Convergence Analysis of the Discovery of Dynamics via Deep Learning

Yiqi Gu[a] & Qiang Du[b] & Haizhao Yang[c] & Chao Zhou[a]

[a] Department of Mathematics, National University of Singapore

[b] Department of Applied Physics and Applied Mathematics, Columbia University

[c] Department of Mathematics, Purdue University

Contact Information:
Department of Mathematics,
National University of Singapore,
Level 4, Block S17, 10 Lower Kent Ridge Road,
Singapore 119076
Email: `matguy@nus.edu.sg`

Abstract

Identifying dynamics from observed data has always been significant and challenging in a wide range of areas. The combination of linear multistep methods (LMMs) and deep learning [2, 4, 5] is recently successfully employed to discover dynamics, whereas a rigorous convergence analysis of this approach is still missing. In this work, we put forward an error estimate for the deep network-based LMMs using the approximation property of Floor-ReLU networks [3]. It indicates for certain families of LMMs, the l^2 grid error is of $O(h^p)$ where h is the time step size and p is the local truncation error order, as long as the network size is sufficiently large. Moreover, the numerical results of some physically relevant examples are consistent with our theory.

1 Linear Multistep Methods

Consider the following dynamical system with an initial condition

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}(t) = \boldsymbol{f}(\boldsymbol{x}(t)), \quad 0 < t < T, \tag{1}$$

$$\boldsymbol{x}(0) = \boldsymbol{x}_{\text{init}}, \tag{2}$$

The linear M-multistep method is widely utilized in solving dynamical systems. Let $N > 0$, $h = T/N$ and $t_n = nh$ for $n = 0, 1, \cdots, N$. The goal is to compute $\boldsymbol{x}_n \approx \boldsymbol{x}(t_n)$. Suppose $\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_{M-1}$ are given states, then $\boldsymbol{x}_n$ for $n = M, M+1, \cdots, N$ can be computed by the following linear M-multistep scheme,

$$\sum_{m=0}^{M} \alpha_m \boldsymbol{x}_{n-m} = h \sum_{m=0}^{M} \beta_m \boldsymbol{f}(\boldsymbol{x}_{n-m}), \quad n = M, M+1, \cdots, N, \tag{3}$$

Define the local truncation error $\boldsymbol{\tau}_{h,n}$ as

$$\boldsymbol{\tau}_{h,n} = \frac{1}{h}\sum_{m=0}^{M} \alpha_m \boldsymbol{x}(t_{n-m}) - \sum_{m=0}^{M} \beta_m \boldsymbol{f}(\boldsymbol{x}(t_{n-m})), \tag{4}$$

for $n = M, M+1, \cdots, N$. A LMM is said to have order p if

$$\max_{M \le n \le N} \|\boldsymbol{\tau}_{h,n}\|_\infty = O(h^p), \quad \text{as } h \to 0. \tag{5}$$

Common LMM schemes include Adams-Bashforth (A-B), Adams-Moulton (A-M), and Backwards Differentiation Formula (BDF) schemes.

2 Discovery of Dynamics

Let $\boldsymbol{x}(t) \in C^\infty([0,T])^d$ and $\boldsymbol{f}(\boldsymbol{z}) : \mathbb{R}^d \to \mathbb{R}^d$ be two vector-valued functions satisfying the following dynamics

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}(t) = \boldsymbol{f}(\boldsymbol{x}(t)), \quad 0 < t < T. \tag{6}$$

Here both $\boldsymbol{x}(t)$ and $\boldsymbol{f}(\boldsymbol{z})$ are unknown. Now given $\boldsymbol{x}_n = \boldsymbol{x}(t_n)$ for $n = 0, \cdots, N$, the objective is to determine $\boldsymbol{f}(\boldsymbol{z})$, i.e. to find a closed-form expression for $\boldsymbol{f}(\boldsymbol{z})$ or to evaluate $\boldsymbol{f}(\boldsymbol{x}_i)$ for all i.

One effective approach is to build a discrete relation between $\boldsymbol{x}_i$ and $\boldsymbol{f}(\boldsymbol{x}_i)$ by LMMs [1], namely,

$$\sum_{m=0}^{M} \alpha_m \boldsymbol{x}_{n-m} = h \sum_{m=0}^{M} \beta_m \boldsymbol{f}_{n-m}, \quad n = M, M+1, \cdots, N, \tag{7}$$

where $\boldsymbol{f}_i \in \mathbb{R}^d$ is an approximation of $\boldsymbol{f}(\boldsymbol{x}_i)$.

