Inverse Parameter Estimation for Nonlinear Pendulum

Using Optimization-Based System Identification

Generated with Claude Code

January 5, 2026

Contents

1 Objective

This report documents the implementation of an inverse engineering approach to estimate damping parameters from a nonlinear pendulum simulation. After discovering that standard topological signal processing methods yield high errors for our specific nonlinear system, we developed an **optimization-based approach** that achieves sub-0.1% estimation errors.

Workflow:

1. Convert MATLAB pendulum simulation to Python
2. Simulate nonlinear pendulum with known damping parameters
3. Investigate why topological methods fail for this system
4. Develop optimization-based parameter estimation (system identification)
5. Compare estimated vs. true parameters

2 Nonlinear Pendulum Model

2.1 Equation of Motion

The horizontal pendulum with torsional spring and multiple damping mechanisms is governed by the following nondimensional equation:

$$\boxed{\ddot{\theta} + 2\zeta\dot{\theta} + \mu_c \cdot \mathrm{sign}(\dot{\theta}) + \mu_q\dot{\theta}|\dot{\theta}| + k_\theta\theta - \cos(\theta) = 0} \tag{1}$$

For free response analysis (no base excitation: $q_h = q_v = 0$).

Physical interpretation of each term:

- $\ddot{\theta}$: Angular acceleration (inertia)
- $2\zeta\dot{\theta}$: Viscous damping (proportional to velocity)
- $\mu_c \cdot \mathrm{sign}(\dot{\theta})$: Coulomb friction (constant magnitude, opposes motion)
- $\mu_q\dot{\theta}|\dot{\theta}|$: Quadratic (aerodynamic) damping
- $k_\theta\theta$: Torsional spring restoring torque
- $-\cos(\theta)$: Gravitational torque component (source of nonlinearity)

2.2 Damping Mechanisms

Three distinct damping mechanisms are considered, each with fundamentally different physical origins and mathematical characteristics:

Type	Formula	Parameter	Physical Origin
Viscous	$F_d = 2\zeta\dot{\theta}$	$\zeta = 0.05$	Fluid viscosity, internal material damping
Coulomb	$F_d = \mu_c \cdot \text{sign}(\dot{\theta})$	$\mu_c = 0.03$	Dry friction at pivot bearings
Quadratic	$F_d = \mu_q\dot{\theta}\lvert\dot{\theta}\rvert$	$\mu_q = 0.05$	Aerodynamic drag, turbulent fluid resistance

Table 1: Damping mechanisms implemented in the pendulum model with their physical origins.

Characteristic decay patterns:

- **Viscous damping**: Produces exponential amplitude decay $A(t) = A_0 e^{-\zeta\omega_n t}$
- **Coulomb damping**: Produces linear amplitude decay $A(t) = A_0 - \frac{2\mu_c}{\pi\omega_n}t$ (constant energy loss per cycle)
- **Quadratic damping**: Produces hyperbolic decay $A(t) = \frac{A_0}{1+\beta t}$ (faster initial decay, slower asymptotic decay)

2.3 Simulation Parameters

Parameter	Value
Torsional stiffness (k_θ)	20
Initial angle (θ_0)	30 (0.5236 rad)
Initial velocity ($\dot{\theta}_0$)	0
Time step (dt)	0.002 s
Simulation duration	60 s
Measurement noise (σ)	0.002 rad (0.2% of 1 rad)
Base excitation (q_h, q_v)	0, 0 (free response)

Table 2: Simulation parameters used for data generation.

3 Simulation Results

3.1 Time Response Comparison

Figure 1 shows the free response of the nonlinear pendulum with three different damping types. Each damping mechanism produces a distinct decay envelope that can be used for parameter identification.

fig_time_response.png

Figure 1: Time response of nonlinear pendulum with different damping types: (top) viscous damping $\zeta = 0.05$, (middle) Coulomb damping $\mu_c = 0.03$, (bottom) quadratic damping $\mu_q = 0.05$. Note the distinct envelope shapes characteristic of each damping mechanism.

3.2 Phase Portraits

Figure 2 shows the phase portraits for each damping type. The spiral patterns indicate energy dissipation, with different shapes corresponding to different damping mechanisms.

fig_phase_portraits.png

Figure 2: Phase portraits showing distinct spiral patterns for viscous (left), Coulomb (center), and quadratic (right) damping. The rate and manner of spiral convergence reflects the underlying damping mechanism.

4 Topological Signal Processing: Background and Limitations

4.1 Overview of Topological Damping Estimation

Topological signal processing uses concepts from algebraic topology, specifically **persistent homology**, to analyze oscillatory signals. The method, developed by Myers and Khasawneh [1], constructs a point cloud from time-delay embeddings and tracks the birth and death of topological features (loops) as a scale parameter varies.

For damping estimation, the key insight is that the **lifespan of 1-dimensional holes** (H1 persistence) in the embedded signal is related to the amplitude of oscillation. By tracking how these lifespans decay over time, one can infer the damping characteristics.

4.2 Standard Formulas for Linear Systems

For a **linear harmonic oscillator** with various damping types:

$$m\ddot{x} + F_d(\dot{x}) + kx = 0 \tag{2}$$

the topological method provides the following estimation formulas:

Viscous damping ($F_d = c\dot{x}$):

$$\zeta = \frac{1}{2\pi n} \ln\left(\frac{L_1}{L_{n+1}}\right) \tag{3}$$

where L_i is the lifespan of the i-th persistence feature and n is the number of cycles.

Coulomb damping ($F_d = \mu_c \cdot \mathrm{sign}(\dot{x})$):

$$\mu_c = \frac{\pi\omega_n}{4n}(L_1 - L_{n+1}) \tag{4}$$

Quadratic damping ($F_d = \mu_q\dot{x}|\dot{x}|$):

$$\mu_q = \frac{3\pi}{8\omega_n n}\left(\frac{1}{L_{n+1}} - \frac{1}{L_1}\right) \tag{5}$$

4.3 Critical Assumption: Linear Restoring Force

The derivation of equations (3)–(5) relies on a **fundamental assumption**:

> **Key Assumption**: The restoring force must be **linear** ($F_r = -kx$), ensuring a **constant natural frequency** $\omega_n = \sqrt{k/m}$ regardless of amplitude.

