Dynamic Function Learning through Control of Ensemble Systems

Wei Zhang,[1] **Vignesh Narayanan,** [2] **Jr-Shin Li** [1]

[1] Washington University in St. Louis
[2] University of South Carolina
wei.zhang@wustl.edu, vignar@sc.edu, jsli@wustl.edu

Abstract

Learning tasks involving function approximation are prevalent in numerous domains of science and engineering. The underlying idea is to design a learning algorithm that generates a sequence of functions converging to the desired target function with arbitrary accuracy by using the available data samples. In this paper, we present a novel interpretation of iterative function learning through the lens of ensemble dynamical systems, with an emphasis on establishing the equivalence between convergence of function learning algorithms and asymptotic behavior of ensemble systems. In particular, given a set of observation data in a function learning task, we prove that the procedure of generating an approximation sequence can be represented as a steering problem of a dynamic ensemble system defined on a function space. This in turn gives rise to an ensemble systems-theoretic approach to the design of "continuous-time" function learning algorithms, which have a great potential to reach better generalizability compared with classical "discrete-time" learning algorithms.

Introduction

Learning a function from measurement data is a common task prevalent across diverse domains of science and engineering. Typical applications include system identification (Schoukens and Ljung 2019; Ljung 1999; Narendra and Kannan 1990), reinforcement learning (Sutton and Barto 1998; Bertsekas et al. 2000; Doya 2000), classification (Abu-Mostafa, Magdon-Ismail, and Lin 2012; Goodfellow, Bengio, and Courville 2016a), and model learning (inverse) problems (Tarantola 2005). The key idea behind the many existing techniques to accomplishing these tasks is to generate a sequence of function estimates from an initial guess that tend toward the target function using observation data. The primary challenge to efficiently solve this problem concerns with the design of update rules that generate a convergent sequence of functions such that the target function forms the limit point of this sequence. In general, the initial guess of the sequence and the observed data (both quality and quantity) play a significant role to warrant convergence of the sequence to the desired target.

Our Contributions In this work, we study a fundamental question on function approximation via iterative algorithms from a systems-theoretic viewpoint. Namely, can we formally represent an iterative learning algorithm as a dynamical system? What are the necessary and sufficient conditions for such an iterative learning algorithm to converge in terms of the associated dynamical system representations? We shall answer these highly non-trivial questions by making the following contributions. We show that for every iterative learning algorithm generating a convergent sequence of functions (converging to a target function), there exists an *ensemble control system* defined on the function space with an equilibrium point at the target function and vice versa. Moreover, we show that the convergence of the learning algorithm is guaranteed, regardless of the initial guess, when the associated ensemble control system is ensemble controllable. In addition, examples are provided to numerically verify the conclusion, which further demonstrates the applicability of leveraging the proposed ensemble-system theoretic approach to designing "continuous-time" function learning algorithms with better generalizability comparing with classical "discrete-time" algorithms.

Related works

In this paper, we study the function learning problems, and in particular, draw a parallel between the dynamics induced by iterative function learning task and the propagation of an inhomogeneous ensemble dynamical system. An (inhomogeneous) ensemble system is a parameterized family of dynamical systems evolving on a function space. Fundamental investigations of ensemble dynamical systems and their properties, e.g., controllability and observability, have been conducted in series of works over the last two decades (Li 2006, 2011; Zeng et al. 2016; Chen 2020; Narayanan, Zhang, and Li 2020; Li, Zhang, and Tie 2020)). Specifically, *ensemble controllability* of both time-varying and time-invariant linear (Li 2011; Li, Zhang, and Tie 2020; Zeng and Allgöewer 2016), bilinear (Li and Khaneja 2009; Zhang and Li 2021), and some classes of nonlinear ensemble systems (Kuritz, Zeng, and Allgöwer 2018; Li, Dasanayake, and Ruths 2013) have been studied in the literature.

Recently, there has been a renewed interest in the use of techniques from dynamical systems and control to gain insights into the commonly encountered learning problems

and to synthesize new tools to tackle them. For instance, the connection between control systems and certain classes of computational neural networks have been studied in (Weinan 2017; Haber and Ruthotto 2017; Lu et al. 2018; He et al. 2016). In particular, these developments view the common learning problems, such as weight identifiability from data (Albertini and Sontag 1993), controllability (Sontag and Sussmann 1997; Sontag and Qiao 1999), and stability (Michel, Farrell, and Porod 1989; Hirsch 1989) of neural networks, from a dynamical system viewpoint. In this context, more recently, function approximation problems and the concept of *universality* of a class of deep residual networks were analyzed through the lens of homogeneous dynamic ensembles (Tabuada and Gharesifard 2020; Agrachev and Sarychev 2020). Different from the works presented earlier (Tabuada and Gharesifard 2020; Agrachev and Sarychev 2020), we introduce a new notion interpreting the evolution of an iterative function learning algorithm as the time propagation of an inhomogeneous ensemble system, and the convergence toward a desired function is interpreted as a steering problem of an ensemble system. Specifically, we illustrate that the dynamic properties of the learning algorithm can be studied and analyzed through a dynamically equivalent systems, an inhomogeneous ensemble system.

