B.13: Summary of the equations discovered in the Unrolled SINDy experiments. Each row lists the name of the PDE/ODE, its analytical expression, and the dictionary of candidate terms used during the discovery process.

Equation Name	Equation Expression	Candidate Terms Dictionary
Cubic Damped Oscillator	$\begin{cases} u_t = -0.1\,u^3 + 2.0\,v^3, \\ v_t = -2.0\,u^3 - 0.1\,v^3 \end{cases}$	$\{1,\ u,\ v,\ u^2,\ uv,\ v^2,\ u^3,\ u^2v,\ uv^2,\ v^3,\ u^4,\ u^3v,\ u^2v^2,\ uv^3,\ v^4\}$
Advection Equation	$u_t = -0.4\,u_x$	$\{1,\ u,\ u^2,\ u^3,\ u_x,\ u_{xx},\ u_{xxx}\}$
2D Reaction-Diffusion Equation	$\begin{cases} u_t = u - u^3 + v^3 + 0.1\,u_{xx} + 0.1\,u_{yy} + u^2v - uv^2, \\ v_t = v - u^3 - v^3 + 0.1\,v_{xx} + 0.1\,v_{yy} - u^2v - uv^2 \end{cases}$	$\{1,\ u,\ v,\ u^2,\ v^2,\ u^3,\ v^3,\ u_x,\ v_x,\ u_y,\ v_y,\ u_{xx},\ v_{xx},\ u_{yy},\ v_{yy},\ u_{xy},\ v_{xy},\ uv,\ u^2v,\ uv^2\}$
Kuramoto-Sivashinsky Equation	$u_t = -u\,u_x - u_{xx} - 5\,u_{xxxx}$	$\{1,\ u,\ u_x,\ u_{xx},\ u_{xxx},\ u_{xxxx},\ uu_x\}$

TABLE B.14: Summary of equations discovered in Unrolled iNeural-SINDy experiments. Each row lists the name of the ODE/PDE, its analytical expression, and the dictionary of candidate terms used during the discovery process.

Equation Name	**Equation Expression**	**Candidate Terms Dictionary**
Cubic Damped Oscillator	$\begin{cases} u_t = -0.1\,u^3 + 2.0\,v^3, \\ v_t = -2.0\,u^3 - 0.1\,v^3 \end{cases}$	$\{1,\ u,\ v,\ u^2,\ uv,\ v^2,\ u^3,\ u^2v,\ uv^2,\ v^3\}$
Linear Damped Oscillator	$\begin{cases} u_t = -0.1\,u + 2.0\,v, \\ v_t = -2.0\,u - 0.1\,v \end{cases}$	$\{1,\ u,\ v,\ u^2,\ uv,\ v^2\}$
Fitz-Hugh Nagumo system	$\begin{cases} u_t = 1.0\,u - 1.0\,v - \frac{1}{3}u^3 + 0.1, \\ v_t = 0.1\,u - 0.1\,v \end{cases}$	$\{1,\ u,\ v,\ u^2,\ uv,\ v^2,\ u^3,\ u^2v,\ uv^2,\ v^3\}$

Bibliography

[1] W. Ross Ashby. "Principles of the self-organizing dynamic system". In: *The Journal of General Psychology* 37.2 (1947), pp. 125–128.

[2] Alexander Baev et al. "Metaphotonics: An emerging field with opportunities and challenges". In: *Physics Reports* 594 (2015), pp. 1–60.

[3] Anthony Baez et al. "Guaranteeing Conservation Laws with Projection in Physics-Informed Neural Networks". In: *NeurIPS 2024 Workshop on Data-driven and Differentiable Simulations, Surrogates, and Solvers*.

[4] Philip Ball. "The self-made tapestry: pattern formation in nature". In: (1999).

[5] Christoph Bandt and Bernd Pompe. "Permutation Entropy: a Natural Complexity Measure for Time Series". In: *Physical review letters* 88.17 (2002), p. 174102.

