Physics-Informed Parameter Identification: From Splines to Kolmogorov-Arnold Networks

René Schenkendorf
Harz University of Applied Sciences
Wernigerode, Germany
Email: rschenkendorf@hs-harz.de

***Abstract*—Mathematical models are crucial for advanced process control and digital twins. Model parameter identification, however, often involves a trade-off between statistical rigor and computational tractability. Nonlinear Least Squares (NLS) is the benchmark, enforcing physical laws as hard constraints. By contrast, relaxation methods, such as Scientific Machine Learning (SciML), offer flexibility but introduce soft constraints that can obscure parameter identifiability, particularly in dense data regimes where hard constraints typically yield high-precision estimates. In this work, we systematize these perspectives within a unified generalized Tikhonov regularization framework and clarify the structural link between Iterative Principal Differential Analysis (iPDA) and Physics-Informed Kolmogorov-Arnold Networks (PIKANs). Benchmarking five strategies on nonlinear sloshing dynamics reveals a nuanced reality: under dense sampling, NLS achieves superior accuracy, whereas relaxation methods introduce systematic bias in weakly identifiable parameters. However, in the challenging regime of sparse sampling combined with high noise, NLS can converge to poor local minima, while PIKANs demonstrate crucial robustness. These findings caution against uncritical adoption of SciML for well-posed inverse problems while highlighting their essential value under data scarcity, advocating for problem-specific strategy selection.**

***Index Terms*—Parameter Identification; Principal Differential Analysis; Splines; Kolmogorov-Arnold Networks; Scientific Machine Learning**

I. Introduction

In recent decades, the systematic use of mathematical models has attracted considerable attention across industries, from chemical reaction engineering to high-speed robotics. These models are crucial for advanced process control, optimization, and digital twins. For instance, in the process industries, precise dynamic models are prerequisites for Model Predictive Control (MPC) [1] and Observer Design [2], enabling optimal operation even when hardware sensors are limited. Rigorous parameter identification is essential in these applications, necessitating efficient and robust inverse problem solvers. While the model structure is typically dictated by physical laws, specific parameters, e.g., reaction rates or damping coefficients, are rarely known *a priori* and must be inferred from noisy experimental data.

A representative example is liquid sloshing dynamics, which poses a persistent challenge across scales, ranging from microliter dispensing in laboratory automation to large-scale fuel transport in vehicles and ships [3]. Accurate sloshing models are essential for vibration suppression, spillage prevention, and trajectory optimization in high-precision positioning systems.

The classical hard-constraint route, nonlinear least squares with repeated ODE integration at every optimizer step, can be prohibitive for stiff or high-dimensional models and sensitive to poor initial guesses. To avoid this repeated-integration burden, Principal Differential Analysis (PDA) [4] and its iterative variant (iPDA) [5] relax the hard constraint by using basis functions (typically B-splines) to approximate the state trajectory, transforming the dynamic optimization into an algebraic regression.

Recent advances in Scientific Machine Learning (SciML) [6] have revisited this relaxation philosophy. Physics-Informed Neural Networks (PINNs) [7] replace the spline basis with deep neural networks and have been applied to parameter identification in structural dynamics [8], although they often struggle with spectral bias [9]. The recently proposed Physics-Informed Kolmogorov-Arnold Networks (PIKANs) [10] utilize learnable activation functions on edges. To the best of our knowledge, the structural relationship of PIKANs to classical spline-based identification has not yet been made explicit, which motivates the present work.

The contribution of this work is twofold: First, we systematize measurement-based and physics-based identification approaches within a common variational landscape (Section II). Besides NLS, we interpret PDA/iPDA and SciML methods not as fundamentally distinct algorithms, but as varying degrees of constraint relaxation within a generalized Tikhonov framework. Second, we benchmark five strategies (NLS, PDA, iPDA, PINN, PIKAN) on nonlinear oscillatory dynamics. The results serve as a caution against the uncritical adoption of SciML: we highlight that "soft" physics enforcement can inadvertently mask ill-posedness, leading to biased parameter estimates, i.e., a risk less prevalent in hard-constraint solvers.

The remainder of the work is structured as follows: Section II introduces the unified theoretical framework. Section III describes the slosh dynamics case study. Section IV presents the quantitative benchmark results, followed by a critical discussion in Section V and conclusions in Section VI.

II. Methodology: Computational Approaches to the Inverse Problem

We consider the inverse problem for a time-invariant dynamic system $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t); \boldsymbol{p})$ with noisy discrete measurements $\mathbf{y}_i = \mathbf{h}(\mathbf{x}(t_i)) + \boldsymbol{\epsilon}_i$. The objective is to estimate the model parameters $\boldsymbol{p}$ given the data $\{\mathbf{y}_i, t_i\}$.

