EMILY: Extracting sparse Model from ImpLicit dYnamics

Ayan Banerjee and ABANERJ3@ASU.EDU
Sandeep K.S. Gupta SANDEEP.GUPTA@ASU.EDU
IMPACT Lab, Arizona State University

Editors: Cecília Coelho, Bernd Zimmering, M. Fernanda P. Costa, Luís L. Ferrás, Oliver Niggemann

Abstract

Sparse model recovery requires us to extract model coefficients of ordinary differential equations (ODE) with few nonlinear terms from data. This problem has been effectively solved in recent literature for the case when all state variables of the ODE are measured. In practical deployments, measurements of all the state variables of the underlying ODE model of a process are not available, resulting in implicit (unmeasured) dynamics. In this paper, we propose EMILY, that can extract the underlying ODE of a dynamical process even if much of the dynamics is implicit. We show the utility of EMILY on four baseline examples and compare with the state-of-the-art techniques such as SINDY-MPC. Results show that unlike SINDY-MPC, EMILY can recover model coefficients accurately under implicit dynamics.

Keywords: sparse recovery, SINDY-MPC, automatic differentiation, implicit dynamics

1. Introduction

Recovering ordinary differential equation (ODE) based first principle models that estimate measurements of a dynamical process, also called sparse model recovery, is of recent interest (Banerjee and Gupta, 2024; Maity et al., 2024). While model learning mainly focuses on fitting the data (Lamrani et al., 2018, 2021), model recovery has two distinct objectives: a) extracting accurate coefficients of the underlying ODE, and b) fitting the resulting ODE model to data. Model recovery has applications in explainable digital twin learning, which are becoming increasingly useful in many biomedical fields including precision medicine, generation of patient specific data to aid in artificial intelligence for health, expert guided systems (Kamboj et al., 2024) and in general in large scale in-silico testing of human centered autonomous systems (Banerjee et al., 2023). The requirement of explainability implies that the underlying model of the digital twin should be based on first principles such as physics laws, or physiological/ mechanical/ chemical processes. A hallmark of such models is that they are sparse in the function space, i.e., the models only have a few nonlinear components among the combinatorial options.

Recently, there has been several techniques proposed for this model recovery task including Spare Indentification of Nonlinear Dynamics (SINDY) (Kaiser et al., 2018), SINDY-Model Predicitive Control (MPC) (Kaiser, 2024), physics informed neural networks + sparse regression for partial differential equations (PDE) (Chen et al., 2021). However, one of the major drawbacks of all existing techniques is that they assume full measurability of the process. This implies that the state-of-the-art model recovery techniques require measurements of all the state variables. In practice, this is a major limiting assumption. This is because, it requires installation of multiple sensors only for the purpose of digital twin learning. In human-centered systems, this requires more wearable or intrusive sensors, reducing usability, increasing costs, and making deployment less feasible.

Table 1: Related works in model recovery.

Approach	Implicit	Assumptions
Ho Kalman, Eigen system (Oymak and Ozay, 2021)	No	Linear system
Genetic Algorithm (Schmidt and Lipson, 2009)	No	Low dimensional nonlinear systems
SINDy (Quade et al., 2018)	No	Known sparsity threshold
SINDy-MPC (Kaiser et al., 2018)	No	Known sparsity threshold
E-SINDY (Fasel et al., 2022)	Weak	Known sparsity threshold
Neural ODE + metriplectic structure (Lee et al., 2021)	No	Known metriplectic structure
PINNs + Sparse Regression (Chen et al., 2021)	Weak	Physics loss for original coefficients
This paper	**Yes**	**Black box ODE solver in the loss**

To further illustrate the point consider that our aim is to learn a metabolic model for an individual with Type 1 Diabetes for the purpose of large scale in-silico testing of a personalized automated insulin delivery (AID) system. The metabolic model comes from first principle knowledge of the glucose insulin interaction in the human body such as Bergman Minimal Model (Bergman, 2021) (Eqn. 1).

$$\begin{aligned} \dot{i(t)} &= -ni(t) + p_4 u_1(t), \dot{i_s(t)} = -p_1 i_s(t) + p_2(i(t) - i_b) \\ \dot{G(t)} &= -i_s(t)G(t) - p3(G(t) - G_b) + u2(t)/VoI, \end{aligned} \tag{1}$$

where $i(t)$ is the interstitial insulin concentration, $i_s(t)$ is the plasma insulin concentration, and $G(t)$ is the plasma glucose concentration, u_1 is the external insulin delivery from the AID, and u_2 is the meal ingested by the individual. Whereas, p_1, p_2, i_b, p_3, p_4, n, and $1/V_oI$ are all patient specific model coefficients. As seen from Eqn. 1, it is sparse in the nonlinear functional space since it only uses eight of the possible 2^5 nonlinear terms if we limit to the polynomial order 2 functional space.

