Proceedings of the 14th International Symposium on Process Systems Engineering – PSE 2021+
June 19-23, 2022, Kyoto, Japan
http://dx.doi.org/10.1016/B978-0-323-85159-6.50290-6

Training Stiff Dynamic Process Models via Neural Differential Equations

William Bradley[a], Gabriel S. Gusmão[a], Andrew J. Medford[a], Fani Boukouvala[a]*

[a]*Georgia Institute of Technology, School of Chemical & Biomolecular Engineering, 311 Ferst Dr., Atlanta, GA, 30332, USA*
fani.boukouvala@chbe.gatech.edu

Abstract

A common step in developing generalizable, dynamic mechanistic models is to fit unmeasured parameters to measured data. Fitting differential equation-based models can be computationally expensive due to the presence of nonlinearity and stiffness. This work proposes a two-stage indirect approach where Neural ODEs approximate state derivatives, which are used to estimate the parameters of a differential model. In addition to its computational efficiency, the proposed method demonstrates the ability to work in concert with direct methods to accurately estimate parameters, even in the case of stiff systems. The method is shown here for the training of a microkinetic model.

Keywords: Neural Networks, Parameter Estimation, Stiff ODEs, Neural ODEs.

1. Introduction

The task of finding parameter values of a differential equation (DE) model to explain available experimental data is ubiquitous throughout engineering. The physical meaning of these DE models (also referred to here as a mechanistic model) permit the modeler to predict a system's behavior in unexplored experimental spaces, assuming the parameters have been estimated correctly. However, due to the complexity of DE systems, methods that automate their parameter estimation must often balance efficiency and accuracy. Gradient-based 'direct' methods either rely on repeated integration of the ODEs being regressed, or formulating a constrained nonlinear program discretizing the system of ODEs to solve for the parameter values (Li et al. 2005, Hamilton 2011). Both 'direct' methods face computational tractability issues, which become more severe when the initial parameter estimates are far from the true values, or the ODEs are nonlinear with respect to their parameters. Another problem, common to reaction systems, is the presence of rate terms which vary over large orders of magnitude, resulting in a system with fast and slow dynamics (i.e., at different timescales). Ultimately, to make these regression problems tractable for direct methods, a modeler may need to apply model reduction strategies, ranging from setting tight bounds on parameters to fixing insensitive parameters. Such strategies require domain expertise, which may not be available, as well as user-intervention, preventing automation of the parameter estimation process.

As an alternative to the direct approach, an indirect parameter estimation approach has been proposed, which avoids discretizing the mechanistic model (Swartz and Bremermann 1975, Brunel 2008). In this 2-stage approach, the experimental data is interpolated by a data-driven model, which is differentiated to obtain system derivative estimates. Those derivative estimates combined with state estimates of the interpolating model can be used to estimate the parameters of the mechanistic DEs via nonlinear programming (NLP). The indirect 2-stage approach is so named since it breaks up a single regression problem into two regression problems whose combined computational cost is generally less than that of the direct approaches. Yet despite having the advantage of being computationally cheap, this method is often limited in accuracy due to the difficulty in accurately estimating a system's derivative information.

Recently, we proposed using Neural ODEs (NODEs) as the data-driven surrogate to interpolate measurement data (Figure 1) and for estimating system derivatives (Bradley and Boukouvala 2021). In that work, NODEs compared favorably with other methods for automated parameter estimation of a nonlinear mechanistic DE. However, one class of DEs not covered in that work were those with 'stiff' dynamics. This class of problems can be particularly challenging for parameter estimation methods. One reason for this is the need for numerical methods that balance the number of functional evaluations (i.e., computation) and stability (i.e.. accuracy). Recent work has evaluated numerical techniques for fitting Neural ODEs to stiff system data, and for parameter estimation of stiff systems (Kim et al. 2021), however further work is needed to develop methods that are both general and accurate.