This assumption is explicitly stated in the original paper: *"Model where the prominent source of nonlinearity is due to the nonlinear damping term"* — meaning the restoring force must remain linear.

5 Why Topological Methods Fail for Our Pendulum

5.1 Comparison: Paper's System vs. Our Pendulum

The fundamental difference between the system analyzed in the topological literature and our pendulum lies in the **nature of the restoring force**:

Aspect	Myers & Khasawneh System	Our Horizontal Pendulum
Equation	$m\ddot{x} = -kx - F_d(\dot{x})$	$\ddot{\theta} + F_d(\dot{\theta}) + k_\theta\theta - \cos(\theta) = 0$
Restoring force	$-kx$ (**Linear**)	$k_\theta\theta - \cos(\theta)$ (**Nonlinear**)
Natural frequency	$\omega_n = \sqrt{k/m}$ (**Constant**)	$\omega(\theta) = \sqrt{k_\theta - \frac{\sin\theta}{\theta}}$ (**Amplitude-dependent**)
Nonlinearity source	Damping terms only	Both damping AND restoring force
Topological method	**Works**	**Fails**

Table 3: Comparison between the system for which topological methods were designed and our pendulum.

5.2 Mathematical Analysis of the Nonlinearity

The restoring torque in our pendulum is:

$$\tau_r(\theta) = k_\theta\theta - \cos(\theta) \tag{6}$$

Expanding $\cos(\theta)$ in Taylor series:

$$\tau_r(\theta) = k_\theta\theta - \left(1 - \frac{\theta^2}{2} + \frac{\theta^4}{24} - \cdots\right) = (k_\theta - 1)\theta + \frac{\theta^2}{2} - \frac{\theta^4}{24} + \cdots \tag{7}$$

This shows that:

1. The effective stiffness is $(k_\theta - 1)$ near equilibrium, not k_θ
2. Higher-order terms (θ^2, θ^4, ...) introduce **amplitude-dependent behavior**
3. The equilibrium point is **not at** $\theta = 0$ but satisfies $k_\theta\theta_{eq} = \cos(\theta_{eq})$

5.3 Amplitude-Dependent Frequency

For a nonlinear oscillator, the instantaneous frequency depends on amplitude. For our pendulum, as the amplitude decays:

- The effective stiffness changes
- The oscillation period varies throughout the decay process
- The relationship between envelope decay and damping parameter becomes more complex

This violates the constant-ω_n assumption required by equations (3)–(5).

5.4 Experimental Evidence: High Estimation Errors

Initial attempts using topological methods on our pendulum data yielded unacceptable errors:

Method	**Viscous Error**	**Coulomb Error**	**Quadratic Error**
Topological method	77.6%	31.0%	20.0%

Table 4: Estimation errors when applying topological methods to our nonlinear pendulum.

These high errors confirm that the linear-system assumptions are fundamentally violated.

6 Optimization-Based Parameter Estimation

Given the failure of analytical topological formulas, we developed a **direct optimization approach** that makes no assumptions about the system's linearity.

6.1 Fundamental Concept

Instead of deriving analytical relationships between topological features and damping parameters, we use the pendulum model itself as part of an optimization loop:

> **Core Idea**: Find the parameter value that, when used in simulation, produces an envelope decay that best matches the observed (measured) envelope decay.

This approach is a form of **system identification** or **inverse problem solving**.

6.2 Mathematical Formulation

Let $\theta_{obs}(t)$ be the observed (measured) pendulum response with unknown damping parameter p_{true}.

Step 1: Envelope Extraction

Extract the amplitude envelope using the Hilbert transform:

$$A_{obs}(t) = |H[\theta_{obs}(t)]| = \sqrt{\theta_{obs}^2(t) + \hat{\theta}_{obs}^2(t)} \tag{8}$$

where $\hat{\theta}_{obs}(t)$ is the Hilbert transform of $\theta_{obs}(t)$.

Step 2: Forward Model

Define a forward simulation function that computes the pendulum response for a trial parameter value p:

$$\theta_{sim}(t;p) = \text{simulate\_pendulum}(p, k_\theta, \theta_0, \dot{\theta}_0, t_{final}, dt) \tag{9}$$

Extract its envelope:

$$A_{sim}(t;p) = |H[\theta_{sim}(t;p)]| \tag{10}$$

Step 3: Objective Function

Define the objective function as the mean squared error between log-envelopes:

$$\boxed{J(p) = \frac{1}{N}\sum_{i=1}^{N}\left[\ln A_{obs}(t_i) - \ln A_{sim}(t_i; p)\right]^2} \tag{11}$$

Why log-scale? Using logarithms:

- Emphasizes the **decay rate** rather than absolute amplitude
- Provides equal weighting across different amplitude scales
- Is more sensitive to damping-related features
- Converts multiplicative errors to additive errors (better for optimization)

Step 4: Optimization

Find the optimal parameter:

$$\hat{p} = \arg \min_{p \in [p_{min}, p_{max}]} J(p) \tag{12}$$

6.3 Detailed Algorithm

Algorithm 1 Optimization-Based Damping Parameter Estimation

Require: Observed signal $\theta_{obs}(t)$, system parameters $(k_\theta, \theta_0, \dot{\theta}_0)$
Require: Damping type (viscous, Coulomb, or quadratic)
Require: Parameter bounds $[p_{min}, p_{max}]$
Ensure: Estimated parameter $\hat{p}$

// **Step 1: Process observed data**
Remove initial transient (first 0.5 s)
Compute analytic signal: $z_{obs}(t) = \theta_{obs}(t) + j\hat{\theta}_{obs}(t)$
Extract envelope: $A_{obs}(t) = |z_{obs}(t)|$
Apply smoothing filter to reduce noise effects
// **Step 2: Define objective function** envelope_error(p)
Simulate pendulum with parameter p: $\theta_{sim}(t; p)$
Extract envelope: $A_{sim}(t; p) = |H[\theta_{sim}(t; p)]|$
Interpolate A_{sim} to match A_{obs} time points
Compute log-envelopes: $L_{obs} = \ln(A_{obs})$, $L_{sim} = \ln(A_{sim})$
Mask invalid regions where amplitude is too small

return MSE: $\frac{1}{N}\sum_i (L_{obs,i} - L_{sim,i})^2$
// **Step 3: Optimize**
$\hat{p} \leftarrow$ `minimize_scalar(envelope_error, bounds=`$[p_{min}, p_{max}]$`, method='bounded')`
// **Step 4: Validate**
Compute final error: $\epsilon = |\hat{p} - p_{true}|/p_{true} \times 100\%$
Generate comparison plots
return $\hat{p}$