[6] Jean-Pierre Bérenger. "A perfectly matched layer for the absorption of electromagnetic waves". In: *Journal of Computational Physics* 114.2 (1994), pp. 185–200. ISSN: 0021-9991.

[7] Jean-Pierre Berenger. "Three-Dimensional Perfectly Matched Layer for the Absorption of Electromagnetic Waves". In: *J. Comput. Phys.* 127.2 (1996), 363–379. ISSN: 0021-9991.

[8] Giorgio Bertotti. *Hysteresis in magnetism: for physicists, materials scientists, and engineers*. Academic press, 1998.

[9] Tom Beucler et al. "Enforcing analytic constraints in neural networks emulating physical systems". In: *Physical review letters* 126.9 (2021), p. 098302.

[10] ME Bitterman. "Classical conditioning since Pavlov". In: *Review of general psychology* 10.4 (2006), pp. 365–376.

[11] Lorenzo Boninsegna, Feliks Nüske, and Cecilia Clementi. "Sparse learning of stochastic dynamical equations". In: *The Journal of Chemical Physics* 148.24 (Mar. 2018). ISSN: 1089-7690. DOI: 10.1063/1.5018409. URL: http://dx.doi.org/10.1063/1.5018409.

[12] Jörn Bonse and Stephan Gräf. "Maxwell meets Marangoni—a review of theories on laser-induced periodic surface structures". In: *Laser & Photonics Reviews* 14.10 (2020), p. 2000215.

[13] Jörn Bonse, Sabrina V Kirner, and Jörg Krüger. "Laser-induced periodic surface structures (LIPSS)". In: *Handbook of laser micro-and nano-engineering* (2020), pp. 1–59.

[14] Gert-Jan Both et al. "DeepMoD: Deep learning for model discovery in noisy data". In: *Journal of Computational Physics* 428 (Mar. 2021), p. 109985. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2020.109985. URL: http://dx.doi.org/10.1016/j.jcp.2020.109985.

[15] Eduardo Brandao. "Complexity Methods in Physics-Guided Machine Learning". PhD thesis. Jean Monnet University, France, 2024.

[16] Eduardo Brandao et al. “Learning Complexity To Guide Light-Induced Self-Organized Nanopatterns”. In: *Physical Review Letters* 130.22 (May 2023). ISSN: 1079-7114. DOI: 10.1103/physrevlett.130.226201. URL: http://dx.doi.org/10.1103/PhysRevLett.130.226201.

[17] Eduardo Brandao et al. “Learning Pde To Model Self-Organization of Matter”. In: *Entropy* 24.8 (2022). ISSN: 1099-4300. DOI: 10.3390/e24081096. URL: https://www.mdpi.com/1099-4300/24/8/1096.

[18] Daniel A Braun, Carsten Mehring, and Daniel M Wolpert. “Structure learning in action”. In: *Behavioural brain research* 206.2 (2010), pp. 157–165.

[19] Dennis Bray. “Protein molecules as computational elements in living cells”. In: *Nature* 376.6538 (1995), pp. 307–312.

[20] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. “Discovering governing equations from data by sparse identification of nonlinear dynamical systems”. In: *Proceedings of the national academy of sciences* 113.15 (2016), pp. 3932–3937.

[21] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. “Sparse identification of nonlinear dynamics with control (SINDYc)”. In: *IFAC-PapersOnLine* 49.18 (2016), pp. 710–715.

[22] Nina Byers. “E. Noether’s discovery of the deep connection between symmetries and conservation laws”. In: *arXiv preprint physics/9807044* (1998).

[23] JH Byrne. “Chapter 7: Learning and memory”. In: *Neuroscience Online: An Electronic Textbook for the Neurosciences. The University of Texas Health Science Center at Houston (UTHealth)* (1997).