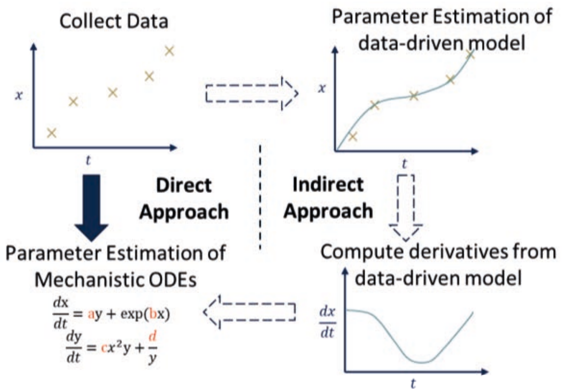

Figure 1. Depiction of the direct vs indirect approach to parameter estimation

2. Methods

In this work, several approaches, and potential combinations thereof, are compared for the parameter estimation of stiff DEs. To start, direct approaches find the parameters p to a mechanistic model $f(x, p)$ by minimizing the following discrepancy function:

$$\min \sum (x_{k,j,meas} - x_{k,j,pred})^2 \quad (1)$$

$$s.t.\ \frac{dx_{k,MM}}{dt} = f(x_k, p) \quad (2)$$

Here, K state variables x_k, where $k = 1, \dots, K$, are measured and predicted at time points j, where $j = 1, \dots, J$, by integrating the mechanistic model (MM) with respect to independent variable t. Though statistically robust, this method can be computationally intensive. For such cases, a 2-stage indirect approach can be attractive.

As illustrated in Figure 1, the 2-stage indirect approach fits the parameters of the mechanistic model by solving 2 separate regression problems. In the first stage, the parameters of the data-driven model are fitted using the original measurement data. In the second stage, the parameters of the mechanistic ODE are found using the state and derivative estimates of the data-driven model. The data-driven model used in our work is a NODE model. This is done by first solving Eq.(1) subject to Eq.(3):

$$s.t.\ \frac{dx_{k,NODE}}{dt} = NN(x_k, w) \quad (3)$$

Neural Network parameters w are fitted to minimize an objective function equal to the sum of squared errors between the model prediction and measured state data. Once the

NODE is trained, derivative estimates are obtained by evaluating the trained NODE at times where measured data was collected using the same process conditions of the measured data. Following the procedure of (Bradley and Boukouvala 2021), we exclude derivative estimates at time t=0, which tend to be less reliable, to improve parameter estimates of the mechanistic DE. For stage two, an NLP is formulated as in Eq.(4) and (5) to find the parameters of the mechanistic DE without integrating the mechanistic DE.

$$\min \sum \left(\frac{dx_{j,k,NODE}}{dt} - \frac{dx_{j,k,MM}}{dt}\right)^2 \tag{4}$$

$$s.t.\ \frac{dx_{j,k,MM}}{dt} = f(x_{j,k,NODE}, p) \tag{5}$$

Depending on the required accuracy, the indirect 2-stage approach may be sufficient for the needs of the model-building problem at hand. However, if increased accuracy is required, we hypothesized a more robust fit would require including the mechanistic model constraints when fitting the measured state data. A tempting option would be a simultaneous approach, which combines the objective functions of the 2-stage approach into a single hybrid objective function:

$$\min \sum (x_{k,j,meas} - x_{k,j,pred})^2 + \lambda \sum \left(\frac{dx_{j,k,NODE}}{dt} - \frac{dx_{j,k,MM}}{dt}\right)^2 \tag{6}$$

Like the indirect approach, Eq.(6) uses the data-driven Neural ODE of Eq.(2) to fit the state data and provide derivative estimates. However, in the hybrid objective both the Neural ODE fit and mismatch between NODE and mechanistic DE are minimized simultaneously, their relative weights controlled by the hyperparameter lambda, λ.

A final alternative to increasing model fidelity is to fit the mechanistic DE directly (i.e., minimize Eq.(1) subject to Eq.(2)). However, as mentioned earlier this incurs an increased compute overhead. In the case of stiff systems, the increased compute cost comes from the finer discretization required to stably integrate the mechanistic model. To reduce compute costs, the direct approach can use parameter estimates informed by the indirect approach. Specifically, the parameters estimated from the 2-stage fitting are used as an initial guess for the DE of Eq.(2). A single application of the indirect followed by the direct approach is herein referred to as the incremental approach.