Since the linear system (7) might have less equations than unknowns (differ by N_a), We need introduce auxiliary conditions to make it uniquely solvable. Assume the LMM has order p, one way is to compute the initial N_a unknowns by one-sided FDM of order p, i.e.,

$$f_i = \frac{1}{h}\sum_{m=0}^{p} \gamma_m x_{i+m}, \quad i = 0, 1, \cdots, N_a - 1, \tag{8}$$

Combing (7) and (8) leads to a augmented linear system

$$\boldsymbol{A}_h \vec{\boldsymbol{f}}_h = \vec{\boldsymbol{b}}_h, \tag{9}$$

3 Neural Network Approximation

We introduce a network $\hat{f}(\boldsymbol{z}) \in \mathcal{N}_{\hat{\mathcal{M}}}$ to approximate $f(\boldsymbol{z})$, an arbitrary component of $\boldsymbol{f}(\boldsymbol{z})$. Then it is expected from (7) and (8)

$$\sum_{m=0}^{M} \alpha_m x_{n-m} = h \sum_{m=0}^{M} \beta_m \hat{f}(\boldsymbol{x}_{n-m}), \quad n = M, M+1, \cdots, N, \tag{10}$$

and

$$\hat{f}(\boldsymbol{x}_i = \frac{1}{h}\sum_{m=0}^{p} \gamma_m x_{i+m}, \quad i = 0, 1, \cdots, N_a - 1, \tag{11}$$

where $\boldsymbol{x}_n$ for $n = 0, \cdots, N$ are given data.

Under the deep learning framework, we need to solve the optimization: find

$$J_{a,h}(\hat{f}_{\hat{\mathcal{M}}}) = \min_{\hat{u} \in \mathcal{N}_{\hat{\mathcal{M}}}} J_{a,h}(\hat{u}), \tag{12}$$

where

$$J_{a,h}(\hat{u}) := \frac{1}{N}\Big(\sum_{i=0}^{N_a-1}\left|\hat{u}(\boldsymbol{x}_i) - \frac{1}{h}\sum_{m=0}^{p}\gamma_m x_{i+m}\right|^2 + \sum_{n=M}^{N}\left|\sum_{m=0}^{M}\beta_m \hat{u}(\boldsymbol{x}_{n-m}) - \sum_{m=0}^{M} h^{-1}\alpha_m x_{n-m}\right|^2\Big). \tag{13}$$

4 Convergence

Define the l^2 grid norm given $h > 0$,

$$|f|_{2,h} := \left(\frac{1}{N+1}\sum_{i=0}^{N} |f(\boldsymbol{x}_i)|^2\right)^{\frac{1}{2}}, \forall f \in C(\mathbb{R}^d). \tag{14}$$

Theorem *Suppose $\boldsymbol{x} \in C^\infty([0,T])^d$ and $\boldsymbol{f} \in C(\mathbb{R}^d)^d$ are related by (6). Let f be an arbitrary component of $\boldsymbol{f}$. Also, suppose $\boldsymbol{x}_n = \boldsymbol{x}(t_n)$ for $n = 0, \cdots, N$ are prescribed. Then for any $h > 0$, there exist $J_0, K_0 \in \mathbb{N}^+$ such that for all $J > J_0$, $K > K_0$,*

$$\left|\hat{f}_{\hat{\mathcal{M}},h} - f\right|_{2,h} < C\kappa_2(\boldsymbol{A}_h)h^p, \tag{15}$$

where C is a constant independent of h; $\hat{f}_{\hat{\mathcal{M}},h} \in \mathcal{N}_{\hat{\mathcal{M}}}$ is a minimizer of $J_{a,h}$ defined by (13) corresponding to a LMM with order p, and the admissible set $\mathcal{N}_{\hat{\mathcal{M}}}$ consists of all Floor-ReLU networks with sizes $\hat{\mathcal{M}} = \{64dK + 3, \max\{d, 5J + 13\}\}$.