[24] Ceyron. *Kuramoto-Sivashinsky Solver with Exponential Time Differencing in JAX*. Accessed: 2025-09-17. 2025. URL: https://github.com/Ceyron/machine-learning-and-simulation/blob/main/english/fft_and_spectral_methods/ks_solver_etd_in_jax.ipynb.

[25] MM Chaves, J Flexas, and Carla Pinheiro. “Photosynthesis under drought and salt stress: regulation mechanisms from whole plant to cell”. In: *Annals of botany* 103.4 (2009), pp. 551–560.

[26] Chao Chen, Hui Li, and Xiaowei Jin. “An invariance constrained deep learning network for partial differential equation discovery”. In: *Physics of Fluids* 36.4 (2024).

[27] Ricky TQ Chen et al. “Neural ordinary differential equations”. In: *Advances in neural information processing systems* 31 (2018).

[28] Leon Chua. “Memristor-the missing circuit element”. In: *IEEE Transactions on circuit theory* 18.5 (2003), pp. 507–519.

[29] Taco Cohen and Max Welling. “Group equivariant convolutional networks”. In: *International conference on machine learning*. PMLR. 2016, pp. 2990–2999.

[30] Thomas M Cover, Joy A Thomas, et al. “Entropy, relative entropy and mutual information”. In: *Elements of information theory* 2.1 (1991), pp. 12–13.

[31] M. C. Cross and P. C. Hohenberg. “Pattern formation outside of equilibrium”. In: *Rev. Mod. Phys.* 65.3 (July 1993), pp. 851–1112. ISSN: 1539-0756. DOI: 10.1103/RevModPhys.65.851.

[32] Matthew Dawber, KM Rabe, and JF Scott. “Physics of thin-film ferroelectric oxides”. In: *Reviews of modern physics* 77.4 (2005), pp. 1083–1130.

[33] Mari Cruz Díaz-Barradas et al. "How do Mediterranean shrub species cope with shade? Ecophysiological response to different light intensities". In: *Plant Biology* 20.2 (2018), pp. 296–306.

[34] Michael Domjan and Burkhard. "The principles of learning". In: *Belmont, CA: Wadsworth/Thomas Learning* (2003).

[35] Nathan Doumèche, Gérard Biau, and Claire Boyer. "On the convergence of PINNs". In: *Bernoulli* 31.3 (2025), pp. 2127 –2151. DOI: 10.3150/24-BEJ1799.

[36] Jiafei Duan et al. "A Survey on Machine Learning Approaches for Modelling Intuitive Physics". In: *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization. 2022, pp. 5444–5452.

[37] Benjamin Dusser et al. "Controlled nanostructrures formation by ultra fast laser pulses for color marking". In: *Optics express* 18.3 (2010), pp. 2913–2924.

[38] Charles El Helou et al. "Intelligent electroactive material systems with self-adaptive mechanical memory and sequential logic". In: *Proceedings of the National Academy of Sciences* 121.14 (2024), e2317340121.

[39] Balint Eles et al. "Mechanisms driving self-organization phenomena in random plasmonic metasurfaces under multipulse femtosecond laser exposure: a multitime scale study". In: *Nanophotonics* 11.10 (2022), pp. 2303–2318.

[40] Irving R Epstein and John A Pojman. *An introduction to nonlinear chemical dynamics: oscillations, waves, patterns, and chaos*. Oxford university press, 1998.

[41] Urban Fasel et al. "Ensemble-SINDy: Robust sparse model discovery in the low-data, high-noise limit, with active learning and control". In: *Proceedings of The Royal Society A: Mathematical, Physical and Engineering Sciences* (2022). DOI: 10.1098/rspa.2021.0904.

[42] Vladimir Yu Fedorov and Jean-Philippe Colombier. "Light-matter interaction at rough surfaces: A morphological perspective on laser-induced periodic surface structures". In: *Physical Review B* 110.11 (2024), p. 115438.