Ensemble Systems and Ensemble Control

An *ensemble system* is a parameterized family of dynamical systems evolving on a common manifold $M \subseteq \mathbb{R}^n$ of the form

$$\frac{d}{dt}x(t,\beta) = f(x(t,\beta),\beta,u(t)), \tag{1}$$

where the system parameter β takes values on $\Omega \subseteq \mathbb{R}^d$, $u(t) \in \mathbb{R}^m$ is the control input, and $f(\cdot,\beta,u(t))$ is a vector field on M for each fixed $\beta \in \Omega$ and u. A canonical *ensemble control* task is to design a β-independent control input $u(t)$ that steers the whole family of systems from an initial profile $x_0(\beta) = x(0,\beta)$ to a desired final profile $x_F(\beta)$ for all β. By regarding the state variable $x(t,\beta)$ as a function of β, the ensemble system in (1) can be considered as a single dynamical system evolving on a space of M-valued functions defined on Ω, denoted by $\mathcal{F}(\Omega, M)$.

Ensemble Controllability

Controllability is one of the most fundamental properties of a dynamical system, which characterizes the ability of the control input to precisely steer a control system between any two given points in the state-space. For an ensemble system as in (1), the parameter space Ω is generally an infinite set so that the state-space $\mathcal{F}(\Omega, M)$ is an infinite-dimensional manifold; or, in another words, the system is an infinite-dimensional system. For such a system, the classical notion of controllability, i.e., exact controllability of steering the ensemble system between two functions in $\mathcal{F}(\Omega, M)$, can be too restrictive. Hence, we introduce the concept of ensemble controllability to characterize the ability to control an ensemble system in the approximation sense.

Definition 1 (Ensemble controllability) *The system in* (1) *is said to be* ensemble controllable *on the function space* $\mathcal{F}(\Omega, M)$ *if for any* $\varepsilon > 0$ *and starting with any initial profile* $x_0 \in \mathcal{F}(\Omega, M)$*, there exist a time* $T > 0$ *and a control law* $u : [0,T] \to \mathbb{R}^m$ *that steers the system into an* ε*-neighborhood of a desired target profile* $x_F \in \mathcal{F}(\Omega, M)$*, i.e.,* $d(x(T,\cdot), x_F(\cdot)) < \varepsilon$*, where* $d : \mathcal{F}(\Omega, M) \times \mathcal{F}(\Omega, M) \to \mathbb{R}$ *is a metric on* $\mathcal{F}(\Omega, M)$.

Definition 1 shows that ensemble controllability is a notion of approximate controllability, in which the final time T may depend on the approximation accuracy ε.

Remark 1 (Ensemble controllability and convergence) *Ensemble controllability further conveys the idea of function convergence, namely,* $x(T,\cdot) \to x_F(\cdot)$ *as* T *is sufficiently large. This is essentially a continuous-time analogue to the convergence of a sequence of functions, e.g., generated by a learning algorithm.*

Inspired by the observation in Remark 1, in the rest of the paper, our goal is to rigorously establish and characterize the relationship between the process of generating a sequence of functions via an iterative learning algorithm and the propogation of an ensemble system. Meanwhile, we will also investigate the role of control inputs and ensemble controllability of ensemble systems played in the design and convergence analysis of function learning algorithms.

Function Learning from an Ensemble Control Viewpoint

The focus of this section is to formulate function learning as an ensemble control problem. We will develop a universal framework to transform the design of function learning algorithms to that of ensemble control laws, and rigorously establish the equivalence between convergence of learning algorithms and ensemble controllability.

Ensemble systems adapted to function learning algorithms

The most widely-used approach to learning a function is to generate a sequence of functions converging to it. By treating the index of the sequence as time, it is natural to assume that the generation of the sequence follows the time-evolution of some dynamical system. In addition, because each term in the sequence is a function, the associated dynamical system is necessarily an ensemble system evolving on the same function space as the sequence does.