Throughout this work, we use PyTorch's LBFGS solver and IPOPT within PYOMO as the nonlinear optimizers of the stage 1 and stage 2 regression problems, respectively. For the sake of consistency, the structure of the NODE is fixed to a single hidden layer with tanh activation function and 15 hidden nodes. Further, we assume minimal knowledge of the true parameters prior to model-fitting, and thus all parameters are initialized to the same order of magnitude, specifically a value of 2, for the direct and indirect approaches.

3. Results

To demonstrate the effectiveness of the 2-stage approach, we chose as an example a microkinetic model (MKM) for heterogeneous catalysis (Gusmão et al. 2020). MKMs represent a large class of coupled differential equations which exhibit stiffness due to the presence of both slow and fast rate terms caused by parameter values varying over large orders of magnitude. The MKM system of ODEs governed by a material balance and rate equations are outlined in Figure 2. Table 1 lists the true parameter values.

$$\frac{d[A]}{dt} = -r_1 \qquad \frac{d[C]}{dt} = -r_3 \qquad r_1 = k_3[A][*] - k_4[A*]$$

$$\frac{d[A*]}{dt} = r_1 - r_5 \qquad \frac{d[C*]}{dV} = r_3 + r_5 \qquad r_2 = k_5[B][*] - k_6[B*]$$

$$\frac{d[B]}{dt} = -r_2 \qquad \frac{d[D*]}{dt} = 2r_4 - r_5 \qquad r_3 = k_7[C][*] - k_8[C*]$$

$$\frac{d[B*]}{dt} = r_2 - r_4 \qquad \frac{d[*]}{dt} = -r_1 - r_2 - r_3 - r_4 + r_5 \qquad r_4 = k_{11}[B*][*] - k_{12}[D*]^2$$

$$r_5 = k_{13}[A*][D*] - k_{14}[C*][*]$$

Figure 2. Full MKM system of ODE equations

In this process, gaseous reactants A and B adsorb to a solid surface to form intermediate species before the final product C desorbs into the gas phase. Reactants bound to a catalyst surface site [*] are indicated by an asterisk '*'. All reactions are reversible.

Two datasets were used to represent possible fitting scenarios for the 2-stage approach, each comprising data simulated from two sets of initial conditions. In one dataset, state variables are sampled 15 times logarithmically for each run in the range t = [10e-3, 0.5], amounting to a sample size of 30 datapoints. The second dataset includes the same number of points sampled linearly from time t=0 to t=0.5. At first, the 2-stage approach was applied on the linear dataset. Specifically, the data was used to fit a NODE whose derivatives were then used to solve for the parameters of the mechanistic DE. The parameters found through this approach are compiled in Table 1, column labelled 'Linear Indirect'. Results show that some of the parameters found differ significantly from the true parameters. This is not surprising since data is available only sparsely at earlier times where state values change rapidly due to the stiffness of the system.

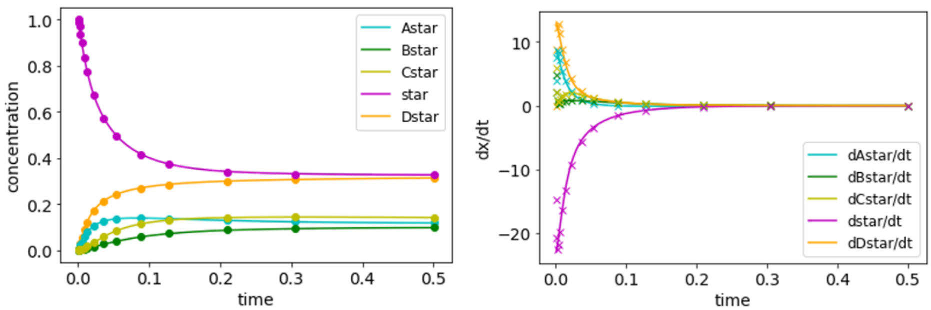

Figure 3. NODE fit (solid lines) to log-sampled data (dots). True derivative shown with x's.