[43] John D Ferry. *Viscoelastic properties of polymers*. John Wiley & Sons, 1980.

[44] Vivian E Ferry, Albert Polman, and Harry A Atwater. "Modeling light trapping in nanostructured solar cells". In: *ACS nano* 5.12 (2011), pp. 10055–10064.

[45] Richard J Field and Maria Burger. "Oscillations and traveling waves in chemical systems". In: *(No Title)* (1985).

[46] Camilla Fiorini et al. "Generalizing the SINDy approach with nested neural networks". In: *ESAIM: Proceedings and Surveys* 81 (2025), pp. 168–192.

[47] Ali Forootani. *iNeural-SINDy: Implementation*. https://github.com/Ali-Forootani/iNeural_SINDy_paper. 2023.

[48] Ali Forootani, Pawan Goyal, and Peter Benner. "A robust sparse identification of nonlinear dynamics approach by combining neural networks and an integral form". In: *Engineering Applications of Artificial Intelligence* 149 (2025), p. 110360.

[49] Fabian Fuchs et al. "Se (3)-transformers: 3d roto-translation equivariant attention networks". In: *Advances in neural information processing systems* 33 (2020), pp. 1970–1981.

[50] Ian Goodfellow et al. *Deep learning*. Vol. 1. 2. MIT press Cambridge, 2016.

[51] Pawan Goyal and Peter Benner. "Discovery of nonlinear dynamical systems using a Runge-Kutta inspired dictionary-based sparse regression approach". In: *Proceedings. Mathematical, physical, and engineering sciences* 478 (June 2022), p. 20210883. DOI: 10.1098/rspa.2021.0883.

[52] Samuel Greydanus, Misko Dzamba, and Jason Yosinski. "Hamiltonian neural networks". In: *Advances in neural information processing systems* 32 (2019).

[53] Ernst Hairer, Syvert P. Nørsett, and Gerhard Wanner. *Solving ordinary differential equations. I: Nonstiff problems.* English. 2nd revised ed., 3rd corrected printing. Vol. 8. Springer Ser. Comput. Math. Berlin: Springer, 2010. ISBN: 978-3-642-05163-0.

[54] Zhongkai Hao et al. "Physics-informed machine learning: A survey on problems, methods and applications". In: *arXiv preprint arXiv:2211.08064* (2022).

[55] Moritz Hoffmann, Christoph Fröhner, and Frank Noé. *Reactive SINDy: Discovering governing reactions from concentration data*. Oct. 2018. DOI: 10.1101/442095.

[56] Serim Ilday et al. "Rich complex behaviour of self-assembled nanoparticles far from equilibrium". In: *Nature communications* 8.1 (2017), p. 14942.

[57] Ameya D Jagtap, Ehsan Kharazmi, and George Em Karniadakis. "Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems". In: *Computer Methods in Applied Mechanics and Engineering* 365 (2020), p. 113028.

[58] Kadierdan Kaheman, J. Nathan Kutz, and Steven L. Brunton. "SINDy-PI: a robust algorithm for parallel implicit sparse identification of nonlinear dynamics". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 476.2242 (Oct. 2020). ISSN: 1471-2946. DOI: 10.1098/rspa.2020.0279. URL: http://dx.doi.org/10.1098/rspa.2020.0279.

[59] Eric R Kandel et al. *Principles of neural science*. Vol. 4. McGraw-hill New York, 2000.

[60] George Em Karniadakis et al. "Physics-informed machine learning". In: *Nature Reviews Physics* 3.6 (May 2021). ISSN: 2522-5820.

[61] Nathan C Keim and Sidney R Nagel. "Generic transient memory formation in disordered systems with noise". In: *Physical review letters* 107.1 (2011), p. 010603.

[62] Nathan C Keim et al. "Memory formation in matter". In: *Reviews of Modern Physics* 91.3 (2019), p. 035002.