Specifically, if $\kappa_2(\boldsymbol{A}_h)$ is uniformly bounded for all $h > 0$, then

$$\lim_{J,K\to\infty, h\to 0} \left|\hat{f}_{\hat{\mathcal{M}},h} - f\right|_{2,h} = 0. \tag{16}$$

It can be shown for A-B schemes of $1 \le N \le 6$ and BDF schemes of all M, $\kappa_2(\boldsymbol{A}_h)$ is uniformly bounded.

5 Experiments

5.1 Problem with Accurate Data

The first example is following model problem

$$\begin{cases} \dot{x_1} = x_2, \\ \dot{x_2} = -x_1, \\ \dot{x_3} = 1/x_2^2, \\ [x_1, x_2, x_3]_{t=0} = [0, 1, 0], \end{cases} \quad t \in [0, 1], \tag{17}$$

whose states can be explicitly given by $x_1 = \sin(t)$, $x_2 = \cos(t)$, $x_3 = \tan(t)$. The training error (grid error) and testing error versus width W for various depth L are presented in Fig. 1. The error decay v.s. h are presented for various families of LMMs in Figure 2.

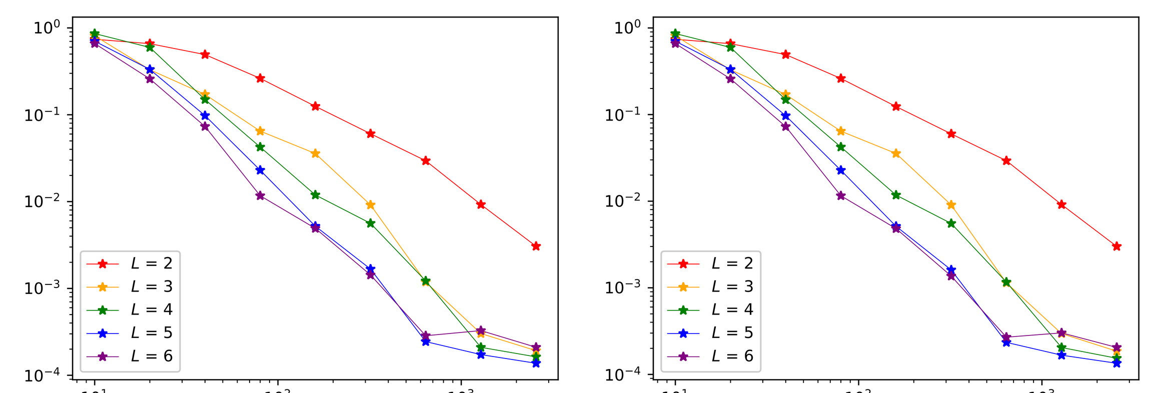

Figure 1: *Errors versus W in the dynamics discovery of (17). Left: training error; Right: testing error.*

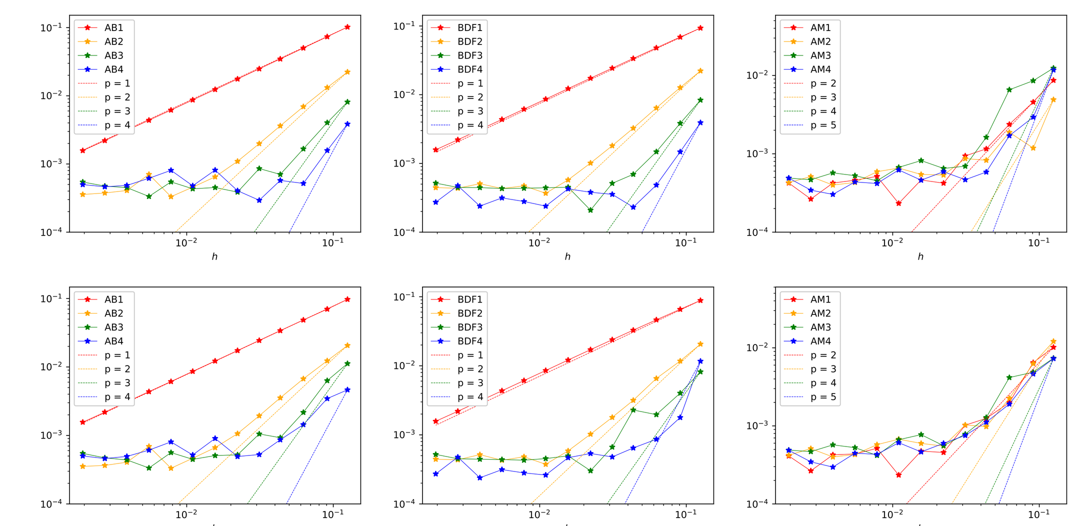

Figure 2: Errors versus h in the dynamics discovery of (17). Left: training error; Right: testing error.