To extract this model from measurements, state-of-the-art techniques will need measurements of i, i_s, G, u_1, and u_2. However, in a real deployment only G, u_1, and u_2 are available which are logged by a continuous glucose monitor (CGM) and the insulin pump. A comprehensive blood panel is required to measure i_s and i, which is infeasible in deployment.

Unavailability of measurements of the state variables result in the problem of sparse model recovery with implicit dynamics, i.e. dynamics that is not measured. Currently, to the best of our knowledge, state-of-art techniques may not be able to solve this problem.

Contribution: The main contribution of this paper is that it proposes EMILY, a solution to the problem of sparse model recovery with implicit dynamics. EMILY uses neural architectures with automated differentiation such as liquid time constant neural networks (LTC-NN) in a carefully crafted network to represent model coefficients of a sparse model as a function of implicit dynamics. It then searches through a constrained set of implicit dynamics guided by an ODE solver to recover the original model coefficients of the sparse model. We show application of EMILY on baseline examples established in Kaiser et al. (2018) and compare it with SINDY-MPC for model recovery with all variables measured, and extensions of SINDY-MPC by combining with non-sparse system identification techniques for the model recovery with implicit dynamics problem.

2. Related Works

Table 1 summarizes the recent works on model recovery. In the linear domain, the problem of system identification is solved using techniques such as Ho Kalman or Eigen system realization algorithm (ERA) Oymak and Ozay (2021). Such techniques attempt to fit a linear model to data but they are not designed to extract the underlying sparsity structure.

The initial work on extracting nonlinear model from data used stratified symbolic regression and genetic programming Schmidt and Lipson (2009). This approach did not scale with the dimension of the state space and also did not consider the implicit dynamics. Significant breakthrough was achieved through introduction of sparse identification of nonlinear dynamics (SINDy) and SINDy-MPC Kaiser et al. (2018) that tackles inputs. We show in this paper that SINDy-MPC cannot, however, efficiently tackle implicit dynamics.

Physics-informed Neural Networks (PINN) utilize the concept of automatic differentiation (AD) to perform accurate forward and inverse analysis of nonlinear physics models and it has been used in many practical domains Chen et al. (2021). However, such architectures are black boxes, are only used to solve the differential equation, and cannot recover the model. Recently, PINNs have been integrated with sparse regression to recover model coefficients Chen et al. (2021). A major assumption in these approaches is the knowledge of the physics loss for the original model coefficients. This is an impractical circular assumption since the original model coefficients are unknown in real-world examples. Recently, with the advent of the neural ordinary differential equation (NODE) architecture there has been a class of approaches for forecasting while maintaining metriplectic structures Lee et al. (2021), i.e. algebraic structures in models induced by laws of physics such as energy conservation Lee et al. (2021). In such approaches implicit dynamics were not originally addressed. Attempts have been made to incorporate extraction of implicit models using PINNs and SINDy strategy Kaheman et al. (2020), however, the unmeasured state variables are only limited to the differentials of the original state variables. We term these approaches as "weakly implicit".

3. Preliminaries and Problem Statement

We consider a nonlinear ordinary differential equation model in Eqn: 2 with n dimensional state space represented by vector X.

$$\dot{X} = f(X, U, \theta), \tag{2}$$

where f is a parameterized nonlinear function that is sparse in the nonlinear function domain, U is the m dimensional external input, and θ is the p dimensional model coefficient.
Sparsity: An n-dimensional model with M^{th} order non-linearity can utilize $\binom{M+n}{n}$ nonlinear terms. A sparse model only includes a few nonlinear terms $p << \binom{M+n}{n}$. Sparsity structure of a model is the set of nonlinear terms used by the model.
Implicit dynamics: Sensing constraints prevent the measurement of all the state variables in X. Therefore, we assume that many state variables cannot be measured or estimated. This is captured by the sensing matrix C, a diagonal matrix where $c_{ii} = 1$ if a sensor measures $x_i \in X$, and $c_{ii} = 0$, otherwise. Consequently, only the sampled traces of $Y = CX$ are available as sensed data, leaving much of the system's nonlinear dynamics implicit.
Identifiable model: A model in Eqn. 2 is identifiable Verdière et al. (2020) for a sensing matrix C, if there exists a time $t_I > 0$, such that $\forall \theta, \tilde{\theta} \in \mathcal{R}^p$, we have:

$$\forall t \in [0, t_I],\ Cf(X(t), U(t), \theta) = Cf(X(t), U(t), \tilde{\theta}) \implies \theta = \tilde{\theta}. \tag{3}$$