[63] EA Keshner, JHJ Allum, and CR Pfaltz. "Postural coactivation and adaptation in the sway stabilizing responses of normals and patients with bilateral vestibular deficit". In: *Experimental brain research* 69.1 (1987), pp. 77–92.

[64] A Ya Khinchin. *Mathematical foundations of information theory*. Courier Corporation, 2013.

[65] Shigeru Kondo and Takashi Miura. "Reaction-diffusion model as a framework for understanding biological pattern formation". In: *science* 329.5999 (2010), pp. 1616–1620.

[66] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. "Estimating mutual information". In: *Physical Review E-Statistical, Nonlinear, and Soft Matter Physics* 69.6 (2004), p. 066138.

[67] Hans Lambers, F Stuart Chapin III, and Thijs L Pons. *Plant physiological ecology*. Springer Science & Business Media, 2008.

[68] Abdeldjalil Latrach et al. "A critical review of physics-informed machine learning applications in subsurface energy systems". In: *Geoenergy Science and Engineering* 239 (2024), p. 212938.

[69] Zongyi Li et al. "Fourier Neural Operator for Parametric Partial Differential Equations". In: *International Conference on Learning Representations*.

[70] Zongyi Li et al. "Physics-informed neural operator for learning partial differential equations". In: *ACM/IMS Journal of Data Science* 1.3 (2024), pp. 1–27.

[71] Joowon Lim and Demetri Psaltis. "MaxwellNet: Physics-driven deep neural network training based on Maxwell's equations". In: *APL Photonics* 7.1 (Jan. 2022), p. 011301. ISSN: 2378-0967.

[72] Q Yu Linda, Robert C Wilson, and Matthew R Nassar. "Adaptive learning is structure learning in time". In: *Neuroscience & Biobehavioral Reviews* 128 (2021), pp. 270–281.

[73] J. M. Liu. "Simple technique for measurements of pulsed Gaussian-beam spot sizes". In: *Opt. Lett.* 7.5 (1982), pp. 196–198. ISSN: 1539-4794. DOI: 10.1364/OL.7.000196.

[74] Amanda L Lizier and Ann Reich. "Learning through work and structured learning and development systems in complex adaptive organisations: ongoing disconnections". In: *Studies in Continuing Education* 43.2 (2021), pp. 261–276.

[75] Gang Long et al. "PHC-GAN: Physical Constraint Generative Adversarial Network for Single Image Dehazing". In: *32nd IEEE ICTAI conference*. IEEE, 2020, pp. 545–549.

[76] Zichao Long et al. "Pde-net: Learning pdes from data". In: *International conference on machine learning*. PMLR. 2018, pp. 3208–3216.

[77] Ricardo Lopez-Ruiz, Hector L Mancini, and Xavier Calbet. "A statistical measure of complexity". In: *Physics letters A* 209.5-6 (1995), pp. 321–326.

[78] Lu Lu et al. "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators". In: *Nature machine intelligence* 3.3 (2021), pp. 218–229.

[79] Lu Lu et al. "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators". In: *Nature machine intelligence* 3.3 (2021), pp. 218–229.

[80] Yu-Wei Lu, Ling-Yan Li, and Jing-Feng Liu. "Influence of surface roughness on strong light-matter interaction of a quantum emitter-metallic nanoparticle system". In: *Scientific Reports* 8.1 (2018), p. 7115.

[81] Chunyue Lv, Lei Wang, and Chenming Xie. "A hybrid physics-informed neural network for nonlinear partial differential equation". In: *International Journal of Modern Physics C* 34.06 (2023), p. 2350082.

[82] SA Maier. *Plasmonics: Fundamentals and Applications*. 2007.

[83] Ghaith Makey et al. "Universality of dissipative self-assembly from quantum dots to human cells". In: *Nature Physics* 16.7 (2020), pp. 795–801.

