

# Addressing the Closure Problem: State Estimation and Neural Approaches for Partially Observable Dynamical Systems

Matthew G. Galarza

November 19, 2025

## 1 Introduction

A fundamental challenge in the modeling of complex dynamical systems is one of closure where the evolution of the resolved variables depends on unresolved states, processes, and scales that are not directly accessible. This mismatch between the “true” system and its analytical or reduced approximation stems from partial observability, missing or inaccurate physics, and parameter uncertainty. The challenge is ubiquitous, appearing in a variety of diverse fields including turbulence modeling, climate science, molecular dynamics, and biological systems.

### 1.1 Mathematical Formulation

Consider a general dynamical system with state variables partitioned into resolved (observable) and unresolved (hidden) components:

$$\frac{d}{dt} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} f(\mathbf{u}, \mathbf{v}, t) \\ g(\mathbf{u}, \mathbf{v}, t) \end{bmatrix} \quad (1)$$

where  $\mathbf{u}$  represents resolved/observable states and  $\mathbf{v}$  represents unresolved/hidden states. When only  $\mathbf{u}$  is observable, the evolution equation becomes:

$$\frac{d\mathbf{u}}{dt} = f(\mathbf{u}, \mathbf{v}, t) \quad (2)$$

The closure problem emerges because the right-hand side depends on the unobservable  $\mathbf{v}$ , requiring a closure model:

$$\frac{d\mathbf{u}}{dt} = f(\mathbf{u}, \mathcal{C}(\mathbf{u}, t), t) \quad (3)$$

where  $\mathcal{C}(\mathbf{u}, t)$  is the closure approximation that models the effect of unresolved variables.

### 1.2 Classical Approaches to Closure

#### 1.2.1 Statistical Closure

In turbulence modeling, Reynolds averaging leads to the Reynolds-Averaged Navier-Stokes (RANS) equations where the closure problem manifests as the Reynolds stress tensor. Classical approaches include:

- **Eddy viscosity models:** Approximate unresolved stresses through enhanced dissipation
- **Scale similarity models:** Assume self-similar behavior across scales
- **Gradient transport models:** Model fluxes as proportional to mean gradients

### 1.2.2 Phenomenological Closure

Many fields develop closure models based on physical reasoning:

- **Subgrid-scale models:** In Large Eddy Simulation (LES), model energy cascade to unresolved scales
- **Coarse-graining:** In molecular dynamics, derive effective potentials for coarse-grained representations
- **Parameterization schemes:** In climate models, represent cloud physics and convection through simplified relationships

## 1.3 Modern Data-Driven Approaches

### 1.3.1 Neural Closure Models

Recent advances leverage neural networks to learn closure relationships from data:

$$\mathcal{C}(\mathbf{u}, t) = \mathcal{N}_{\theta}(\mathbf{u}, t) + \int_{-\infty}^t \mathcal{K}_{\phi}(\mathbf{u}(s), t-s) ds \quad (4)$$

where  $\mathcal{N}_{\theta}$  captures Markovian (instantaneous) effects and  $\mathcal{K}_{\phi}$  models non-Markovian (memory) effects.

### 1.3.2 Physics-Informed Machine Learning

Hybrid approaches combine physical constraints with data-driven learning:

- **Universal Differential Equations:** Augment known physics with neural network corrections
- **Neural Differential-Algebraic Equations:** Enforce conservation laws through algebraic constraints
- **Operator learning:** Learn mappings between function spaces for closure relationships

## 1.4 Key Challenges in Closure Modeling

1. **Structural uncertainty:** The functional form of the closure is often unknown
2. **Scale separation:** Unresolved scales may span multiple orders of magnitude
3. **Non-Markovian effects:** Hidden states may have long memory effects
4. **Generalization:** Closure models must extrapolate to conditions beyond training data
5. **Stability:** Inappropriate closures can lead to numerical instabilities or unphysical solutions
6. **Interpretability:** Understanding what the closure model has learned remains challenging

## 2 Specific Problem Formulation

### 2.1 Application to Electromechanical Systems

We now apply these closure modeling concepts to a specific challenge in electromechanical systems where complex internal dynamics manifest through limited observable quantities. Our system exemplifies many of the classical closure challenges: multiple unobservable states influencing observable dynamics, cascaded physics across different scales, and regime-switching behavior that complicates traditional modeling approaches.