5.2 Lorenz System

The second example is following Lorenz system

$$\begin{cases} \dot{x_1} = 10(x_2 - x_1), \\ \dot{x_2} = x_1(28 - x_3) - x_2, \\ \dot{x_3} = x_1 x_2 - 8x_3/3, \\ [x_1, x_2, x_3]_{t=0} = [-8, 7, 27], \end{cases} \quad t \in [0, 1], \tag{18}$$

The error decay v.s. h are presented for various families of LMMs in Figure 3. The dynamics of the true governing function and the approximate neural network are also presented in Figure 4, from which we observe the neural network can identify the chaotic dynamics effectively.

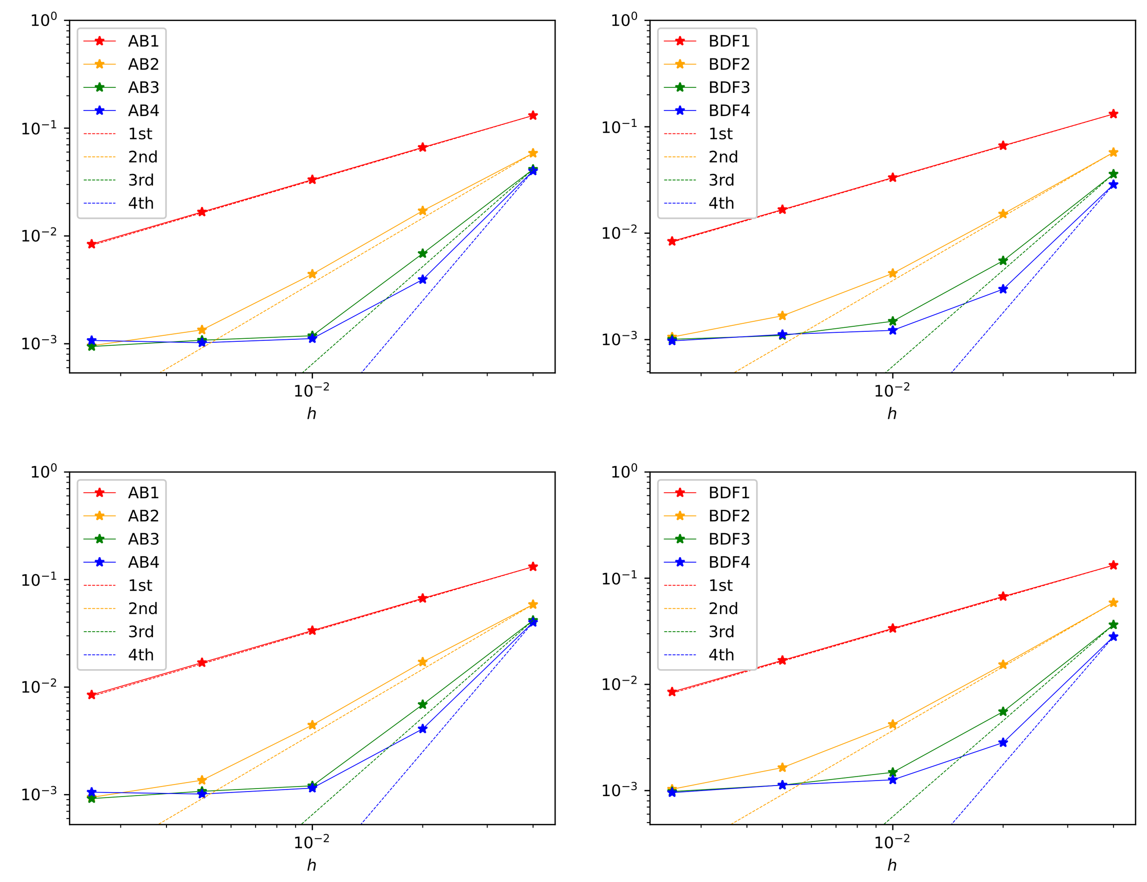

Figure 3: Errors versus h in the dynamics discovery of (18). Left: training error; Right: testing error.