The fitting procedure was repeated with data sampled logarithmically with respect to time. The fit of the Neural ODE to the log-sampled data is presented in Figure 3 for the adsorbed species. Noticeably, despite the NODE fitting the state outputs perfectly (effectively to machine precision), the data-driven model does not capture the exact profile of the derivatives. This result is believed to be due to the inherent flexibility of NODEs, which are not as constrained in outcomes as the simulating mechanistic model. The results of the 2-stage regression including the mean absolute error (MAE) of the fitted model on the log-sampled data are compiled in Table 1 ('Log Indirect' column).

Aiming to improve the accuracy of the fitted mechanistic model, the simultaneous approach was applied using various values for lambda. However, minimizing the hybrid objective function did not result in significantly improved parameter estimates vs the indirect approach, notwithstanding its higher compute cost. This finding was again attributed to the flexibility of NODEs, their being able to interpolate state data despite estimating derivatives that may not exactly match the 'true' derivatives. Due to their low accuracy, results of the simultaneous approach were not included in Table 1.

Instead, the remaining columns in Table 1 include the computational cost and model accuracy from integrating the mechanistic DE during training, either using an uninformed initial guess (i.e., the direct approach) or initializing the mechanistic parameters with the parameters found by the 2-stage methods (i.e., the incremental approach). Figure 4 displays the mechanistic model fit to the linearly-sampled data via the indirect and direct approach. The direct and incremental approaches gave similar simulated trajectories so only the results of the direct method are plotted.

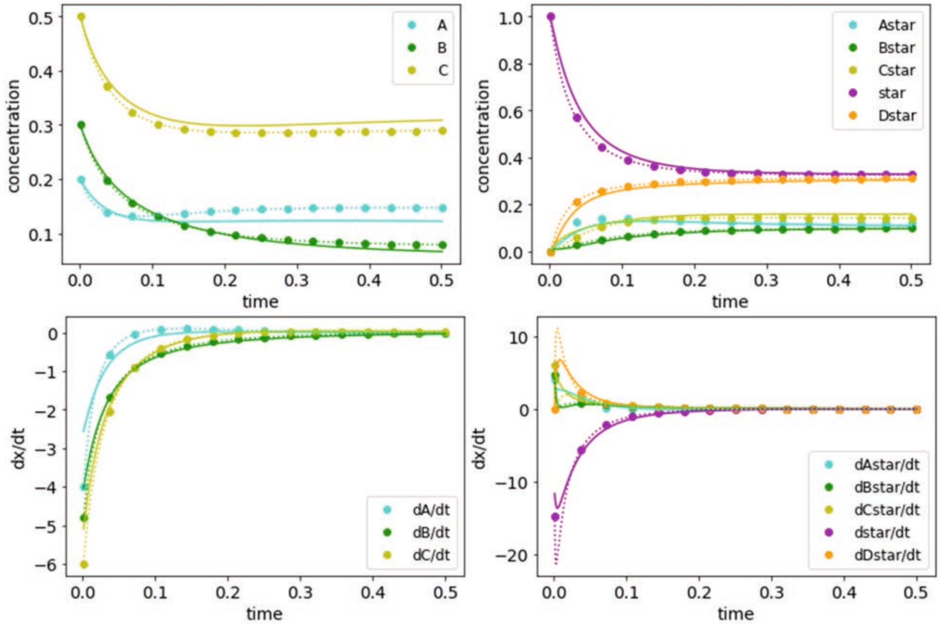

Figure 4. State and derivative estimates of the mechanistic model after parameter estimation via indirect (solid line) and direct approach (dotted line) on linearly-sampled data (solid dots).