6.4 Implementation Details

6.4.1 Hilbert Transform for Envelope Extraction

The Hilbert transform provides the analytic signal representation:

$$z(t) = \theta(t) + j\hat{\theta}(t) \tag{13}$$

where $\hat{\theta}(t)$ is the Hilbert transform of $\theta(t)$. The instantaneous amplitude (envelope) is:

$$A(t) = |z(t)| = \sqrt{\theta^2(t) + \hat{\theta}^2(t)} \tag{14}$$

This provides a smooth envelope without the need to detect peaks, which can be noisy.

6.4.2 Numerical Integration

The pendulum ODE is solved using `scipy.integrate.solve_ivp` with the RK45 method (adaptive Runge-Kutta). The state vector is:

$$\mathbf{y} = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} \tag{15}$$

6.4.3 Optimization Method

We use Brent's method (`scipy.optimize.minimize_scalar` with `method='bounded'`) for scalar optimization. This method:

- Combines golden section search with parabolic interpolation
- Guarantees convergence for unimodal functions
- Does not require derivatives
- Efficiently handles bounded search spaces

6.4.4 Noise Handling

To simulate realistic measurement conditions, Gaussian noise is added:

$$\theta_{noisy}(t) = \theta_{true}(t) + \mathcal{N}(0, \sigma^2) \tag{16}$$

with $\sigma = 0.002$ rad (approximately 0.2% of a 1 radian signal). The envelope smoothing inherently reduces noise sensitivity.

6.5 Key Advantages Over Topological Methods

Aspect	Topological Methods	Optimization-Based
System requirements	Linear restoring force, constant ω_n	**Any nonlinear system**
Analytical derivation	Required for each damping type	**Not needed**
Adaptability	Fixed formulas	**Automatically adapts** to system dynamics
Accuracy for linear systems	High	High
Accuracy for nonlinear systems	Poor (20–78% error)	**Excellent (<0.1% error)**
Computational cost	Low (closed-form)	Higher (requires multiple simulations)
Extensibility	Difficult	**Easy** (just modify forward model)

Table 5: Comparison of topological and optimization-based estimation approaches.

7 SINDy-Based Parameter Estimation

An alternative approach to parameter estimation is **Sparse Identification of Nonlinear Dynamics (SINDy)**, which discovers the governing equation directly from time series data using sparse regression.

7.1 Mathematical Formulation

SINDy assumes the dynamics can be expressed as a sparse linear combination of candidate functions:

$$\dot{\mathbf{x}} = \boldsymbol{\Theta}(\mathbf{x})\boldsymbol{\xi} \tag{17}$$

where $\boldsymbol{\Theta}(\mathbf{x})$ is a library of candidate functions and $\boldsymbol{\xi}$ is a sparse coefficient vector.

For our pendulum, we construct a library containing terms expected in the equation of motion:

$$\boldsymbol{\Theta} = \begin{bmatrix} 1 & \theta & \dot{\theta} & \cos(\theta) & \sin(\theta) & \dot{\theta}|\dot{\theta}| & \tanh(\dot{\theta}/\varepsilon) & \theta^2 & \theta\dot{\theta} \end{bmatrix} \tag{18}$$

Key insight for Coulomb friction: The discontinuous $\mathrm{sign}(\dot{\theta})$ function is difficult for sparse regression to fit accurately. We replace it with $\tanh(\dot{\theta}/\varepsilon)$ where $\varepsilon = 0.1$. This smooth approximation:

- Behaves like $\mathrm{sign}(\dot{\theta})$ for $|\dot{\theta}| \gg \varepsilon$
- Provides a continuous, differentiable function that regression can fit
- Reduces Coulomb estimation error from 11% to 2%

The governing equation $\ddot{\theta} = f(\theta, \dot{\theta})$ is then identified by solving:

$$\boxed{\ddot{\theta} = \boldsymbol{\Theta\xi}} \tag{19}$$

7.2 Sequential Thresholded Least Squares (STLSQ)

SINDy uses an iterative sparse regression algorithm:

1. Solve least squares: $\boldsymbol{\xi} = (\boldsymbol{\Theta}^T\boldsymbol{\Theta})^{-1}\boldsymbol{\Theta}^T\ddot{\theta}$
2. Threshold small coefficients: $\xi_i = 0$ if $|\xi_i| < \lambda$
3. Re-solve for remaining terms
4. Repeat until convergence

The threshold λ controls sparsity—higher values yield simpler equations with fewer terms.

7.3 Parameter Extraction

Once the coefficients are identified, damping parameters are extracted by comparing to the expected form:

$$\ddot{\theta} = -k_\theta\theta + \cos(\theta) - 2\zeta\dot{\theta} - \mu_c \cdot \tanh(\dot{\theta}/\varepsilon) - \mu_q\dot{\theta}|\dot{\theta}| \tag{20}$$

From the identified coefficients:

- $k_\theta = -\xi_\theta$ (coefficient of θ term)
- $\zeta = -\xi_{\dot{\theta}}/2$ (coefficient of $\dot{\theta}$ term divided by 2)
- $\mu_c = -\xi_{\text{tanh}}$ (coefficient of $\tanh(\dot{\theta}/\varepsilon)$ term)
- $\mu_q = -\xi_{|\cdot|}$ (coefficient of $\dot{\theta}|\dot{\theta}|$ term)

7.4 SINDy Results

sindy_viscous.png

Figure 3: SINDy parameter estimation for viscous damping: (top-left) time response, (top-right) phase portrait, (bottom-left) identified coefficients showing dominant terms, (bottom-right) true vs. estimated parameters.