[84] Carver Mead. "Neuromorphic electronic systems". In: *Proceedings of the IEEE* 78.10 (2002), pp. 1629–1636.

[85] Chuizheng Meng et al. "When physics meets machine learning: A survey of physics-informed machine learning". In: *Machine Learning for Computational Science and Engineering* 1.1 (2025), p. 20.

[86] Xuhui Meng et al. "PPINN: Parareal physics-informed neural network for time-dependent PDEs". In: *Computer Methods in Applied Mechanics and Engineering* 370 (2020), p. 113250.

[87] Daniel A. Messenger and David M. Bortz. "Weak SINDy for partial differential equations". In: *Journal of Computational Physics* 443 (Oct. 2021), p. 110525. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2021.110525. URL: http://dx.doi.org/10.1016/j.jcp.2021.110525.

[88] Elizabeth M Middleton and Alan H Teramura. "Understanding photosynthesis, pigment and growth responses induced by UV-B and UV-A irradiances". In: *Photochemistry and photobiology* 60.1 (1994), pp. 38–45.

[89] G. Nicolis and I. Prigogine. *Self-organization in nonequilibrium systems: From dissipative structures to order through fluctuations*. New York: Wiley-Interscience, 1977.

[90] Lukas Novotny and Bert Hecht. *Principles of nano-optics*. Cambridge university press, 2012.

[91] Keiron O'shea and Ryan Nash. "An introduction to convolutional neural networks". In: *arXiv preprint arXiv:1511.08458* (2015).

[92] Emanuel Parzen. "On estimation of a probability density function and mode". In: *The annals of mathematical statistics* 33.3 (1962), pp. 1065–1076.

[93] Josep Peñuelas and Joan Llusià. "BVOCs: plant defense against climate warming?" In: *Trends in plant science* 8.3 (2003), pp. 105–109.

[94] George Perrakis et al. "Impact of Hybrid Electromagnetic Surface Modes on the Formation of Low Spatial Frequency LIPSS: A Universal Approach". In: *Laser & Photonics Reviews* (2024), p. 2301090.

[95] David Pfau et al. "Ab initio solution of the many-electron Schrödinger equation with deep neural networks". In: *Physical review research* 2.3 (2020), p. 033429.

[96] Luis S Piloto et al. "Intuitive physics learning in a deep-learning model inspired by developmental psychology". In: *Nature human behaviour* 6.9 (2022), pp. 1257–1267.

[97] Ilya Prigogine and Isabelle Stengers. *Order Out of Chaos: Man's New Dialogue with Nature*. Bantam Books, 1984.

[98] Ilya Prigogine and Pierre Van Rysselberghe. "Introduction to thermodynamics of irreversible processes". In: *Journal of The Electrochemical Society* 110.4 (1963), p. 97C.

[99] Alfio Quarteroni and Alberto Valli. *Numerical approximation of partial differential equations*. Springer, 1994.

[100] Md Ashiqur Rahman, Zachary E Ross, and Kamyar Azizzadenesheli. "U-NO: U-shaped Neural Operators". In: *Transactions on Machine Learning Research* (2023). ISSN: 2835-8856. URL: https://openreview.net/forum?id=j3oQF9coJd.

[101] Maziar Raissi and George Em Karniadakis. "Hidden physics models: Machine learning of nonlinear partial differential equations". In: *Journal of Computational Physics* 357 (2018), pp. 125–141.

[102] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational physics* (2019).

[103] Juergen Reif. "Dynamics and processes on laser-irradiated surfaces". In: *Nanomaterials* 13.3 (2023), p. 379.

[104] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer. 2015, pp. 234–241.

[105] Anton Rudenko and Jean-Philippe Colombier. "How light drives material periodic patterns down to the nanoscale". In: *Ultrafast Laser Nanostructuring: The Pursuit of Extreme Scales*. Springer, 2023, pp. 209–255.