A. The Benchmark: Nonlinear Least Squares (NLS)

$$\hat{\boldsymbol{p}}_{\mathrm{NLS}} = \arg\min_{\boldsymbol{p}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{h}(\mathbf{x}(t_i;\boldsymbol{p}))\|^2, \quad \text{s.t. } \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u};\boldsymbol{p}). \tag{1}$$

This yields the maximum-likelihood estimator under Gaussian noise and serves as the reference standard for accuracy. We use single-shooting NLS as the prototypical hard-constraint baseline; related hard-constraint variants such as multiple shooting and orthogonal collocation share the same fidelity guarantee but differ in conditioning. From a computational perspective, NLS requires solving the Ordinary Differential Equation (ODE) initial value problem at every iteration of the optimization loop, which scales poorly with stiffness and dimensionality ($O(N_{\text{steps}})$ per gradient evaluation).

B. Unified Variational Framework for Relaxation Methods

To mitigate the sensitivities of shooting methods (such as NLS) to numerical conditioning, initial guesses, and process noise or model mis-specification, alternative methods *relax* the hard physics constraint. Instead of enforcing $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u};\boldsymbol{p})$ exactly, they introduce a trajectory approximator $\hat{\mathbf{x}}_\alpha$ (parameterized by coefficients α in a space $\mathcal{A}$). We define the joint generalized Tikhonov regularization functional [11] as follows:

$$(\hat{\boldsymbol{p}}, \hat{\alpha}) = \arg\min_{\boldsymbol{p},\alpha\in\mathcal{A}} \Bigg[\underbrace{\sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{h}(\hat{\mathbf{x}}_\alpha(t_i))\|^2}_{\mathcal{L}_{\text{data}}} + \lambda \underbrace{\int_0^T \|\dot{\hat{\mathbf{x}}}_\alpha(t) - \mathbf{f}(\hat{\mathbf{x}}_\alpha(t),\mathbf{u}(t);\boldsymbol{p})\|^2 dt}_{\mathcal{L}_{\text{physics}}} \Bigg]. \tag{2}$$

Here, λ is the regularization parameter that balances data fidelity ($\mathcal{L}_{\text{data}}$) and physical consistency ($\mathcal{L}_{\text{physics}}$). Table I classifies the relaxation strategies (PDA/iPDA and SciML) as instances of this framework.

TABLE I
CLASSIFICATION OF RELAXATION METHODS. INSTANCES OF EQ. (2).

Method	Approx. $\mathcal{A}$	Constraint λ	Opt. Strategy
PDA	B-Splines	Decoupled	Sequential (two-step)
iPDA	B-Splines	Finite $\lambda > 0$ (Soft)	Iterative (multi-step)
PINN	Deep NN	Finite $\lambda > 0$ (Soft)	Joint gradient descent
PIKAN	K-A Net	Finite $\lambda > 0$ (Soft)	Joint gradient descent

C. Spline-Based Relaxation: PDA and iPDA

Principal Differential Analysis (PDA) defines the approximation space $\mathcal{A}$ of Eq. (2) as a finite-dimensional B-spline space $\mathcal{S}_{d,\Xi}$, i.e., piecewise polynomials of degree d over a knot sequence $\Xi = \{\xi_1, \ldots, \xi_K\}$. In classical PDA, the optimization is decoupled: coefficients α are first fitted to the data (ignoring physics, i.e., minimizing only $\mathcal{L}_{\text{data}}$), and then $\boldsymbol{p}$ is estimated by minimizing $\mathcal{L}_{\text{physics}}$ given the fixed smoothing spline-based surrogate. iPDA implements the fully coupled minimization of Eq. (2), gradually enforcing the physics constraint ($\lambda > 0$) to align the spline derivative with the vector field $\mathbf{f}(\cdot)$.

D. Deep Learning Relaxation: SciML

Scientific Machine Learning (SciML) generalizes the iPDA framework by replacing the linear spline basis with universal nonlinear function approximators.

Physics-Informed Neural Networks (PINN) choose $\mathcal{A}$ of Eq. (2) as the space of deep neural networks, i.e., Multi-Layer Perceptrons (MLPs). The training process provides a direct stochastic gradient descent (SGD) approximation to the joint minimization of Eq. (2). While effective, MLPs are prone to spectral bias [9], which hinders efficient capture of high-frequency dynamics, and are known to be sensitive to architectural choices and loss-balancing schemes [12].

Physics-Informed Kolmogorov-Arnold Networks (PIKAN) [10] propose an alternative framework based on the Kolmogorov-Arnold representation theorem, placing learnable activation functions on edges. By parameterizing these activations as B-splines, the activation function reads as:

$$\varphi(z) = w_b \cdot \mathrm{SiLU}(z) + w_s \cdot \sum_{k=1}^{K} c_k B_k(z). \tag{3}$$

Thus, PIKANs effectively bridge the gap between classical spline methods (iPDA) and deep learning as discussed below.