A couple trends are worth noting. Firstly, the MAE of the final simulation is lower after using the direct approach, regardless of sampling strategy, indicating increased accuracy can be gained via the direct approach. What's more, applying the incremental approach offers compute savings over the direct approach with an uninformed initial guess, at least for the log-sampled case. However, when fitting the linearly-sampled data, the compute savings from incremental approach are negligible. At least two factors are believed to cause this discrepancy. First, the parameters found through the indirect approach on the linear data were further from the true parameters than for the log-sampled case, offering a poorer initial guess. Second, an increased number of Euler steps were required between datapoints for integrating mechanistic model on the log-sampled data (n=256 vs 56 in the linear case) to avoid divergence issues near the equilibrium region, exacerbating the computational load in the log-sampled case when uninformed initial estimates are used. Ultimately, this indicates that, given a sufficiently sampled experimental space, the incremental approach can merge the direct and indirect approaches in ways that balance both accuracy and efficiency.

Table 1. Table of compute times, parameters estimated, and model errors for different approaches

	True params	Log Indirect	Log Incr.	Log Direct	Linear Indirect	Linear Incr.	Linear Direct
Fit (s) Time	N/A	15.21	137.7	362.31	12.62	74.22	80.94
MAE	N/A	2.46E-3	6.05E-4	4.52E-4	1.45E-2	4.70E-4	2.69E-4
k_3	20	20.46	19.98	19.97	12.82	19.67	19.80
k_4	8	9.061	7.983	7.994	4.556	7.854	7.917
k_5	16	16.50	15.69	15.81	13.49	16.07	15.84
k_6	4	3.490	3.825	3.886	2.637	4.051	3.938
k_7	12	11.38	11.99	12.09	10.19	11.89	11.92
k_8	8	7.695	7.998	8.048	6.719	7.940	7.957
k_{11}	1200	2615	2607	1793	400.4	446.5	1809
k_{12}	400	849.3	871.6	600.8	138.7	147.6	604.0
k_{13}	2000	1672	1662	1117	38.28	2999	1745
k_{14}	1600	1320	1332	890.3	24.05	2401	1395

*Abbreviations: Incr. (Incremental Approach)

4. Conclusions

This work demonstrated a method for accelerating the regression of mechanistic ODEs for stiff systems and evaluated the ability of NODEs to estimate mechanistic ODE parameters with a large magnitude of variability in their true values using different sampling strategies. While the NODE-based incremental approach presents a promising step towards automated parameter estimation of stiff systems, several challenges remain. Neural Networks have limited ability to make predictions that vary over large orders of magnitude, common to stiff systems, which to overcome may require modifying the data-driven model structure to enable greater accuracy. In future, a comparison with incremental and simultaneous methods employing other surrogate models should be performed to assess the Neural ODE’s suitability as a general-purpose DE estimator.

References

Bradley, W. and F. Boukouvala (2021). "Two-Stage Approach to Parameter Estimation of Differential Equations Using Neural ODEs." Industrial & Engineering Chemistry Research.

Brunel, N. J. B. (2008). "Parameter estimation of ODE's via nonparametric estimators." Electron. J. Statist. **2**: 1242-1267.

Gusmão, G. S., A. P. Retnanto, S. C. da Cunha and A. J. Medford (2020). "Kinetics-Informed Neural Networks." arXiv preprint arXiv:2011.14473.

Hamilton, F. (2011). "Parameter Estimation in Differential Equations: A Numerical Study of Shooting Methods." SIAM Undergraduate Research Online **Volume 4**.

Kim, S., W. Ji, S. Deng and C. Rackauckas (2021). "Stiff Neural Ordinary Differential Equations." Chaos **31 9**: 093122.

Li, Z., M. R. Osborne and T. Prvan (2005). "Parameter estimation of ordinary differential equations." IMA Journal of Numerical Analysis **25**(2): 264-285.

Swartz, J. and H. Bremermann (1975). "Discussion of parameter estimation in biological modelling: Algorithms for estimation and evaluation of the estimates." Journal of Mathematical Biology **1**(3): 241-257.