### 2.2 System Description

We consider a complex electromechanical system governed by an analytical model with six state variables. The system dynamics can be expressed as:

$$\frac{d\mathbf{x}}{dt} = f_{\text{analytical}}(\mathbf{x}, u_{\text{force}}; \boldsymbol{\theta}_{\text{physics}}) \quad (5)$$

where  $\mathbf{x} = [x_1, x_2, x_3, x_4, x_5, V]^T$  represents the full state vector with  $V$  being the voltage output,  $u_{\text{force}}$  is the input force, and  $\boldsymbol{\theta}_{\text{physics}}$  are known physical parameters.

### 2.3 Observability Challenge

The fundamental challenge arises from partial observability:

- **Available measurements:** Input force  $u_{\text{force}}(t)$  and output voltage  $V(t)$
- **Hidden states:** Five internal states  $\{x_1, x_2, x_3, x_4, x_5\}$  that influence system dynamics but are not directly observable
- **Closure problem:** How to accurately model the influence of unobserved states on the observable voltage dynamics

This creates a closure problem where the evolution of the observable state depends on hidden states through complex cascaded physics:

$$\frac{dV}{dt} = g(V, \mathbf{x}_{\text{hidden}}, u_{\text{force}}) \quad (6)$$

where  $\mathbf{x}_{\text{hidden}} = [x_1, x_2, x_3, x_4, x_5]^T$  are not available from experimental data.

### 2.4 Hybrid Dynamics

The system exhibits regime-switching behavior when sufficient input force causes electrode collision, introducing discontinuous dynamics:

$$f(\mathbf{x}, u) = \begin{cases} f_{\text{normal}}(\mathbf{x}, u) & \text{if no collision} \\ f_{\text{collision}}(\mathbf{x}, u) + F_{\text{contact}} & \text{if collision occurs} \end{cases} \quad (7)$$

## 3 Proposed Solution Methods

### 3.1 Neural Differential-Algebraic Equations via Operator Splitting

Building on recent advances in neural DAEs [1], we propose an operator splitting approach that separates known physics from learned closure terms.

### 3.1.1 Mathematical Framework

The system is reformulated as a semi-explicit DAE:

$$\frac{dx}{dt} = f(\mathbf{x}, \mathbf{y}, u; \boldsymbol{\theta}_f) \quad (8)$$

$$0 = g(\mathbf{x}, \mathbf{y}, u) \quad (9)$$

where  $\mathbf{x}$  are differential states,  $\mathbf{y}$  are algebraic states representing closure relationships, and  $g$  enforces physical constraints.

### 3.1.2 Operator Splitting Scheme

Following the Lie-Trotter splitting methodology:

$$\mathbf{y}^{(t+\Delta t)} = h\left(\mathbf{x}^{(t)}, V^{(t)}, u^{(t)}; \boldsymbol{\theta}_h\right) \quad (\text{Algebraic Update})$$

$$\mathbf{x}^{(t+\Delta t)} = \text{ODESolve}\left(f, \{\mathbf{x}^{(t)}, \mathbf{y}^{(t+\Delta t)}, u^{(t)}\}; \boldsymbol{\theta}_f\right) \quad (\text{Differential Update})$$

The neural surrogate  $h : \mathbb{R}^{n_x} \times \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}^{n_y}$  learns the algebraic closure mapping from observable quantities.

## 3.2 Universal Differential Equations with Physics Constraints

An alternative approach combines known analytical models with neural network corrections:

$$\frac{d\mathbf{x}}{dt} = f_{\text{analytical}}(\mathbf{x}, u) + \mathcal{N}_{\boldsymbol{\theta}}(\mathbf{x}, u, t) \quad (10)$$

where  $\mathcal{N}_{\boldsymbol{\theta}}$  is a neural network that learns the discrepancy between the analytical model and true system dynamics.