To rigorously bridge function learning algorithms and ensemble systems, let $h : \Omega \to \mathbb{R}^n$ denote the function to be learned and $h_0 : \Omega \to \mathbb{R}^n$ be the initial guess of h, then the algorithm learning h can be represented by the iterative formula

$$h_{k+1} = h_k + \Delta h_k, \quad k \in \mathbb{N}, \tag{2}$$

where $\mathbb{N}$ denotes the set of nonnegative integers and $\Delta h_k : \Omega \to \mathbb{R}^n$ is the update rule at the k^{th} iteration, generally depending on the gradient of h_k, for each $k \in \mathbb{N}$. Because $\mathcal{F}(\Omega, \mathbb{R}^n)$ is a vector space, it is possible to define a norm $\|\cdot\|$ on it for the convergence analysis of the algorithm, i.e., $h_k \to h$ if and only if $\|h_k - h\| \to 0$.

Next, to acquire an ensemble systems-theoretic understanding of the function learning algorithm with the iterative relation described in (2), we think of the iterations as the Euler discretization of an unforced ensemble system defined on $\mathcal{F}(\Omega, \mathbb{R}^n)$ of the form as in (1), that is,

$$\frac{d}{dt}x(t,\beta) = f(x(t,\beta),\beta). \tag{3}$$

For convenience, we also use x_t to denote the state variable $x(t,\cdot)$ of the system. To be adapted to the iterative rule in (2), the ensemble system in (3) is required to satisfy $x_0 = h_0$ and $(t_{k+1} - t_k)f(x_{t_k},\cdot) = \Delta h_k$ for a sequence of time t_k such that $t_k \to \infty$ as $k \to \infty$. In this case, we say the ensemble system in (3) is *adapted* to the learning algorithm in (2). Moreover, we also assume that $f : \mathbb{R}^n \times \Omega \to \mathbb{R}^n$ is a smooth function (in both arguments), which, for example, can be guaranteed by the condition that $\Delta h_k : \Omega \to \mathbb{R}^n$ are smooth functions for all $k \in \mathbb{N}$.

Undoubtedly, one of the most crucial criteria for evaluating the performance of a learning algorithm is convergence. As shown in the following proposition, the transformation of an iterative learning algorithm in (2) to the adapted ensemble system in (3) also carries over the convergence analysis of the algorithm to the stability analysis of the system.

Proposition 1 *If the sequence of functions $\{h_k\}_{k\in\mathbb{N}}$ in $\mathcal{F}(\Omega, \mathbb{R}^n)$ generated by the learning algorithm in* (2) *converges to a function $h \in \mathcal{F}(\Omega, \mathbb{R}^n)$, then there is an ensemble system in the form of* (3) *defined on $\mathcal{F}(\Omega, \mathbb{R}^n)$ adapted to this learning algorithm such that it has an equilibrium point at h.*

Proof. See Appendix for the complete proof. The main idea is to construct an ensemble system defined on $\mathcal{F}(\Omega, \mathbb{R}^n)$ in the form of (3) whose sampled trajectory approximates the "tail" of the sequence $\{h_k\}_{k\in\mathbb{N}}$ with arbitrary accuracy so that the system stabilizes to h. □

Remark 2 *We note that Proposition 1 only demonstrates the existence of an ensemble system being able to stabilize at the limit point of the sequence generated by the learning algorithm, and it by no means indicates that every ensemble system adapted to the same algorithm has this property.*

Dynamic function learning via ensemble systems

The association of stable ensemble systems to convergent function learning algorithms discussed above takes the first step towards the goal of understanding function learning problems through the lens of ensemble systems theory. In this section, we will reverse the engineering to generate function learning algorithms by using ensemble systems.

To be more specific, given a function learning task and an ensemble system as in (3), we would like to know whether the iterative algorithm generated by the Euler discretization of the ensemble system is able to accomplish the learning task. According to Proposition 1, it is necessary that the ensemble system has an equilibrium point at the function to be learned. However, this is not sufficient to guarantee the convergence of the learning algorithm generated by the ensemble system to the desired function. Additionally, we also need to make sure the initial guess to be accurate enough in the sense of lying in the region of attraction of the equilibrium point. These two conditions together then give rise to a converse of Proposition 1 as follows.