E. Structural Equivalence: PIKAN and iPDA

The key insight is that both methods solve the same optimization problem when the network architecture reduces to a spline basis.

a) Core Observation.: Consider a shallow PIKAN where the output is a weighted sum of B-spline basis functions:

$$\hat{x}(t) = \sum_{k=1}^{K} c_k B_k(t), \tag{4}$$

with learnable coefficients $\mathbf{c} = [c_1, \ldots, c_K]^\top$. This is precisely the concept used in iPDA [5]. When we discretize the unified loss function (Eq. (2)) at measurement times $\{t_i\}$ and collocation points $\{\tau_m\}$, we obtain:

$$\mathcal{L}(\mathbf{c},\boldsymbol{p}) = \underbrace{\|\mathbf{y} - \mathbf{B}\mathbf{c}\|^2}_{\text{Data fit}} + \lambda \underbrace{\|\dot{\mathbf{B}}\mathbf{c} - \mathbf{F}(\mathbf{B}_\tau\mathbf{c},\boldsymbol{p})\|^2}_{\text{Physics residual}}, \tag{5}$$

where $\mathbf{B}_{ij} = B_j(t_i)$ is the spline basis matrix evaluated at measurement times, $\dot{\mathbf{B}}_{mj} = \frac{d}{dt}B_j(\tau_m)$ contains the basis derivatives at collocation points, and $\mathbf{F}$ stacks the ODE right-hand side evaluations [5].

b) Equivalence Result.: Under matched discretization (i.e., same basis functions and collocation grid), training a shallow spline PIKAN is mathematically equivalent to solving the iPDA coefficient optimization problem. The proof follows directly from the uniqueness of the B-spline representation: since $\hat{x}(t)$ in Eq. (4) uniquely determines $\mathbf{c}$, evaluating the PIKAN loss at the discrete points yields exactly Eq. (5), i.e., the standard iPDA objective. This exact equivalence holds for shallow

architectures where the network function space coincides with the spline space $\mathcal{S}_{d,\Xi}$, i.e., when the network reduces to $\hat{x}(t) = \sum_{k=1}^{K} c_k B_k(t)$ without additional nonlinear activation components ($w_b = 0$ in Eq. (3)). Deeper architectures introduce compositional expressivity beyond this space. However, the training objective retains the same penalized least-squares structure, and local optimization steps can be interpreted as iterative linearization around the current trajectory estimate.

This insight might be helpful for practitioners, as it implies that hyperparameter choices (e.g., number of knots, spline degree) in PIKANs can follow established guidelines from the functional data literature [13].

c) Information-Theoretic Perspective.: An additional aspect in comparing these methods is the ratio of available information (data points N) to the number of parameters being estimated. While NLS estimates only the physical parameters $\boldsymbol{p}$, relaxation methods jointly estimate trajectory approximators with substantially more degrees of freedom: iPDA uses spline coefficients (see Table II), PINNs use multi-layer weights, and PIKANs additionally use parameterized edge functions. The number of auxiliary parameters thus typically exceeds the physical parameter count by orders of magnitude.

In general, the Cramér-Rao bound sets fundamental limits on parameter estimation: the covariance of unbiased parameter estimators (including model parameters, spline coefficients or network weights) satisfies $\mathrm{Cov}(\hat{\boldsymbol{p}}, \hat{\alpha}) \geq \mathbf{F}^{-1}$, where $\mathbf{F}$ is the Fisher information matrix; this mirrors maximum-likelihood PDA estimation, where physical parameters and spline coefficients form one extended vector [14]. Critically, this bound depends not only on the overall parameter count but also on parameter sensitivities, i.e., the Fisher information specific to each parameter. In neural network contexts, when jointly optimizing physical parameters $\boldsymbol{p}$ and network weights $\mathbf{w}$, the effective Fisher information available for $\boldsymbol{p}$ is distributed across a much larger-dimensional space. This creates an information partitioning effect: although the total information in the data remains fixed, spreading it across exponentially more degrees of freedom raises the parameter uncertainties in practice.