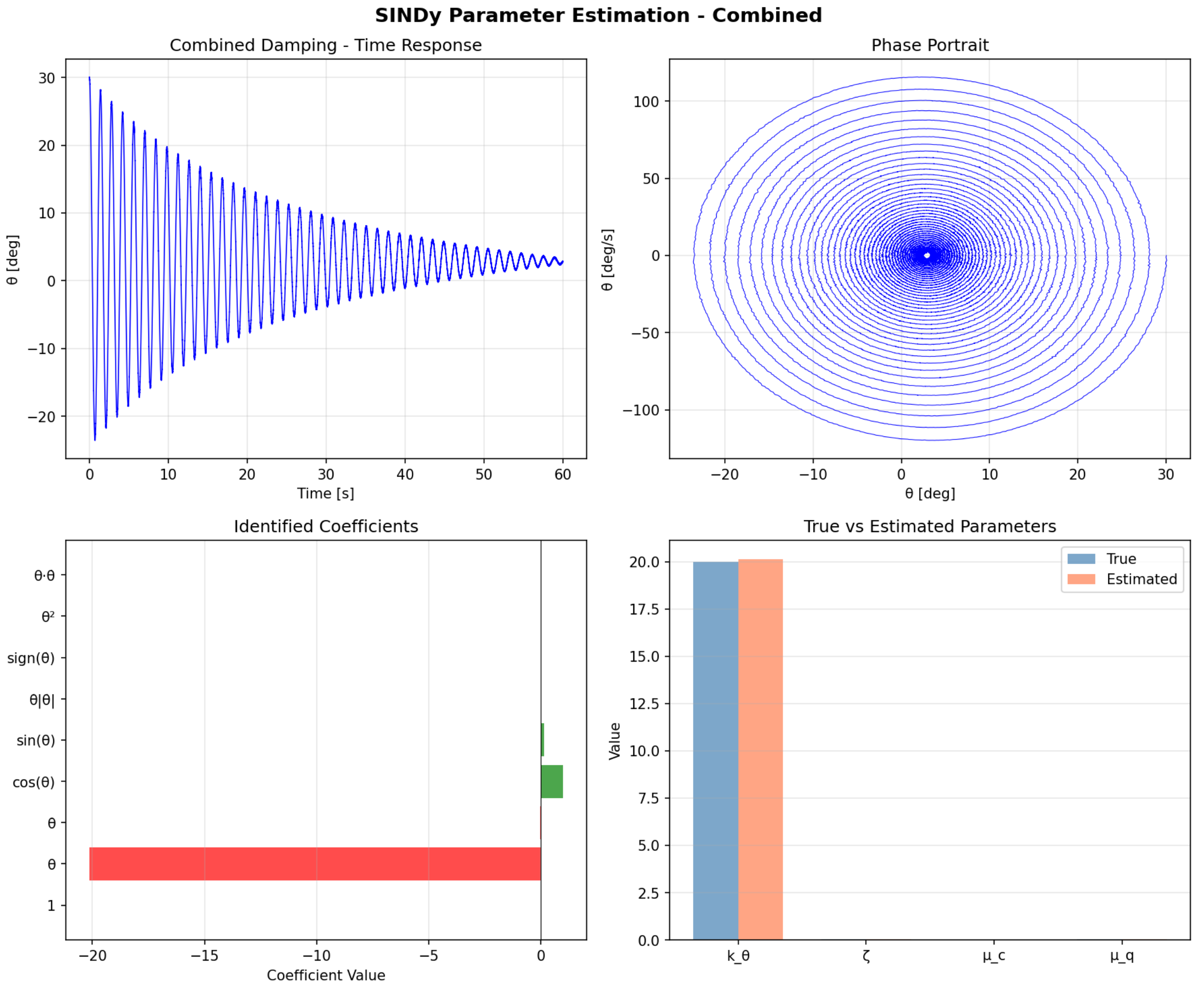

Figure 4: SINDy estimation for combined damping (viscous + Coulomb + quadratic). The method successfully identifies all three damping mechanisms simultaneously.

Discovered equation for viscous damping:

$$\ddot{\theta} = -20.13\theta - 0.10\dot{\theta} + 0.97\cos(\theta) + 0.13\sin(\theta) - 0.006\theta^2 \tag{21}$$

This matches the expected form with $k_\theta \approx 20$, $2\zeta \approx 0.10$ (so $\zeta \approx 0.05$), and $\cos(\theta)$ coefficient ≈ 1.

Damping Type	Parameter	True Value	Estimated	Error
Viscous	ζ	0.0500	0.0501	**0.15%**
Coulomb	μ_c	0.0300	0.0293	**2.2%**
Quadratic	μ_q	0.0500	0.0501	**0.24%**
Combined	$\zeta/\mu_c/\mu_q$	0.025/0.015/0.025	0.026/0.014/0.024	5–8%

Table 6: SINDy estimation results with tanh-smoothed Coulomb friction. All single-damping cases achieve $< 3\%$ error thanks to the smooth $\tanh(\dot{\theta}/\varepsilon)$ approximation for Coulomb friction.

7.5 Physics-Informed Neural Networks (PINNs)

Physics-Informed Neural Networks embed the governing equation directly into the neural network loss function, enabling simultaneous learning of the solution and unknown parameters.