[106] Anton Rudenko et al. "High-frequency periodic patterns driven by non-radiative fields coupled with Marangoni convection instabilities on laser-excited metal surfaces". In: *Acta Materialia* 194 (2020), pp. 93–105.

[107] Anton Rudenko et al. "Self-organization of surfaces on the nanoscale by topography-mediated selection of quasi-cylindrical and plasmonic waves". In: *Nanophotonics* 8.3 (2019), pp. 459–465.

[108] Cynthia Rudin. "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead". In: *Nature machine intelligence* 1.5 (2019), pp. 206–215.

[109] Samuel H Rudy et al. "Data-driven discovery of partial differential equations". In: *Science advances* 3.4 (2017), e1602614.

[110] Shashank Sarbada and Yung C Shin. "Superhydrophobic contoured surfaces created on metal and polymer using a femtosecond laser". In: *Applied Surface Science* 405 (2017).

[111] Kristof Schütt et al. "Schnet: A continuous-filter convolutional neural network for modeling quantum interactions". In: *Advances in neural information processing systems* 30 (2017).

[112] Claude Elwood Shannon. "A mathematical theory of communication". In: *ACM SIGMOBILE mobile computing and communications review* 5.1 (2001), pp. 3–55.

[113] Jon Shiach. *Introduction to Ordinary Differential Equations*. Accessed: 2025-09-16. n.d. URL: https://jonshiach.github.io/ODEs-book/intro.html.

[114] Yasuhiko Shimotsuma et al. "Self-organized nanogratings in glass irradiated by ultrashort light pulses". In: *Physical review letters* 91.24 (2003), p. 247405.

[115] Khemraj Shukla et al. "Scalable algorithms for physics-informed neural and graph networks". In: *Data-Centric Engineering* 3 (2022), e24.

[116] Juan C Simo and Thomas JR Hughes. *Computational inelasticity*. Springer, 1998.

[117] JZP Skolski, GRBE Römer, J Vincenc Obona, et al. "Modeling laser-induced periodic surface structures: Finite-difference time-domain feedback simulations". In: *Journal of Applied Physics* 115.10 (2014).

[118] Larry R Squire. "Memory systems of the brain: a brief history and current perspective". In: *Neurobiology of learning and memory* 82.3 (2004), pp. 171–177.

[119] Robert Stephany and Christopher Earls. "PDE-LEARN: Using deep learning to discover partial differential equations from noisy, limited data". In: *Neural Networks* 174 (2024), p. 106242. ISSN: 0893-6080. DOI: https://doi.org/10.1016/j.neunet.2024.106242. URL: https://www.sciencedirect.com/science/article/pii/S0893608024001667.

[120] Robert Stephany and Christopher Earls. "PDE-READ: Human-readable partial differential equation discovery using deep learning". In: *Neural Networks* 154 (2022), pp. 360–382.

[121] Razvan Stoian and Jean-Philippe Colombier. "Advances in ultrafast laser structuring of materials at the nanoscale". In: *Nanophotonics* 9.16 (2020), pp. 4665–4688.

[122] E. Stratakis et al. "Laser engineering of biomimetic surfaces". In: *Materials Science and Engineering: R: Reports* 141 (July 2020), p. 100562. ISSN: 0927-796X. DOI: 10.1016/j.mser.2020.100562.

[123] Koji Sugioka and Ya Cheng. *Ultrafast laser processing: from micro-to nanoscale*. CRC Press, 2013.

[124] Ju Swift and Pierre C Hohenberg. "Hydrodynamic fluctuations at the convective instability". In: *Physical Review A* 15.1 (1977), p. 319.

[125] Makoto Takamoto et al. "Pdebench: An extensive benchmark for scientific machine learning". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 1596–1611.

[126] Alan H Teramura and Joe H Sullivan. "Effects of UV-B radiation on photosynthesis and growth of terrestrial plants". In: *Photosynthesis research* 39 (1994), pp. 463–473.