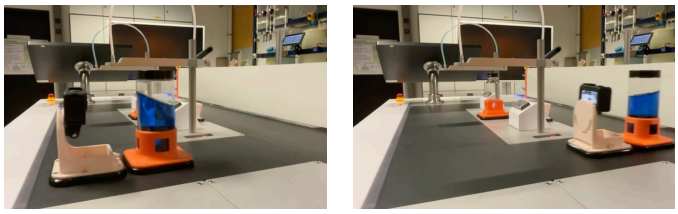
More fundamentally, recent theoretical analysis reveals that PINNs suffer from an implicit bias [15]: when minimizing the physics residual $\mathcal{L}_{\text{physics}}$, the network implicitly optimizes not the original ODE but a modified equation whose solution systematically deviates from the true parameters [15]. This bias is not merely a regularization side-effect but a structural property of the joint optimization landscape. The gradient flow $\frac{\partial \mathcal{L}}{\partial \mathbf{w}}$ provides far more degrees of freedom than $\frac{\partial \mathcal{L}}{\partial \boldsymbol{p}}$, allowing the optimizer to minimize data loss by adjusting trajectory approximations while leaving physical parameters in suboptimal states determined by the modified equation's intrinsic bias.

This implicit bias is further compounded by spectral bias in neural networks [16]: PINNs preferentially fit low-frequency solution components regardless of network architecture, potentially causing systematic parameter bias in regions of high spectral content. Together, these effects (i.e., information partitioning, implicit bias from modified equations, and spectral bias) provide a theoretical basis for the observed parameter deviations in soft-constraint methods. Thus, SciML methods for inverse problems need careful handling in practice, especially in cases where classical formulations like NLS are well-posed.

III. Case Study: System Identification of Liquid Sloshing

We evaluate the methodologies on a synthetic benchmark, i.e., the true model parameters are known. Note that the planar motor liquid sloshing itself was originally parameterized to the real asset. This system serves as a challenging, high-precision testbed for oscillation dynamics, where accurate model parameters are crucial for trajectory planning and control strategies that effectively suppress sloshing [17]–[19].

The synthetic benchmark mimics a physical experimental setup shown in Fig. 1, involving a magnetic levitation planar motor system [20] carrying a cylindrical tank (radius $R = 33\,\text{mm}$, fill level $h = 90\,\text{mm}$) filled with dyed water. Such planar motor systems offer contactless, multi-degree-of-freedom positioning with high dynamics and precision, making them attractive for high-throughput automation and material handling. However, their rapid acceleration capabilities can induce severe sloshing, necessitating accurate dynamic models for trajectory planning and vibration suppression. In the physical reference system, surface deflections are captured via high-speed camera tracking, which motivates the measurement model in this study.



(a) Setup t_1 (b) Setup t_2

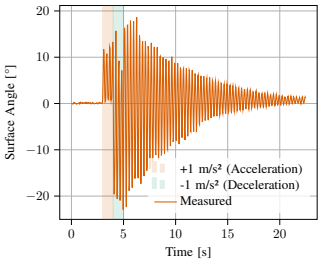
Fig. 1. **Experimental Setup.** The system consists of a liquid-filled tank on a magnetic levitation shuttle. A second shuttle carries a high-speed camera to track the fluid surface. (a) and (b) show the system at different time points during trajectory tracking.

To extract the surface angle, a robust vision pipeline isolates the tank via a fixed Region of Interest (ROI) (Fig. 2a). The dyed fluid is segmented using color-based thresholding, and the surface angle $\theta(t)$ is estimated via linear regression (Fig. 2c,d), providing a time-continuous validation signal (Fig. 2b).

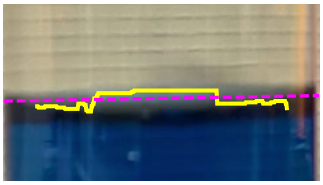
The free-fluid surface is approximated by a mechanical pendulum analogy [3] (Fig. 3), a valid simplification when the fundamental sloshing mode dominates and the fluid depth is sufficient to avoid shallow-water nonlinearities. This lower-order model assumes the liquid mass is split into a rigid part (fixed to the container) and an oscillating part (the pendulum) of mass m. The nonlinear equations of motion for the deflection angle θ (along the x-axis), subject to the container's translational accelerations $\mathbf{u} = [u_x, u_z]^\top$ and gravitational acceleration g, are:

$$\ddot{\theta} = \frac{u_x}{l_\theta}\cos\theta - \frac{g - u_z}{l_\theta}\sin\theta - \frac{d_\theta}{m}\dot{\theta}\cos^2\theta. \tag{6}$$

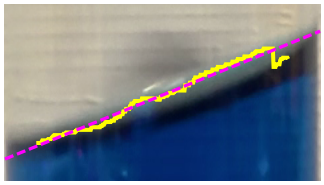
(a) ROI Selection



(b) Angle Time Series



(c) Snapshot t_1



(d) Snapshot t_2

Fig. 2. **Video-based Angle Detection.** (a) Full frame with ROI (red box). (b) Resulting angle trajectory $\theta(t)$. (c, d) Segmentation examples (yellow points) and robust fit (magenta line) from the sequence.