Eqn. 3 effectively means that a model is identifiable if two different model coefficients do not result in identical measurements Y. Basically, this means $\forall \theta_i \in \theta, \frac{dY}{d\theta_i} \neq 0$. In this paper, we assume that despite presence of implicit dynamics the model is identifiable.

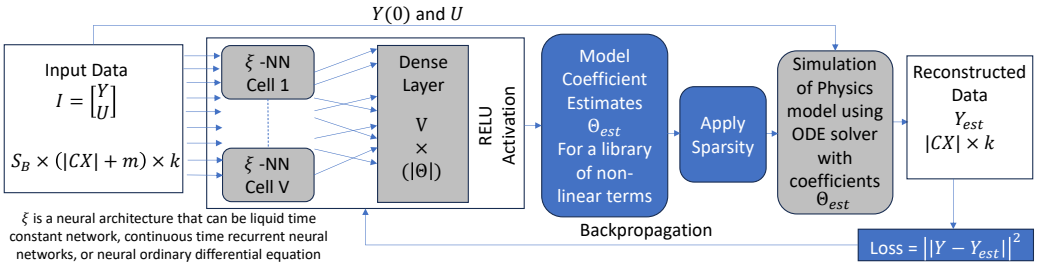

Figure 1: EMILY: Solution for sparse model recovery with implicit dynamics.

Problem Definition 1 (Sparse Model Recovery from Implicit Dynamics): Given N samples of measurements Y and inputs U, obtained from a sparse model in Eqn. 2 with a measurement matrix C such that θ is identifiable, recover $\tilde{\theta}$ such that for $\tilde{Y}$ generated from $f(X, U, \tilde{\theta})$, we have $||Y - \tilde{Y}|| \leq \epsilon$, where ϵ is the maximum tolerable error.

4. Methodology

In EMILY (Fig. 1), we extend neural architectures such as liquid time constant neural network (LTC-NN), continuous time recurrent neural networks (CT-RNN), neural ordinary differential equations (NODE) to obtain advanced neural structures (LTC-NN-MR, CT-RNN-MR, NODE-MR) that can solve the model recovery problem. The forward pass of these advanced neural structures has the same form as bilinear approximations of the implicit dynamics in Eqn. 2 and hence can search through the space of implicit dynamics (proved in Section 5). The measurements of Y, can be used to convert the set of implicit dynamics to an over-determined system of equations that are linear in terms of the model coefficients. As such an over-determined system of equations may have no solution unless either some equations are rejected or are expressed as linear superposition of other equations. To search for a set of consistent equations to estimate model coefficient, a dense layer is utilized. Each output node of the dense layer corresponds to the coefficient of a non-linear term in the library of nonlinear functions that are searched. Given a sparsity threshold of r, any hidden layer output of $\leq r$ is clipped to zero. Then the non-zero hidden layer outputs are used as model structure and the values of the non-zero elements are uses as model coefficients. The search process of the dense layer is guided by a loss function (*ODE loss*) that computes the mean square error between the estimated Y_{est} using an ODE solver **SOLVE**$(Y(0), \Theta, U + U_{ex})$ and the ground truth measurements of Y.

The advanced neural architectures for model recovery (ξ-MR, where ξ is either LTC-NN, CT-RNN, or NODE) (Fig. 1) is implemented by extending the base code available in Hasani (2024). For each example, we extract the training data consisting of temporal traces of Y, and U. The resulting training data is then divided into batches of size S_B. This forms a 3 D tensor of size $S_B \times |Y| + m \times k$.

Each batch is passed through the ξ network with V nodes, resulting in V hidden states. A dense layer is then employed to transform these V hidden states into $p = |\Theta|$ model coefficient estimates. The dense layer is a multi-layer perceptron with ReLU activation function for the model coefficient estimate nodes. The model coefficient estimates, and the initial value $Y(0)$ is passed through an ODE solver, that solves the model in Eqn. 2 with the coefficients Θ_{est}, initial conditions $Y(0)$ and inputs U. The Runge-Kutta integration

method is used in the ODE solver, which gives Y_{est}. The backpropagation of the network is performed using the network loss appended with ODE loss, which is the mean square error between the original trace Y and the estimated trace Y_{est}.