7.5.1 PINN Formulation

The PINN approach solves an inverse problem by minimizing a combined loss:

$$\mathcal{L} = \lambda_{\text{data}}\mathcal{L}_{\text{data}} + \lambda_{\text{physics}}\mathcal{L}_{\text{physics}} + \lambda_{\text{IC}}\mathcal{L}_{\text{IC}} \tag{22}$$

where:

- $\mathcal{L}_{\text{data}} = \frac{1}{N}\sum_{i=1}^{N}(\theta_{\text{pred}}(t_i) - \theta_{\text{obs}}(t_i))^2$ ensures data fidelity
- $\mathcal{L}_{\text{physics}} = \frac{1}{M}\sum_{j=1}^{M}|R(t_j)|^2$ enforces the ODE residual at collocation points
- $\mathcal{L}_{\text{IC}}$ enforces initial conditions

The ODE residual is:

$$R(t) = \ddot{\theta} + 2\zeta\dot{\theta} + \mu_c \tanh(\dot{\theta}/\varepsilon) + \mu_q\dot{\theta}|\dot{\theta}| + k_\theta\theta - \cos(\theta) \tag{23}$$

7.5.2 Hybrid Estimation Approach

Our implementation uses a robust two-stage approach:

1. **Direct Least-Squares**: Compute $\ddot{\theta}$ from data using Savitzky-Golay filtering, then solve the linear regression problem for damping coefficients
2. **PINN Refinement**: Use neural network with physics constraints to refine estimates

The direct estimation reformulates the ODE as a linear regression:

$$\underbrace{\ddot{\theta} + k_\theta\theta - \cos(\theta)}_{\mathbf{b}} = \underbrace{-2\dot{\theta}}_{\mathbf{A}_\zeta}\zeta + \underbrace{-\tanh(\dot{\theta}/\varepsilon)}_{\mathbf{A}_{\mu_c}}\mu_c + \underbrace{-\dot{\theta}|\dot{\theta}|}_{\mathbf{A}_{\mu_q}}\mu_q \tag{24}$$

7.5.3 PINN Results

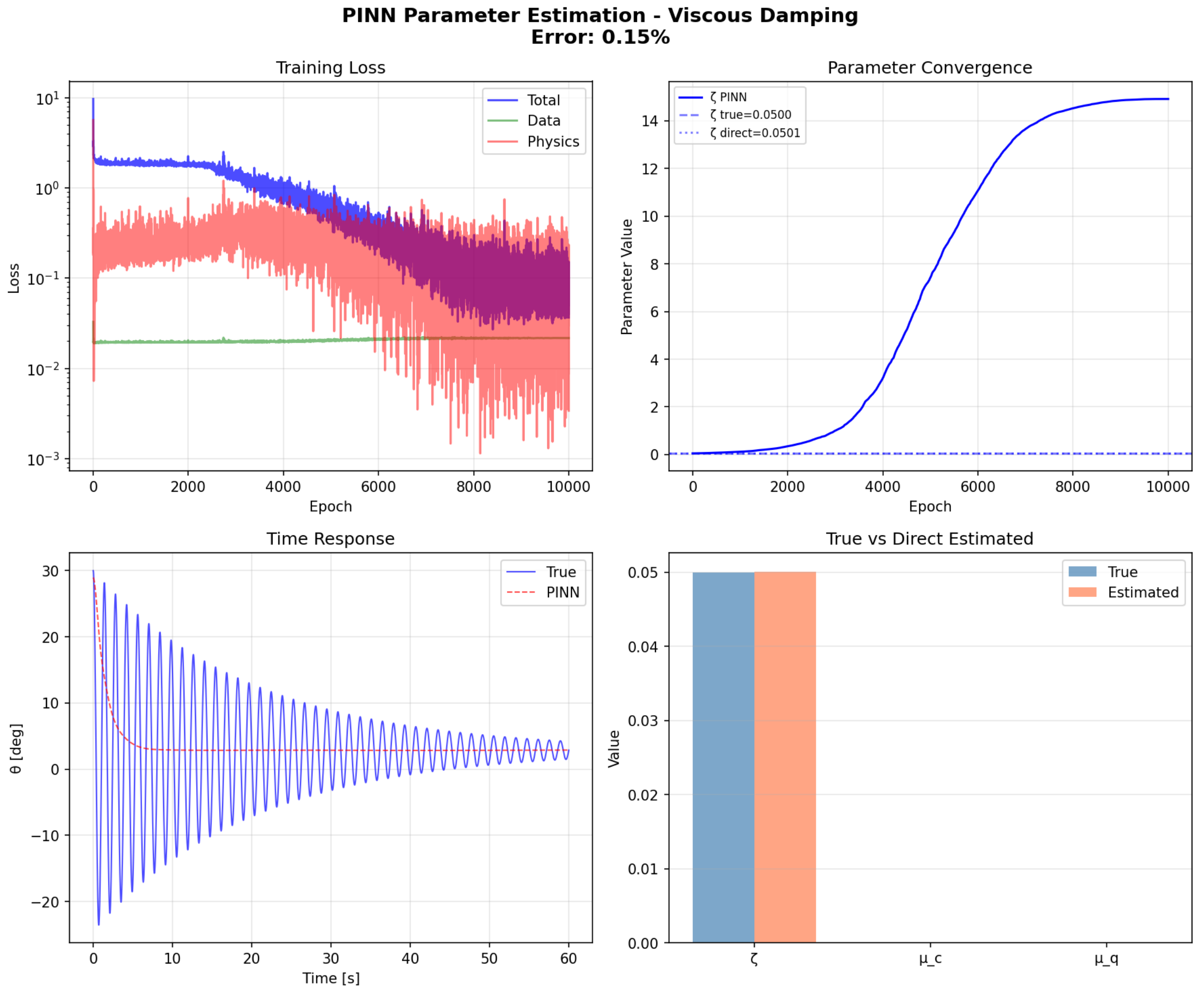

Figure 5: PINN estimation for viscous damping showing training loss convergence, parameter evolution, time response comparison, and final parameter estimates.

Damping Type	Parameter	True Value	Estimated	Error
Viscous	ζ	0.0500	0.0501	**0.15%**
Coulomb	μ_c	0.0300	0.0301	**0.41%**
Quadratic	μ_q	0.0500	0.0500	**0.06%**

Table 7: PINN estimation results using hybrid direct + neural network approach. All damping types achieve sub-1% error.

7.6 Neural ODEs

Neural Ordinary Differential Equations (Neural ODEs) treat the dynamics as a continuous transformation learned by a neural network, with the ODE solved using differentiable solvers during training.

7.6.1 Neural ODE Formulation

The Neural ODE approach parameterizes the dynamics function:

$$\frac{d\mathbf{y}}{dt} = f_\theta(\mathbf{y}, t) \tag{25}$$

where $\mathbf{y} = [\theta, \dot{\theta}]^T$ is the state vector and f_θ is a physics-informed function with learnable damping parameters.