Note that the coefficient d_θ is used here as a lumped damping parameter mapped into the angular coordinate, so that the term $(d_\theta/m)\dot{\theta}$ has units consistent with $\ddot{\theta}$. If d_θ is interpreted in a translational viscous damper model, it represents an effective coefficient after coordinate transformation. Please also note that only the x-direction is considered in this study, reducing the problem to a 1D-system that could be extended to a 2D-system by including the y-direction, as shown in [3].

The system parameters have distinct physical interpretations: l_θ determines the natural frequency $\omega_n \approx \sqrt{g/l_\theta}$, while d_θ governs viscous damping and energy dissipation [3]. Identifying d_θ is particularly challenging because it primarily affects the transient decay envelope, which provides only a brief observation window relative to the dominant oscillation frequency. In this study, we apply a specific excitation profile $u_x(t)$, though in practice, multi-axis excitation profiles (rotate and translate in X, Y, and Z with the planar motor shuttle [20]) generated via optimal trajectory planning [21], [22] could be leveraged for Design of Experiment (DoE) strategies to maximize parameter sensitivity.

To validate the model's fidelity, Fig. 4 shows a high-resolution comparison of the measured angle response with the simulation. The close agreement during the acceleration ($t = 3 - 4\,\mathrm{s}$) and deceleration ($t = 4 - 5\,\mathrm{s}$) phases, as well as the subsequent free oscillation, confirms that the chosen pendulum analogy adequately captures the relevant dynamics for this control-oriented application.

IV. Simulation Results

We evaluate the five methods (NLS, PDA, iPDA, PINN, and PIKAN) on synthetic data generated from the planar motor model. The true system parameters are $l_\theta = 0.045\,\mathrm{m}$, $d_\theta = 0.15\,\mathrm{Ns/m}$, and $m = 0.30\,\mathrm{kg}$. Three noise scenarios are considered: Clean (machine precision), Low Noise ($\sigma = 0.1°$), and High Noise ($\sigma = 1.0°$). The goal is to recover the physical

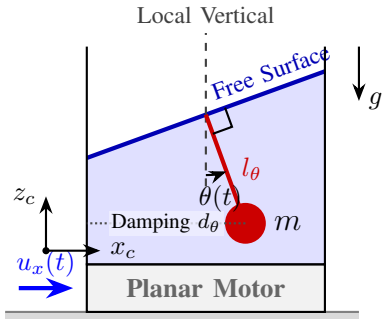


Fig. 3. **Mechanical Analogy of Liquid Sloshing.** The model approximates the first sloshing mode via a pendulum (l_θ, m) attached to the moving planar motor shuttle $\{x_c, z_c\}$. The deflection θ mirrors the surface tilt.

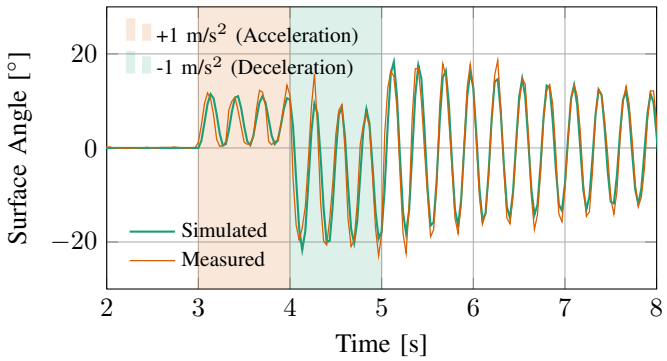


Fig. 4. **Model-Data-Fit.** Comparison of simulated and measured surface angles in the 1–8 s interval, highlighting the signal tracking capabilities.

parameters (l_θ, d_θ) from angle measurements, with the mass m fixed across all strategies. The PIKAN architecture uses a shallow network with cubic B-spline activations.

A. Experimental Protocol and Reproducibility

For comparability and reproducibility, Table II reports all benchmark settings; λ-schedules differ across methods and are doubled for the sparse-sampling scenario. Hyperparameters (knots, λ, optimizer iterations) follow functional-data guidelines [13] and standard SciML defaults rather than per-method grid search, so the reported numbers are representative rather than fully tuned. Results use fixed random seeds; all experiments ran on an Apple M1 CPU (16 GB RAM, Python 3.12).

B. Quantitative Comparison

Table III summarizes parameter identification performance under dense sampling ($N = 501$). The results confirm that NLS achieves near-optimal accuracy for both parameters across all noise levels, with the length parameter l_θ recovered within 0.2% of the true value. In contrast, the relaxation methods exhibit systematic deviations: PDA and iPDA overestimate l_θ by approximately 10–17%, while the neural approaches (PINN, PIKAN) recover l_θ well but systematically overestimate the damping parameter d_θ. Notably, PIKAN exhibits the largest damping bias (up to 0.320 vs. true 0.150), attributable to the information partitioning effect discussed in Section II-B. Regarding computational cost, PDA is fastest ($< 1\,\mathrm{ms}$),

TABLE II
IMPLEMENTATION DETAILS. ALL VALUES AND SETTINGS USED IN THE SIMULATION STUDY (FULL DATA SAMPLES).