[127] Masahiro Toiya, Justin Stambaugh, and Wolfgang Losert. "Transient and oscillatory granular shear flow". In: *Physical review letters* 93.8 (2004), p. 088001.

[128] TF Torrance and James Clerk Maxwell. *A Dynamical Theory of the Electromagnetic Field*. 1982.

[129] George D Tsibidis, Costas Fotakis, and Emmanuel Stratakis. "From ripples to spikes: A hydrodynamical mechanism to interpret femtosecond laser-induced self-assembled structures". In: *Physical Review B* 92.4 (2015), p. 041405.

[130] Alan M Turing. "The Chemical Basis of Morphogenesis". In: *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences* 237 (1952), pp. 37–72.

[131] P. Valtr. "Probability That Random Points Are in Convex Position". In: *Discrete & Computational Geometry* 13.3 (1995), pp. 637–643. ISSN: 1432-0444. DOI: 10.1007/BF02574070. URL: https://doi.org/10.1007/BF02574070.

[132] A. Y. Vorobyev and Chunlei Guo. "Colorizing metals with femtosecond laser pulses". In: (2008). ISSN: 0003-6951.

[133] Manabu Wakuda et al. "Effect of surface texturing on friction reduction between ceramic and steel materials under lubricated sliding contact". In: *Wear* 254.3-4 (2003), pp. 356–363.

[134] Han Wang et al. "Leaf morphological traits as adaptations to multiple climate gradients". In: *Journal of Ecology* 110.6 (2022), pp. 1344–1355.

[135] BT Werner. "Eolian dunes: computer simulations and attractor interpretation". In: *Geology* 23.12 (1995), pp. 1107–1110.

[136] Norbert Wiener. *Cybernetics or Control and Communication in the Animal and the Machine*. MIT press, 2019.

[137] Jared Willard et al. "Integrating physics-based modeling with machine learning: A survey". In: *arXiv preprint arXiv:2003.04919* 1.1 (2020), pp. 1–34.

[138] Heng Yao et al. "Materials roadmap for inscription of nanogratings inside transparent dielectrics using ultrafast lasers". In: *Progress in Materials Science* (2023), p. 101226.

[139] Jianwu Yao et al. "Selective appearance of several laser-induced periodic surface structure patterns on a metal surface using structural colors produced by femtosecond laser pulses". In: *Applied Surface Science* 258.19 (2012), pp. 7625–7632.

[140] Tian Ye et al. "Interpretable intuitive physics model". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 87–102.

[141] Yuan Yin et al. "Augmenting physical models with deep networks for complex dynamics forecasting*". In: *Journal of Statistical Mechanics: Theory and Experiment* 2021.12 (Dec. 2021), p. 124012. ISSN: 1742-5468. DOI: 10.1088/1742-5468/ac3ae5. URL: http://dx.doi.org/10.1088/1742-5468/ac3ae5.

[142] Chengtao Yu et al. "Hydrogels as dynamic memory with forgetting ability". In: *Proceedings of the National Academy of Sciences* 117.32 (2020), pp. 18962–18968.

[143] Hao Zhang, Jean-Philippe Colombier, and Stefan Witte. "Laser-induced periodic surface structures: Arbitrary angles of incidence and polarization states". In: *Physical Review B* 101.24 (2020), p. 245430.

[144] Hao Zhang et al. "Coherence in ultrafast laser-induced periodic surface structures". In: *Physical Review B* 92.17 (2015), p. 174109.

[145] Colby L Zhao. "Solving Allen-Cahn and Cahn-Hilliard Equations using the Adaptive Physics Informed Neural Networks". In: *Communications in Computational Physics* 29.3 (2020).

[146] Vassilia Zorba et al. "Biomimetic artificial surfaces quantitatively reproduce the water repellency of a lotus leaf". In: *Advanced materials* 20.21 (2008), pp. 4049–4054.