5. Theoretical Foundations: Why does EMILY work?

The forward pass of liquid time constant neural network (LTC-NN) is (Hasani et al., 2021):

$$\frac{dh(t)}{dt} = -\frac{h(t)}{\frac{\rho}{1+\rho f_{NN}(h(t),I(t),t,\omega)}} + f_{NN}(h(t), I(t), t, \omega)(A), \tag{4}$$

where $h(t)$ is one hidden state of the LTC-NN, ρ is a time constant parameter, required to assist any autonomous system to reach equilibrium state. As such existence of the $-h(t)/(\rho \div (1 + \rho f_{NN}(h(t), I(t), t, \omega)))$ term indicates an input dependent time constant that matches the structure of Eqn. 2. f_{NN} is the forward pass and is a function of the hidden states, $I(t)$ is the input to the LTC-NN, ω and A are the parameters of the LTC-NN architecture.

Remark 1 *The forward pass of an LTC-NN architecture generates a set of implicit physical dynamics that are equivalent to a bilinear approximations of the system in Eqn. 2.*

Supporting argument:

Algebraic manipulation of the forward pass of LTC-NN architecture gives the structure of Eqn. 5 which allows an input dependent time constant $\frac{\rho}{1+\rho f_{NN}(h(t),I(t),t,\omega)}$.

$$\frac{dh(t)}{dt} = -\frac{h(t)}{\frac{\rho}{1+\rho f_{NN}(h(t),I(t),t,\omega)}} + f_{NN}(h(t), I(t), t, \omega)(A). \tag{5}$$

The stability criteria for any stable system requires the model to have a time constant term as shown in Eqn. 6.

$$\frac{dX}{dt} = -X/\rho + f_{-\rho}(X) + g(X)U_T, \tag{6}$$

where ρ is the time constant of the system and $f_{-\rho}(.)$ is the unperturbed dynamics obtained by removing the time constant component from $f(.)$.

Assuming that the system described by Eqn. 2 is a dynamic causal system, the bilinear approximation (Friston et al., 2003) of the model in Eqn. 6 results in Eqn. 7.

$$\frac{dX}{dt} \approx -X/\rho + f_{-\rho}(X) + BX + CU_T + \sum_j u_T^j D^j X + H, \tag{7}$$

where $B = \frac{\partial(g(X)U_T)}{\partial X}$, $C = \frac{\partial(g(X)U_T)}{\partial U_T}$, and $D^j = \frac{\partial^2(g(X)U_T)}{\partial X \partial u_T^j}$, H is a constant. Rearranging Eqn. 7, we have the similar form as the LTC-NN forward pass in Eqn. 8.

$$\frac{dX}{dt} \approx -\frac{X}{\frac{\rho}{1+\rho(B+\sum_j u_T^j D^j)}} + (f_{-\rho}(X) + CU_T + H). \tag{8}$$

We observe that Eqn. 8 is the same form as Eqn. 5 if the input to the LTC-NN $I(t)$ is a concatenation of Y and U_T. The hidden layers of the LTC-NN model an inflated set of implicit dynamics which may include the unmeasured system variables of the sparse model.

Remark 2 *The inflated set of implicit dynamics in LTC-NN induces an over-determined set of equations in the coefficients of the bilinear approximation of nonlinear model.*

Table 2: Benchmark Examples available in Kaiser et al. (2018).

Example	*Variables*	*Inputs*	*Implicit*	*No of coefficients*
Lotka Volterra	x_1, x_2	1	x_1	4
Chaotic Lorenz System	x_1, x_2, x_3	1	x_1 x_2	4
F8 Crusader tracking	x_1, x_2, x_3	1	x_2	20
Pathogenics attack	x_1, x_2, x_3 x_4, x_5	1	x_3, x_4	13

Supporting argument: The training process of LTC-NN fixes weights and instantiates the hidden layer outputs. The values of the unmeasured variables in X is estimated by the hidden state in each training step utilizing the forward pass and learned LTC-NN weights ω. Hence each forward pass provides an over-determined set of linear equations in the coefficients B, C, and D^js.

The original model coefficients θ are non-linear functions of the coefficients B, C, and D^js, The dense layer is best suited for exploring a large set of possible nonlinear combinations of B, C, and D^js that express θ. An overdetermined system of equations is inconsistent and may be unsolvable. The dense layer guided by the ODE solver induced loss function (ODE Loss) learns a consistent set of linear equations in B, C, and D^js; and it also learns their nonlinear combination to determine θ.