Item	Value
Time horizon T and sampling interval Δt	$T = 5.0\,\text{s}$, $\Delta t = 0.01\,\text{s}$ (100 Hz)
Number of measurements N and collocation M	$N = 501$, $M = 501$ (PINN/PIKAN collocation)
Input profiles $u_x(t), u_y(t), u_z(t)$	Piecewise constant; $u_x = 1.0$ for $t \in [0.2, 0.6)$, $u_x = -1.0$ for $t \in [1.2, 1.6)$, else 0
Noise model and standard deviation σ	Additive i.i.d. Gaussian; Low: 0.1°; High: 1.0°; plus Clean
Mass parameter m	Fixed to $m = 0.30$ in all strategies
NLS	Multi-start (best-of-5); init $p_0 = [0.06, 0.05]$ with $\pm 20\%$ perturbation; Trust Region Reflective solver
PDA / iPDA	Cubic splines ($d = 3$); adaptive knots (~ 80); $\lambda = 10^{-4}$; warmup 15 iters
PINN	Adam (5000 iter, λ_{phys}: 0→0.1) + L-BFGS (500 steps, $\lambda = 1$); CPU
PIKAN	Grid refinement (10→20→40); λ-schedule (0→0.1); L-BFGS (500 steps, $\lambda = 1$); CPU

TABLE III
BENCHMARK RESULTS. ESTIMATED PARAMETERS ($\hat{l}_\theta, \hat{d}_\theta$) AND TIME. TRUE: $l_\theta = 0.045\,\text{M}$, $d_\theta = 0.15\,\text{NS/M}$.

Metric	NLS	PDA	iPDA	PINN	PIKAN
Scenario: Clean					
$\hat{l}_\theta$ [m]	0.0450	0.0530	0.0496	0.0451	0.0449
$\hat{d}_\theta$ [Ns/m]	0.150	0.144	0.173	0.193	0.289
Time [s]	1.76	$< \mathbf{10^{-3}}$	0.56	76	1521
Scenario: Low Noise					
$\hat{l}_\theta$ [m]	0.0450	0.0530	0.0496	0.0451	0.0448
$\hat{d}_\theta$ [Ns/m]	0.152	0.150	0.178	0.201	0.320
Time [s]	1.12	$< \mathbf{10^{-3}}$	0.54	55	1400
Scenario: High Noise					
$\hat{l}_\theta$ [m]	0.0449	0.0529	0.0498	0.0448	0.0451
$\hat{d}_\theta$ [Ns/m]	0.160	0.137	0.163	0.168	0.274
Time [s]	1.09	$< \mathbf{10^{-3}}$	0.50	40	1619

TABLE IV
BENCHMARK RESULTS (10% DATA). COMPARISON UNDER DATA SCARCITY ($N = 51$). NOTE NLS BREAKDOWN VS PIKAN ROBUSTNESS.

Metric	NLS	PDA	iPDA	PINN	PIKAN
Scenario: Clean					
$\hat{l}_\theta$ [m]	0.0450	0.0532	0.0547	0.0438	0.0449
$\hat{d}_\theta$ [Ns/m]	0.150	0.847	0.377	0.256	0.413
Time [s]	0.86	$< \mathbf{10^{-3}}$	0.06	20	1651
Scenario: Low Noise					
$\hat{l}_\theta$ [m]	0.0450	0.0535	0.0550	0.0442	0.0452
$\hat{d}_\theta$ [Ns/m]	0.149	0.874	0.382	0.292	0.235
Time [s]	1.36	$< \mathbf{10^{-3}}$	0.06	20	1672
Scenario: High Noise					
$\hat{l}_\theta$ [m]	**0.0428**	0.0595	0.0569	0.0439	**0.0451**
$\hat{d}_\theta$ [Ns/m]	**1.145**	0.691	0.393	0.468	**0.151**
Time [s]	0.80	$< \mathbf{10^{-3}}$	0.07	23	1272

followed by NLS ($\sim$1 s), while PIKAN requires >20 minutes due to its iterative grid refinement procedure.

C. Robustness Analysis: Data Scarcity (10%)

To demonstrate the limits of the proposed framework, we introduce a more challenging scenario with data scarcity. The dataset is downsampled to 10% of the original snapshots ($N = 51$), retaining the full simulation horizon ($T = 5\,\text{s}$) but with a sparse sampling interval of $\Delta t = 0.1$.