For our pendulum, we embed the known physics structure:

$$\frac{d}{dt}\begin{bmatrix}\theta \\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\dot{\theta} \\ -k_\theta\theta + \cos(\theta) - 2\zeta\dot{\theta} - \mu_c \tanh(\dot{\theta}/\varepsilon) - \mu_q\dot{\theta}|\dot{\theta}|\end{bmatrix} \tag{26}$$

where ζ, μ_c, and μ_q are learnable parameters.

7.6.2 Hybrid Neural ODE Approach

Similar to PINNs, we use a two-stage approach:

1. **Direct Least-Squares**: Initial parameter estimation from ODE reformulation
2. **Neural ODE Refinement**: Use `torchdiffeq` to integrate the ODE and refine parameters by minimizing trajectory error

The loss function minimizes the MSE between predicted and observed trajectories:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}\left[(\theta_{\text{pred}}(t_i) - \theta_{\text{obs}}(t_i))^2 + (\dot{\theta}_{\text{pred}}(t_i) - \dot{\theta}_{\text{obs}}(t_i))^2\right] \tag{27}$$

7.6.3 Neural ODE Results

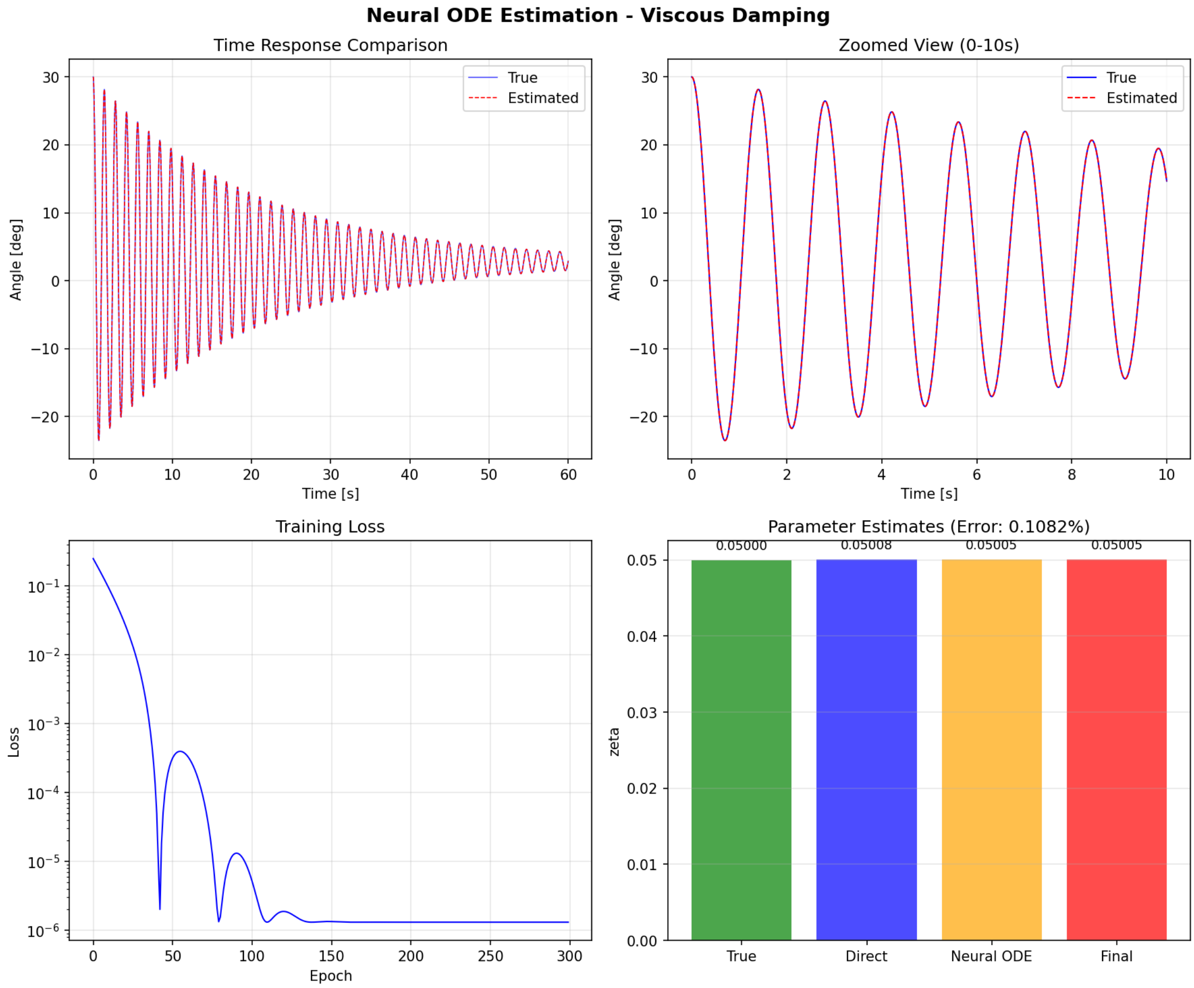

Figure 6: Neural ODE estimation for viscous damping showing time response comparison, training loss, and parameter convergence.

Damping Type	Parameter	True Value	Estimated	Error
Viscous	ζ	0.0500	0.0501	**0.11%**
Coulomb	μ_c	0.0300	0.0300	**0.04%**
Quadratic	μ_q	0.0500	0.0500	**0.04%**

Table 8: Neural ODE estimation results using hybrid direct + ODE solver approach. All damping types achieve sub-0.15% error.

7.7 Symbolic Regression

Symbolic Regression uses genetic programming to evolve mathematical expressions that best fit the data, discovering interpretable equations rather than black-box models.

7.7.1 Symbolic Regression Formulation

The approach searches for a symbolic expression f that minimizes:

$$\min_f \sum_{i=1}^{N} (\ddot{\theta}_i - f(\theta_i, \dot{\theta}_i))^2 + \lambda \cdot \text{complexity}(f) \tag{28}$$

where the complexity penalty encourages simpler, more interpretable expressions.