6. Implementation

EMILY is implemented based on the codebase available at Hasani (2024). Here a generic framework for LTC-NN, CT-RNN, and NODE is implemented using tensorflow 2.7.0. We wrote a custom loss function (code available in supplementary document) that implements the Runge-Kutta solution of the physical dynamics given a vector of model coefficients. We use the general training architecture presented in Hasani (2024) with an ADAM optimizer. All code is available in https://github.com/ImpactLabASU/LTC-NN-MR

7. Evaluation strategy

We show the effectiveness of the proposed technique in extracting model coefficients on the benchmark examples and compare with baseline techniques through evaluation experiments. We use the benchmark examples established in Kaiser et al. (2018), which are summarized in Table 2.

7.1. Baseline Strategies

The state-of-art model recovery technique is SINDY-MPC (Kaiser et al., 2018). However, it cannot tackle implicit dynamics. Hence, we extend SINDY-MPC in two ways:
SINDY-MPC-Weakly Implicit (SINDY-MPC-WI): In this approach, in the n-D system if x_j is measured, all other $n-1$ dimensions are replaced by the $n-1$ differentials of x_j. This is also termed as weakly implicit. If $n > 3$ this approach fails since the state variables x_js may not be $n-1$ differentiable. Hence, this method scales poorly with dimensions if $n-1$ dimensions are implicit. In the benchmark examples, since the maximum number of implicit dynamics is 2, this method does not face problems with gradient computation.
SINDY-MPC Augmented with ERA (SINDY-MPC-ERA): In this method, we first use the $k < n$ measured states in a system identification exercise using ERA (Oymak and Ozay, 2021) to extract an approximate n dimensional linear model. The $n-k$ implicit

dynamics are then replaced by the state estimations obtained from simulating the identified n-dimensional linear model. Then SINDY-MPC is used to recover the nonlinear model.
Baselines with Automated Differentiation: Architectures that enable automatic differentiation such as CT-RNN (Hasani et al., 2022), NODE (Chen et al., 2018), or LTC-NN (Hasani et al., 2021) are capable of representing implicit dynamics. We have already shown the versatility of LTC-NN in representing non-linear implicit dynamics. The LTC-NN nodes can be replaced by NODE or CT-RNN and they would still be able to extract implicit dynamics of different forms since the forward pass of NODE and CT-RNN are different from LTC-NN as shown in Eqn. 9. Hence, such architectures are at a relative advantage over sparse identification mechanisms such as SINDy.

$$\text{CT RNN: } \dot{h(t)} = -\frac{h(t)}{\rho} + f_{CT}(h, I, t, \omega_{CT}), \text{ NODE: } \dot{h(t)} = f_{NODE}(h, I, t, \omega_{NODE}). \tag{9}$$

7.2. Evaluation Experiments

We conduct two types of experiments: a) sparse model recovery with full measurability (SMR-FM), where there are no implicit dynamics and b) sparse model recovery with implicit dynamics (SMR-ID), where the variables highlighted in Table 2 are implicit. For the SMR-FM experiment, we do not show the SINDY-MPC extensions and only report the state-of-the-art SINDY-MPC method. We show all baselines for SMR-ID experiments.

7.3. Training and Testing Strategy

We utilize the same training method as used in SINDy-MPC and as reflected in the code accessed from Kaiser (2024). Batch training was utilized for each example. For each example, we took the same simulation data as SINDy-MPC and divided the traces into 48 instance of training and 16 instance of test each of at least $k = 200$ samples. These training instances were passed to the neural architectures with a batch size $S_B = 32$. The $RMSE_Y$ and $RMSE_\Theta$ are reported on the test data. Code is available in the supplementary document.

7.4. Evaluation metrics

For each evaluation experiment, we use the following two metrics:
Root mean square error in model coefficients ($RMSE_\Theta$): Given the estimated model coefficients Θ_{est} for any technique we computed $RMSE_\Theta$ as:

$$RMSE_\Theta = \sqrt{\frac{1}{p}\sum_{j=1\ldots p}(\Theta^j_{est} - \Theta^j)^2}, \; RMSE_Y = \frac{1}{n}\sum_{l=1\ldots n}\sqrt{\frac{1}{k}\times\sum_{j=1\ldots k}(Y^l_{est}(j) - Y^l(j))^2}. \tag{10}$$

Root mean square error in signal ($RMSE_Y$): Given the estimation of the measured variables Y_{est} for any technique we compute $RMSE_Y$ as in Eqn. 10.
Statistical Analysis: For each model recovery example, we first show the recovered model for EMILY. We show the mean model coefficient. For comparison results we show the mean and standard deviation of each metric.