The results in Table IV reveal a critical issue. Under high noise, the classical NLS solver fails catastrophically ($d_\theta \approx 1.15$ vs. 0.15), as sparse, noisy gradients trap the solver in a local minimum. In contrast, PIKAN recovers $d_\theta = 0.151$ in this scenario. Interestingly, PIKAN's damping estimate even improves from clean (0.413) to high-noise (0.151) under sparse sampling; we interpret this non-monotonic behavior cautiously as a combined effect of spectral bias and noise acting as stochastic regularization that disrupts the biased attractor. A repeated-seed study is needed to separate this from a single-seed coincidence, so PIKAN's sparse/high-noise gain should be read as indicative rather than guaranteed. The learning-based methods (PINN, PIKAN) identify l_θ robustly (≈ 0.045) across all test cases, but systematically overestimate d_θ in this sparse, low-damping regime, indicating that the regularization balance or implicit spectral bias struggles to uncouple the weak damping signal from the dominant oscillatory mode without denser data or damping-specific priors. PIKAN's runtime is significantly higher due to CPU-bound execution, motivating GPU-accelerated implementations. Fig. 5 shows the simulated trajectories for the downsampled, high-noise scenario. The SciML-based methods (PINN, PIKAN) closely follow the reference trajectory; differences are most apparent in the damping envelope during free oscillation, where PIKAN demonstrates superior tracking accuracy compared to PINNs in this challenging training regime.

V. DISCUSSION

The benchmark revealed complementary strengths across methods. Under dense sampling (Table III), NLS achieves near-perfect recovery because the ODE hard constraint ensures high parameter sensitivity (implying a full-rank Fisher information matrix $\mathbf{F}$) and, consequently, a typically well-posed identification problem (local convexity). The relaxation methods, however, systematically overestimate the weakly identifiable damping parameter d_θ (e.g., PIKAN: 0.289 vs. true: 0.150), as the flexible trajectory approximator can absorb residuals that would otherwise constrain d_θ.

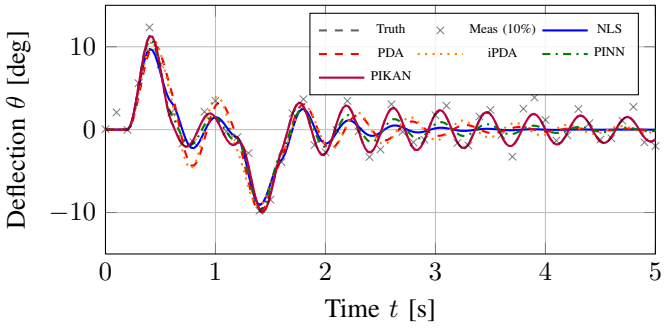


Fig. 5. Trajectory reconstruction (10% data, high noise). NLS yields an over-damped response; PIKAN closely tracks the true dynamics.

The situation reverses specifically under sparse sampling with high noise (Table IV): NLS becomes trapped in local minima ($\hat{d}_\theta = 1.145$), while PIKAN's implicit regularization enables robust recovery ($\hat{d}_\theta = 0.151$). This aligns with established principles in inverse problems [11], [23]: hard constraints maximize identifiability when data is sufficient, but regularization becomes crucial when data is sparse and noisy. For PINNs, this effect is amplified by spectral bias [16], which favors low-frequency approximations. For PIKANs, a similar smoothing effect arises from the B-spline basis $\mathcal{S}_{d,\Xi}$: while local support is maintained, approximation capacity is limited by knot density. Understanding these method-specific biases is essential for selecting appropriate identification strategies in practice.

VI. Conclusions

This work systematically clarifies the relationship between classical spline-based methods and Kolmogorov-Arnold Networks for system identification by demonstrating that PIKANs are a neural realization of the established iPDA methodology. This insight is crucial for interpreting SciML results: the 'learning' in a shallow PIKAN is mathematically analogous to iterative spline smoothing. Empirically, the benchmark results underscore that modern deep learning methods are not a panacea. For well-posed, low-dimensional inverse problems with sufficient data, classical NLS solvers remain superior in both accuracy and efficiency. However, under sparse sampling and high noise, PIKANs demonstrate crucial robustness where hard-constraint solvers may fail. The true potential of PIKANs lies not in replacing NLS for simple tasks but as modular, interpretable building blocks for complex hybrid physics-AI models where classical integration is infeasible. This study serves as both a theoretical bridge and a practical reality check, advocating for problem-specific selection of identification strategies. Future research will target higher-dimensional parameter spaces, real vision-based measurements, repeated-seed statistical analyses, and extensions from ODEs to PDE systems with adaptive regularization.

Acknowledgment

Claude Opus 4.6 (Anthropic AI) was used for language polishing. The authors reviewed all content and take full responsibility for this publication.