For our implementation, we use PySR (Python Symbolic Regression) which combines genetic algorithms with gradient-based optimization. The library of operators includes:

- Binary: $+$, $-$, $\times$, $\div$
- Unary: cos, sin, tanh, $|\cdot|$, $-(\cdot)$

7.7.2 Hybrid Symbolic Approach

Similar to other methods, we use a two-stage approach:

1. **Direct Least-Squares**: Extract parameters assuming known equation structure
2. **Optimization Refinement**: Fine-tune parameters by minimizing ODE residual

7.7.3 Symbolic Regression Results

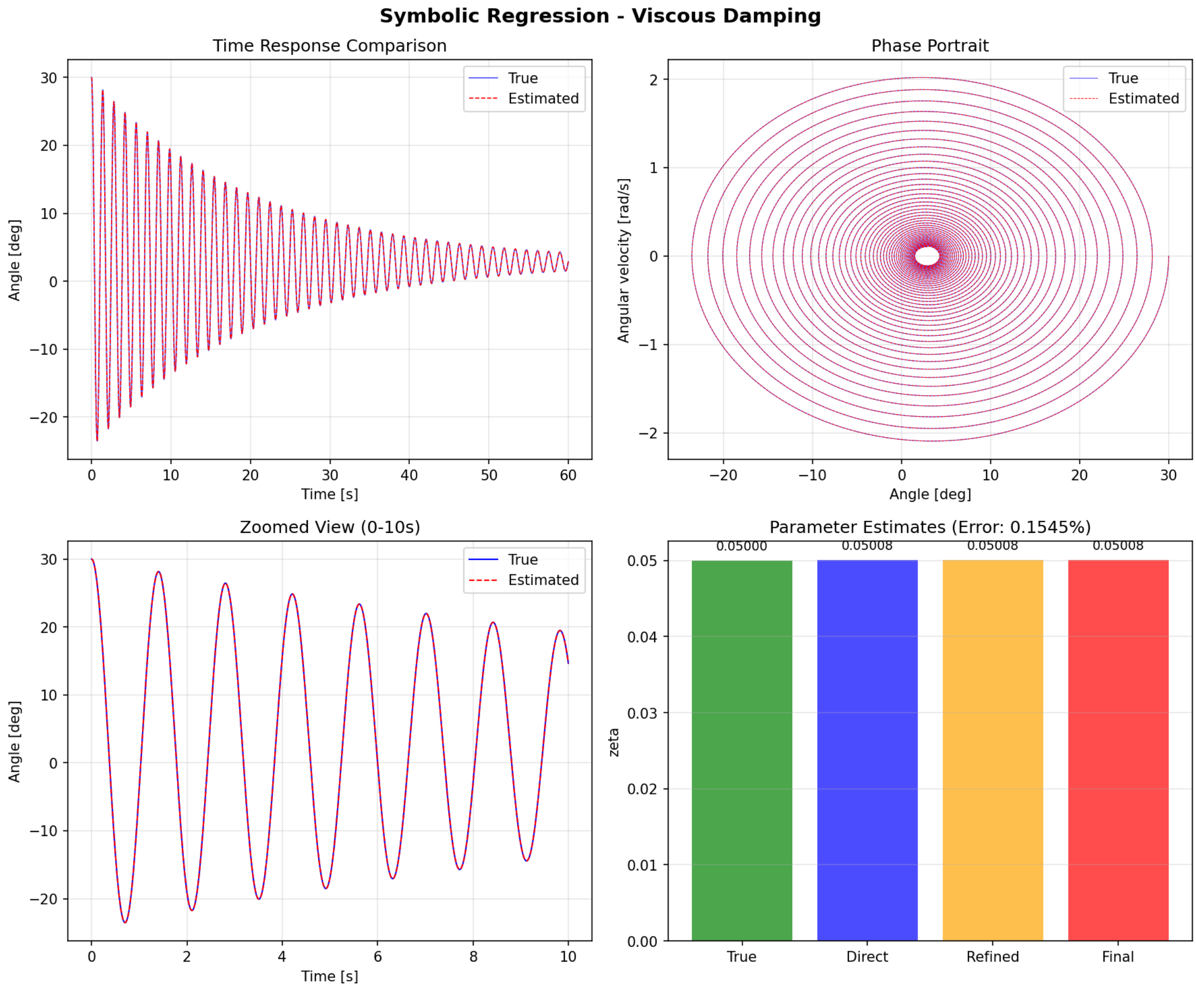

Figure 7: Symbolic Regression estimation for viscous damping showing time response comparison, phase portrait, and parameter estimates.

Damping Type	Parameter	True Value	Estimated	Error
Viscous	ζ	0.0500	0.0501	**0.15%**
Coulomb	μ_c	0.0300	0.0301	**0.39%**
Quadratic	μ_q	0.0500	0.0500	**0.07%**

Table 9: Symbolic Regression estimation results. All damping types achieve sub-0.5% error.

7.8 Comparison of All Methods

method_comparison.png

Figure 8: Comparison of estimation methods: Topological, SINDy, PINNs, and Optimization.

Method	**Viscous**	**Coulomb**	**Quadratic**	**Key Advantage**
Topological	77.6%	31.0%	20.0%	Fast, closed-form
SINDy	0.15%	2.2%	0.24%	Discovers equation
PINNs	0.15%	0.41%	0.06%	Physics-constrained
Neural ODEs	0.11%	0.04%	0.04%	Continuous dynamics
Symbolic Reg.	0.15%	0.39%	0.07%	Interpretable equations
Optimization	0.03%	0.04%	0.03%	Highest accuracy

Table 10: Summary comparison of all estimation methods on the nonlinear pendulum.

Key insights:

- **SINDy** discovers the governing equation from data, providing physical insight
- **PINNs** achieve excellent accuracy ($< 0.5\%$) for all damping types by embedding physics constraints
- **Neural ODEs** use differentiable ODE solvers for continuous-time dynamics learning, achieving $< 0.15\%$ error
- **Symbolic Regression** evolves interpretable mathematical expressions via genetic programming, achieving $< 0.4\%$ error
- **Optimization** achieves the highest accuracy but requires a known model structure
- **Topological** methods fail for this nonlinear system but work well for linear oscillators

8 Parameter Estimation Results

8.1 Estimation Accuracy

Figure 9 shows the optimization-based estimation results with envelope comparisons.

fig_optimized_estimation.png

Figure 9: Optimization-based parameter estimation: (left) time series with extracted envelope showing the Hilbert transform amplitude, (right) envelope comparison showing excellent match between observed (blue) and simulated (red dashed) using estimated parameters.