8. Results

In this section, we show the results for the experiments discussed in Section 7.2.

8.1. Recovered Models

The models extracted by EMILY are given below. All coefficients are rounded up to three decimal points after most significant digit. Original models in supplement.
Lotka Volterra : It has two variables x_1 and x_2 given by the following equations:

$$\dot{x_1} = ax_1 - bx_1x_2,\ \dot{x_2} = -cx_2 + dx_1x_2 + u$$

a = 0.5, b = 0.025, c = 0.5, and d = 0.005
Recovered model:

$$\dot{x_1} = 0.52x_1 - 0.026x_1x_2,\ \dot{x_2} = -0.501x_2 + 0.005x_1x_2 + 0.999u$$

Chaotic Lorenz: The chaotic lorenz system is described in the following equations:

$$\dot{x_1} = \sigma(x_2 - x_1) + u, \dot{x_2} = x_1(\rho - x_3) - x_2, \dot{x_3} = x_1x_2 - \beta x_3,$$

$\sigma = 10$, $\beta = 8/3$, $\rho = 28$.
Recovered model:

$$\dot{x_1} = 10.000(x_2 - x_1) + 0.999u,\ \dot{x_2} = 27.992x_1–1.002x_1x_3–0.998x_2,\ \dot{x_3} = 1.000x_1x_2–2.7x_3$$

F8 Cruiser: The F8 Cruser system is given by:

$$\dot{x_1} = -0.9x_1 + x_3 - 0.09x_1x_3 + 0.47x_1^2 - 0.02x_2^2 - x_1^2x_3 + 3.85x_1^3 - 0.21u + 0.28x_1^2u + 0.47x_1u^2 + 0.6u^3$$
$$\dot{x_2} = x_3,\ \dot{x_3} = -4.208x_1 - 0.396x_3 - 0.47x_1^2 - 3.564x_1^3 - 20.967u + 6.265x_1^2u + 46x_1u^2 + 61.1u^3$$

Recovered model:

$$\dot{x_1} = -0.872x_1 + 0.998x_3 - 0.088x_1x_3 + 0.476{x_1}^2 - 0.0186{x_2}^2–0.970{x_1}^2x_3$$
$$+3.849{x_1}^3 - 0.22u + 0.265{x_1}^2u + 0.472x_1u^2 + 0.63u^3,\ \dot{x_2} = 1.000x_3$$
$$\dot{x_3} = -4.210x_1 - 0.399x_3 - 0.465{x_1}^2 - 3.565{x_1}^3 - 20.978u + 6.267{x_1}^2u + 45.711x_1u^2 + 62.002u^3$$

Pathogenics attack model: The pathogenic attack system is given by:

$$\dot{x_1} = \lambda - dx_1 - \beta(1 - \eta u)x_1x_2,\ \dot{x_2} = \beta(1 - \eta u)x_1x_2 - ax_2 - p_1x_4x_2 - p_2x_5x_2$$
$$\dot{x_3} = c_2x_1x_2x_3 - c_2qx_2x_3 - b_2x_3,\ \dot{x_4} = c_1x_2x_4 - b_1x_4,\ \dot{x_5} = c_2qx_2x_3 - hx_5,$$

with $\lambda = 1$, $d = 0.1$, $\beta = 1$, $a = 0.2$, $p_1 = 1$, $p_2 = 1$, $c_1 = 0.03$, $c_2 = 0.06$, $b_1 = 0.1$, $b_2 = 0.01$, $q = 0.5$, $h = 0.1$, and $\eta = 0.9799$.

Recovered model:

$$\dot{x_1} = 0.939 - 0.1x_1 - 0.982x_1x_2 + 0.98ux_1x_2, \dot{x_2} = 0.982x_1x_2 - 0.98ux_1x_2 - 0.18x_2 - 1x_4x_2 - 1.001x_5x_2$$
$$\dot{x_3} = 0.059x_1x_2x_3 - 0.03x_2x_3 - 0.009x_3, \dot{x_4} = 0.029x_2x_4 - 0.1x_4,\ \dot{x_5} = 0.059x_2x_3 - 0.1x_5$$

8.2. SMR-FM experiments

Table 3 shows that all techniques perform similarly when all state variables are measurable. A one-sided t-test revealed no statistically significant differences between the techniques, with all p-values greater than 0.05.