References

[1] E. F. Camacho and C. Bordons, *Model Predictive Control*, 2nd ed., ser. Advanced Textbooks in Control and Signal Processing. Springer London, 2007.

[2] L. Fortuna, S. Graziani, A. Rizzo, and M. G. Xibilia, *Soft Sensors for Monitoring and Control of Industrial Processes*, ser. Advances in Industrial Control. London: Springer Science & Business Media, 2007.

[3] R. A. Ibrahim, *Liquid Sloshing Dynamics: Theory and Applications*. Cambridge: Cambridge University Press, 2009.

[4] J. O. Ramsay and B. W. Silverman, *Functional Data Analysis*, 2nd ed., ser. Springer Series in Statistics. New York, NY: Springer, 2005.

[5] A. A. Poyton, M. S. Varziri, K. B. McAuley, P. J. McLellan, and J. O. Ramsay, "Parameter estimation in continuous-time dynamic models using principal differential analysis," *Computers & Chemical Engineering*, vol. 30, no. 4, pp. 698–708, 2006.

[6] G. Kutyniok, P. Petersen, M. Raslan, and R. Schneider, "A theoretical analysis of deep neural networks and parametric pdes," *Constructive Approximation*, vol. 55, no. 1, pp. 73–125, 2022.

[7] M. Raissi, P. Perdikaris, and G. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[8] T. Kapoor, H. Wang, A. Núnez, and R. Dollevoet, "Physics-informed neural networks for solving forward and inverse problems in complex beam systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 5, pp. 5981–5995, 2024.

[9] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. A. Hamprecht, Y. Bengio, and A. Courville, "On the spectral bias of neural networks," 2019. [Online]. Available: https://arxiv.org/abs/1806.08734

[10] Z. Liu, Y. Wang, S. Vaidya, F. Ruehle, J. Halverson, M. Soljačić, T. Y. Hou, and M. Tegmark, "Kan: Kolmogorov-arnold networks," 2025. [Online]. Available: https://arxiv.org/abs/2404.19756

[11] A. Tikhonov and V. Arsenin, *Solutions of Ill-Posed Problems*. Washington, D.C.: V.H. Winston & Sons, 1977.

[12] S. Wang, Y. Teng, and P. Perdikaris, "Understanding and mitigating gradient flow pathologies in physics-informed neural networks," *SIAM Journal on Scientific Computing*, vol. 43, no. 5, pp. A3055–A3081, 2021.

[13] J. O. Ramsay, G. Hooker, D. Campbell, and J. Cao, "Parameter estimation for differential equations: A generalized smoothing approach," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 69, no. 5, pp. 741–796, 2007.

[14] M. S. Varziri, K. B. McAuley, and P. J. McLellan, "Parameter and state estimation in nonlinear stochastic continuous-time dynamic models with unknown disturbance intensity," *The Canadian Journal of Chemical Engineering*, vol. 86, no. 5, pp. 828–837, 2008.

[15] T. Luo and Q. Zhou, "On residual minimization for pdes: Failure of pinn, modified equation, and implicit bias," *arXiv preprint arXiv:2310.18201*, 2023.

[16] S. Wang and P. Perdikaris, "When and why pinns fail to train: A neural tangent kernel perspective," *Journal of Computational Physics*, vol. 430, p. 110027, 2021.

[17] L. Moriello, L. Biagiotti, C. Melchiorri, and A. Paoli, "Manipulating liquids with robots: A sloshing-free solution," *Control Engineering Practice*, vol. 78, pp. 129–141, 2018.

[18] H. Gattringer, A. Müller, S. Weitzhofer, and M. Schörgenhumer, "Point to point time optimal handling of unmounted rigid objects and liquid-filled containers," *Mechanism and Machine Theory*, vol. 184, p. 105286, 2023.

[19] W. Aribowo, T. Yamashita, K. Terashima, and H. Kitagawa, "Input shaping control to suppress sloshing on liquid container transfer using multi-joint robot arm," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 3489–3494.

[20] Planar Motor Inc., "Magnetic levitation planar motor technology," https://www.planarmotor.com/, 2024, accessed: 2024.

[21] J. Reinhold, M. Amersdorfer, and T. Meurer, "A dynamic optimization approach for sloshing free transport of liquid filled containers using an industrial robot," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 2336–2341.

[22] K. N. Viet, M. D. Duc, T. C. Duc, and T. L. Nguyen, "Anti-sloshing control: Flatness-based trajectory planning and tracking control with an integrated extended state observer," *IET Cyber-Systems and Robotics*, vol. 6, no. 3, p. e12121, 2024.

[23] R. C. Aster, B. Borchers, and C. H. Thurber, *Parameter Estimation and Inverse Problems*, 3rd ed. Amsterdam: Elsevier, 2018.