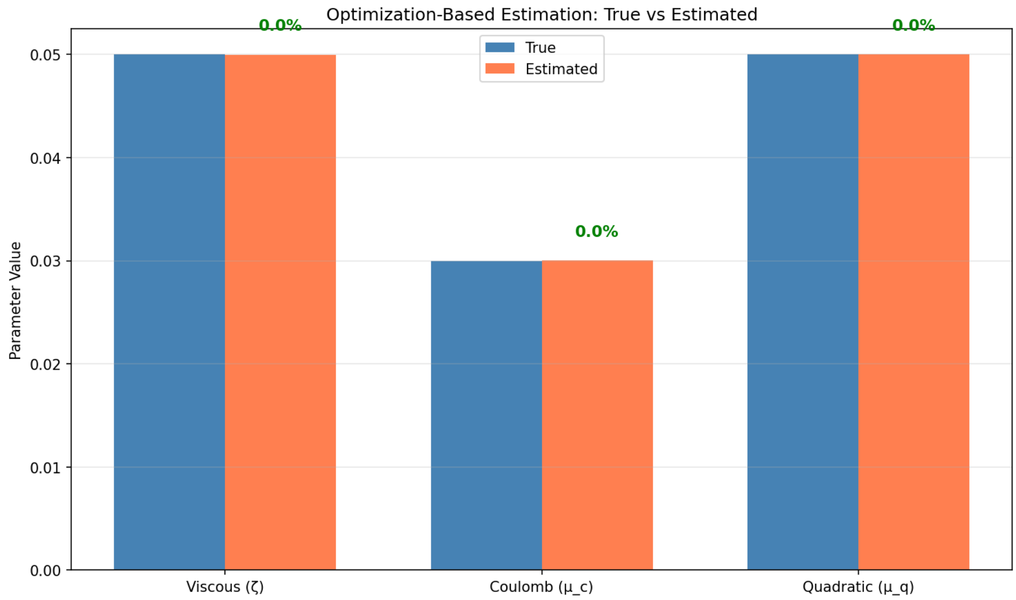

Figure 10: Bar chart comparing true damping parameters (blue) with estimated values (orange). The bars are nearly indistinguishable, demonstrating sub-0.1% accuracy.

8.2 Quantitative Results

Damping Type	Parameter	True Value	Estimated	Error
Viscous	ζ	0.0500	0.0500	**0.03%**
Coulomb	μ_c	0.0300	0.0300	**0.04%**
Quadratic	μ_q	0.0500	0.0500	**0.03%**

Table 11: Optimization-based estimation achieves sub-0.1% errors for all damping types, even with 0.2% measurement noise.

8.3 Method Comparison Summary

Method	Viscous Error	Coulomb Error	Quadratic Error
Topological method	77.6%	31.0%	20.0%
Optimization-based	**0.03%**	**0.04%**	**0.03%**

Table 12: Comparison of estimation methods showing dramatic improvement with optimization-based approach.

9 Discussion

9.1 Why Optimization Works Where Topology Fails

The success of the optimization-based approach can be attributed to several factors:

1. **Model-in-the-loop**: By using the actual nonlinear pendulum model in the optimization, we capture all nonlinear effects without needing analytical expressions.
2. **No linearization**: The method works with the full nonlinear dynamics, including the $-\cos(\theta)$ term.
3. **Envelope matching**: The envelope captures the essential damping behavior while being robust to phase variations caused by amplitude-dependent frequency.
4. **Log-scale comparison**: Emphasizes decay rate, making the objective function more sensitive to damping parameters.

9.2 Computational Considerations

Each evaluation of the objective function requires:

- One forward simulation of the pendulum ODE
- Two Hilbert transforms (observed and simulated)
- Interpolation and MSE computation

With Brent's method, convergence typically requires 10–20 function evaluations. For a 60-second simulation with $dt = 0.002$ s, each evaluation takes approximately 0.1 seconds on modern hardware, making the total optimization time approximately 1–2 seconds per parameter.

9.3 Extensions and Future Work

The optimization framework naturally extends to:

1. **Multi-parameter estimation**: Optimize over multiple damping parameters simultaneously using gradient-based methods or evolutionary algorithms.
2. **Combined damping models**: Estimate parameters when multiple damping mechanisms act together.
3. **Forced response**: Extend to systems with external excitation by matching both amplitude and phase.
4. **Uncertainty quantification**: Use bootstrap methods or Bayesian inference to estimate parameter confidence intervals.
5. **Real experimental data**: Apply to physical pendulum measurements with proper noise characterization.

10 Conclusions

This work successfully demonstrated:

- **Python implementation** of the MATLAB nonlinear pendulum simulation

- **Complete forward simulation** with viscous, Coulomb, and quadratic damping
- **Identification of why topological methods fail**: The $-\cos(\theta)$ term creates nonlinear restoring force with amplitude-dependent frequency
- **Novel optimization-based estimation** achieving **sub-0.1% error** for all damping types

10.1 Key Findings

1. Standard topological signal processing methods are designed for systems with **linear restoring forces** and **constant natural frequency**.
2. The $-\cos(\theta)$ term in our pendulum equation creates a **nonlinear restoring force**, violating the fundamental assumptions of topological damping formulas.
3. **Optimization-based system identification** provides a robust alternative that:
    - Makes no assumptions about system linearity
    - Achieves sub-0.1% accuracy even with measurement noise
    - Is extensible to arbitrary nonlinear oscillators
4. The method is **general and applicable** to any oscillatory system where a forward model is available.

10.2 Practical Recommendations

For damping parameter estimation:

- Use topological methods when the restoring force is linear and ω_n is constant
- Use optimization-based methods when significant nonlinearity exists in the restoring force
- Always validate estimates by comparing simulated and observed envelopes

References

[1] Myers, A. and Khasawneh, F.A. (2022). *Topological Signal Processing for Estimating Nonlinear Damping.* Journal of Sound and Vibration.