Table 3: SMR-FM: Comparison with full measurability. Value in () is standard deviation.

Example	RMSE	SINDY-MPC	EMILY	CT-RNN-MR	NODE-MR
Lotka	$RMSE_\Theta$	0.059 (0.02)	0.048 (0.015)	0.054 (0.03)	0.064 (0.02)
Volterra	$RMSE_Y$	0.03 (0.02)	0.03 (0.018)	0.05 (0.02)	0.088 (0.03)
Chaotic	$RMSE_\Theta$	0.014 (0.008)	0.015 (0.006)	0.022 (0.009)	0.044 (0.012)
Lorenz	$RMSE_Y$	1.7 (0.6)	1.68 (0.4)	3.66 (1.1)	8.1 (3.6)
F8	$RMSE_\Theta$	7.9 (3.2)	6.8 (2.9)	10.5 (4.8)	19.9 (7.4)
Crusader	$RMSE_Y$	3.2 (2.1)	1.57 (1.4)	3.46 (2.6)	7.22 (5.7)
Pathogenics	$RMSE_\Theta$	0.5 (0.2)	0.39 (0.23)	0.43 (0.3)	0.42 (0.3)
attack	$RMSE_Y$	27.8 (9.1)	28.3 (6.2)	28.8 (7.7)	29.5 (9.6)

Table 4: SMR-ID: Comparison with implicit dynamics. Value in () is standard deviation.

Example	RMSE	SINDY-MPC-WI	SINDY-MPC-ERA	EMILY	CT-RNN-MR	NODE-MR
Lotka	$RMSE_\Theta$	2.1(1.9)	0.2 (0.09)	0.054 (0.026)	0.06 (0.035)	0.065 (0.04)
Volterra	$RMSE_Y$	6.2(4.1) $\times 10^3$	0.99 (0.5)	0.03 (0.021)	0.06 (0.04)	0.09 (0.05)
Chaotic	$RMSE_\Theta$	0.1 (0.09)	0.06 (0.05)	0.016 (0.009)	0.023 (0.01)	0.045 (0.023)
Lorenz	$RMSE_Y$	7.2(2.3) $\times 10^6$	10.1 (4.6)	1.7 (0.6)	3.74 (2.0)	8.23 (5.1)
F8	$RMSE_\Theta$	1051 (204)	21.3 (5.4)	7.81 (4.6)	10.9 (6.2)	21.9 (9.1)
Crusader	$RMSE_Y$	653 (121)	43.2 (21.1)	1.6 (1.2)	3.52 (2.9)	7.75 (6.4)
Pathogenics	$RMSE_\Theta$	23.4 (19.1)	6.3 (4.2)	0.45 (0.21)	0.45 (0.31)	0.49 (0.35)
attack	$RMSE_Y$	134.6 (21.1)	65.1 (19.1)	28.9 (7.8)	29.1 (8.2)	29.9 (10.3)

8.3. SMR-ID experiments

Table 4 shows that our attempts at extending SINDY-MPC to tackle implicit dynamics in recovering sparse dynamics failed. The SINDY-MPC-WI technique performed significantly worse than all other approaches in both model recovery and data fitting tasks. This is primarily because simply adding the derivatives of state variables as additional states may not capture the true model structure. The model may involve implicit state variables that are non-linear combinations of other states, rather than just the direct derivatives of the measured state variables. While the SINDY-MPC-ERA method performed significantly better and is closer to neural architecture-based approaches, it still lags behind them. This is because, although ERA captures a model structure that fits the data well, it may fail to recover the true model structure as it does not enforce the sparsity constraint. The EMILY method worked best among all approaches.

9. Conclusions

In this paper, we showed that EMILY can extract underlying first principle based ODE models of a process from data when some of the ODE state variables are not measured. This is an important step towards making model recovery methods applicable in practical deployments. This can enable online real time explainable digital twin learning. We have not evaluated the runtime of EMILY and is an important future work. While recent works on PINNs and SINDY-MPC have considered weakly implicit dynamics where differentials of state variables are included as unmeasured dynamics, to the best of our knowledge, this is the first approach to recover model coefficients under implicit dynamics.

10. Acknowledgements

The work is partly funded by DARPA AMP-N6600120C4020, DARPA FIRE-P000050426, NSF FDT-Biotech 2436801, and Helmsley Charitable Trust - 2-SRA-2017-503-M-B.

References

Ayan Banerjee and Sandeep K.S. Gupta. Recovering implicit physics model under real-world constraints. In *Proceedings of the 27th European Conference on Artificial Intelligence (ECAI)*, 2024.