### 3.2.1 Physics-Informed Loss Function

Training employs a multi-objective loss:

$$\mathcal{L}(\boldsymbol{\theta}) = \lambda_1 \|\mathbf{V}_{\text{pred}} - \mathbf{V}_{\text{obs}}\|^2 + \lambda_2 \mathcal{L}_{\text{constraints}} + \lambda_3 \|\boldsymbol{\theta}\|^2 \quad (11)$$

where:

- $\mathcal{L}_{\text{constraints}}$  enforces conservation laws and physical constraints
- The regularization term prevents overfitting to noise

## 3.3 Generalized Neural Closure Models

Following the framework of neural closure models for turbulent systems, we incorporate both Markovian and non-Markovian closure terms:

$$\frac{\partial u}{\partial t} = \mathcal{D}(u) + \mathcal{M}(u; \boldsymbol{\theta}_M) + \int_{-\infty}^t \mathcal{K}(u(s), t-s; \boldsymbol{\theta}_{NM}) ds \quad (12)$$

where:

- $\mathcal{D}(u)$  represents the known low-fidelity model
- $\mathcal{M}(u; \boldsymbol{\theta}_M)$  captures instantaneous closure effects
- $\mathcal{K}$  models memory effects through time delays

## 4 Implementation Strategy

### 4.1 Phase 1: Non-Collision Regime

#### 1. Data Preprocessing:

- Filter experimental data to exclude collision events
- Normalize states using physics-based bounds
- Create delay embeddings for voltage time series

#### 2. Model Training:

- Initialize neural networks with small random weights
- Use adaptive learning rates with gradient clipping
- Employ early stopping based on validation loss

#### 3. Validation:

- Test extrapolation to unseen operating conditions
- Verify conservation law satisfaction
- Assess robustness to measurement noise

### 4.2 Phase 2: Hybrid Dynamics Extension

1. Implement regime detection based on state constraints
2. Train separate closure models for each regime
3. Develop smooth transition mechanisms between regimes

## 5 Expected Advantages

### 5.1 Operator Splitting Approach

- **Interpretability:** Clear separation between known physics and learned corrections
- **Constraint satisfaction:** Algebraic constraints explicitly enforced at each timestep
- **Computational efficiency:** Leverages existing ODE solvers for differential components

### 5.2 Physics-Informed Learning

- **Data efficiency:** Requires only observable voltage measurements
- **Generalization:** Physics constraints prevent unphysical extrapolation
- **Noise robustness:** Regularization through physical laws

## 6 Preliminary Results and Validation

### 6.1 Benchmark Systems

Initial validation on benchmark problems demonstrates:

- Tank-manifold system: Successfully learned unknown area-height profiles with  $\text{MSE} < 10^{-2}$
- Network dynamics: Captured complex feedback mechanisms with 20 dB SNR noise tolerance
- Extrapolation capability: Models generalize to unseen parameter regimes

### 6.2 Challenges and Mitigation Strategies

1. **Numerical stability:** Addressed through adaptive timestep control and gradient clipping
2. **Local minima:** Mitigated using ensemble training with different initializations
3. **Overfitting:** Prevented through physics-based regularization and cross-validation

## 7 Future Work

### 7.1 Immediate Goals

- Complete implementation of operator splitting framework for the 6-state system
- Conduct systematic hyperparameter optimization
- Compare performance against baseline methods (pure neural ODE, Kalman filtering)

### 7.2 Long-term Objectives

- Extend to full hybrid dynamics with collision modeling
- Develop theoretical guarantees for closure model convergence
- Investigate transfer learning to related electromechanical systems

## 8 Conclusion

The proposed hybrid approach combining operator splitting, physics-informed learning, and neural closure models offers a principled framework for addressing the partial observability challenge in complex dynamical systems. By leveraging both analytical knowledge and data-driven learning, this methodology promises improved accuracy, interpretability, and generalization compared to purely black-box approaches.

## References

- [1] Koch, J., Shapiro, M., Sharma, H., Vrabie, D., & Drgoňa, J. (2024). Learning Neural Differential Algebraic Equations via Operator Splitting. *Proceedings of the Conference on Neural Information Processing Systems*.
- [2] Gupta, A., & Lermusiaux, P. F. (2021). Neural closure models for dynamical systems. *Proceedings of the Royal Society A*, 477(2252), 20210063.
- [3] Chen, R. T., Rubanova, Y., Bettencourt, J., & Duvenaud, D. K. (2018). Neural ordinary differential equations. *Advances in neural information processing systems*, 31.
- [4] Rackauckas, C., Ma, Y., Martensen, J., Warner, C., Zubov, K., Supekar, R., ... & Edelman, A. (2020). Universal differential equations for scientific machine learning. *arXiv preprint arXiv:2001.04385*.