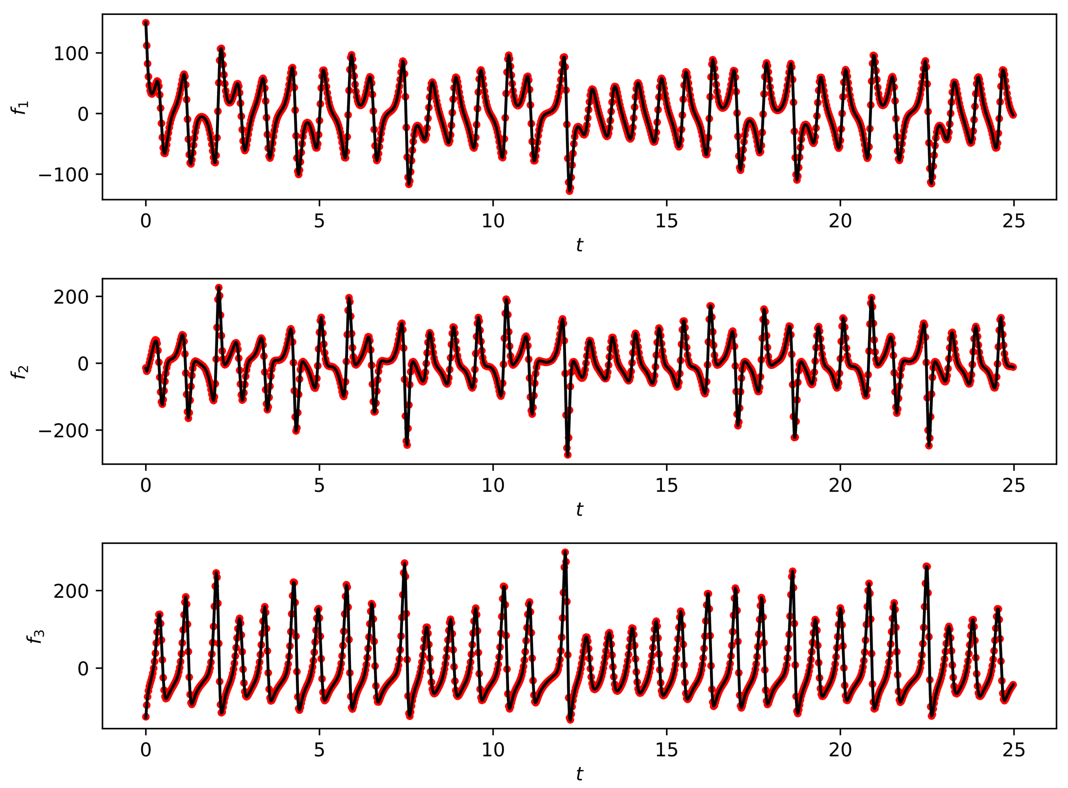

Figure 4: The true governing function (black) and the approximate neural network (red) in the dynamics discovery of (18).

References

[1] R. Keller and Q. Du. Discovery of Dynamics using Linear Multistep Methods. *arXiv e-prints*, arXiv:1912.12728, 2019.

[2] M. Raissi, P. Perdikaris, and G. E. Karniadakis. Multistep neural networks for data-driven discovery of nonlinear dynamical systems. *arXiv e-prints*, arXiv:1801.01236, 2018.

[3] Z. Shen, H. Yang, and S. Zhang. Deep network with approximation error being reciprocal of width to power of square root of depth. *arXiv e-prints*, arXiv:2006.12231, 2020.

[4] R. Tipireddy, P. Perdikaris, P. Stinis, and A. Tartakovsky. A comparative study of physics-informed neural network models for learning unknown dynamics and constitutive relations. *arXiv e-prints*, arXiv:1904.04058, 2019.

[5] X. Xie, G. Zhang, and C. G. Webster. Non-intrusive inference reduced order model for fluids using deep multistep neural network. *Mathematics*, 7(8), 2019.