Ayan Banerjee, Aranyak Maity, Sandeep K.S. Gupta, and Imane Lamrani. Statistical conformance checking of aviation cyber-physical systems by mining physics guided models. In *2023 IEEE Aerospace Conference*, pages 1–8, 2023. doi: 10.1109/AERO55745.2023.10115613.

Richard N Bergman. Origins and history of the minimal model of glucose regulation. *Frontiers in endocrinology*, 11:583016, 2021.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Zhao Chen, Yang Liu, and Hao Sun. Physics-informed learning of governing equations from scarce data. *Nature communications*, 12(1):6136, 2021.

Urban Fasel, J Nathan Kutz, Bingni W Brunton, and Steven L Brunton. Ensemble-sindy: Robust sparse model discovery in the low-data, high-noise limit, with active learning and control. *Proceedings of the Royal Society A*, 478(2260):20210904, 2022.

K.J. Friston, L. Harrison, and W. Penny. Dynamic causal modelling. *NeuroImage*, 19(4): 1273–1302, 2003. ISSN 1053-8119.

Ramin Hasani. Liquid Time Constant Networks. https://github.com/raminmh/liquid_time_constant_networks, 2024.

Ramin Hasani, Mathias Lechner, Alexander Amini, Daniela Rus, and Radu Grosu. Liquid time-constant networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7657–7666, 2021.

Ramin Hasani, Mathias Lechner, Alexander Amini, Lucas Liebenwein, Aaron Ray, Max Tschaikowski, Gerald Teschl, and Daniela Rus. Closed-form continuous-time neural networks. *Nature Machine Intelligence*, 4(11):992–1003, 2022.

Kadierdan Kaheman, J Nathan Kutz, and Steven L Brunton. Sindy-pi: a robust algorithm for parallel implicit sparse identification of nonlinear dynamics. *Proceedings of the Royal Society A*, 476(2242):20200279, 2020.

Eurika Kaiser. SINDY-MPC: Sparse Identification of Nonlinear Dynamics with Model Predictive Control. https://github.com/eurika-kaiser/SINDY-MPC, 2024.

Eurika Kaiser, J Nathan Kutz, and Steven L Brunton. Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proceedings of the Royal Society A*, 474(2219):20180335, 2018.

Payal Kamboj, Ayan Banerjee, and Sandeep K.S. Gupta. Expert knowledge driven human-ai collaboration for medical imaging: a study on epileptic seizure onset zone identification. *IEEE Transactions on Artificial Intelligence*, pages 1–15, 2024. doi: 10.1109/TAI.2024.3396421.

Imane Lamrani, Ayan Banerjee, and Sandeep KS Gupta. Hymn: mining linear hybrid automata from input output traces of cyber-physical systems. In *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, pages 264–269. IEEE, 2018.

Imane Lamrani, Ayan Banerjee, and Sandeep K. S. Gupta. Operational data-driven feedback for safety evaluation of agent-based cyber–physical systems. *IEEE Transactions on Industrial Informatics*, 17(5):3367–3378, 2021. doi: 10.1109/TII.2020.3009985.

Kookjin Lee, Nathaniel Trask, and Panos Stinis. Machine learning structure preserving brackets for forecasting irreversible processes. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 5696–5707. Curran Associates, Inc., 2021.

Aranyak Maity, Ayan Banerjee, Imane Lamrani, and Sandeep K.S. Gupta. Context aware model learning in cyber physical systems. In *2024 IEEE 7th International Conference on Industrial Cyber-Physical Systems (ICPS)*, pages 1–6, 2024. doi: 10.1109/ICPS59941.2024.10640016.

Samet Oymak and Necmiye Ozay. Revisiting ho–kalman-based system identification: Robustness and finite-sample analysis. *IEEE Transactions on Automatic Control*, 67(4): 1914–1928, 2021.

Markus Quade, Markus Abel, J Nathan Kutz, and Steven L Brunton. Sparse identification of nonlinear dynamics for rapid model recovery. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(6), 2018.

Michael Schmidt and Hod Lipson. Distilling free-form natural laws from experimental data. *science*, 324(5923):81–85, 2009.

Nathalie Verdière, David Manceau, Shousheng Zhu, and Lilianne Denis-Vidal. Inverse problem for a coupling model of reaction-diffusion and ordinary differential equations systems. application to an epidemiological model. *Applied Mathematics and Computation*, 375:125067, 2020. ISSN 0096-3003.