Proposition 2 *Consider an ensemble system defined on the function space $\mathcal{F}(\Omega, \mathbb{R}^n)$ as in* (3)*. If $h \in \mathcal{F}(\Omega, \mathbb{R}^n)$ is a stable equilibrium point of the system and $h_0 \in \mathcal{F}(\Omega, \mathbb{R}^n)$ is in the region of attraction of h, then there is a function learning algorithm generated by the ensemble system which converges to h.*

Proof. This directly follows from the definition of stable equilibrium points of dynamical systems, and the detail is in Appendix. □

Propositions 1 and 2 give a necessary and sufficient condition for convergence of function learning algorithms in terms of stability of the adapted ensemble systems. The requirement for the adapted ensemble systems to have stable equilibrium points at the desired functions imposes strong restrictions on their system dynamics. On the other hand, the need for the initial guesses to be in the regions of attraction of the equilibrium points may lead to sensitivity of the learning algorithms generated by these ensemble systems to the initial guesses. To waive these requirements, it is inevitable to force such ensemble systems by external control inputs.

In the presence of a control input, e.g., as the ensemble system in (1), the function learning algorithm in (2) generated by the ensemble system, more specifically the updating rule Δh_k, also depends on $u(t)$. As a result, it is possible to design an appropriate $u(t)$ to enforce the convergence of learning algorithm to the desired function h, even though h may not be an equilibrium point of the uncontrolled system.

Theorem 1 *Given an ensemble control system defined on the function space $\mathcal{F}(\Omega, \mathbb{R}^n)$ as in* (1)*. Then, for any $h \in \mathcal{F}(\Omega, \mathbb{R}^n)$, there is a function learning algorithm generated by the ensemble system converging to h regardless of the initial guess if and only if the system is ensemble controllable on $\mathcal{F}(\Omega, \mathbb{R}^n)$.*

Proof. The idea is to interpret the concept of ensemble controllability in terms of convergence. See Appendix for details. □

Conceptually, Theorem 1 demonstrates the potential for a novel function learning algorithm design method using ensemble control theory. Moreover, it is worth noting that even the system is ensemble uncontrollable, it may still be able to generate a learning algorithm accomplishing the desired function learning task. However, in this case, it is natural that the initial guess is pivotal to the algorithm convergence.

Corollary 1 *For any function $h \in \mathcal{F}(\Omega, \mathbb{R}^n)$, if the initial guess $h_0 \in \mathcal{F}(\Omega, \mathbb{R}^n)$ of h is in the controllable submanifold of the ensemble system in* (1) *containing h, then there is a function learning algorithm as in* (2)*, generated by the ensemble system, converging to h.*

Proof. Because any ensemble system is ensemble controllable on its controllable submanifold, the proof directly follows from Theorem 1 by restricting the ensemble system in (1) to the controllable submanifold containing h and h_0. □

Remark 3 (Robustness to initial guesses) *Theorem 1 and Corollary 1 presented a distinctive feature of the function learning algorithms generated by ensemble control systems, that is, the robustness to initial guesses. With the ability to manipulate the "algorithm dynamics" using a control input, initial guesses are no longer required to be close to the desired function. In particular, under the condition of ensemble controllability, the learning algorithm converges globally; otherwise, it is sufficient to set the initial guess on the same controllable submanifold as the target function.*

On the other hand, Theorem 1 and Corollary 1 also indicate that the function learning algorithm design problem can be formulated as an ensemble control problem, which can be tackled by various well-developed methods, such as pseudospectral (Li et al. 2011) and iterative linearization methods (Wang and Li 2018; Zeng 2019).

Dynamic function learning for parameterized models

In practice, to learn a function $h : \Omega \to \mathbb{R}^n$, it is highly inefficient, or even impractical, to search for the entire space of functions from Ω to $\mathbb{R}^n$. Fortunately, with some prior knowledge about h, it is possible to focus the learning on a subspace of this function space. Of particular interest, it is common to consider the case where functions in this subspace can be indexed by parameters taking values in a set Θ. Consequently, the function learning problem can be formulated as the search of an element $\theta \in \Theta$ such that the function indexed by θ best approximates h in the sense of minimizing a loss function $L : \Theta \to \mathbb{R}$ at θ. In this case, the learning algorithm in (2) reduces to

$$\theta_{k+1} = \theta_k - \alpha_k \nabla L(\theta_k), \tag{4}$$

where $\alpha_k \in \mathbb{R}$ is the learning rate for the k^{th} iteration and ∇L denotes the gradient of L. Then, the dynamical system adapted to (4) is of the form,

$$\frac{d}{dt}\theta(t) = -\nabla L(\theta(t)), \tag{5}$$

defined on Θ. Note that the equilibrium points of the system in (5) are exactly the critical points of the function L. Moreover, an equilibrium point θ^* is stable if and only if the Hessian matrix $-\nabla^2 L(\theta^*)$ is negative-definite, and this observation reveals that the local minima of L are composed of the set of stable equilibrium points. Then, provided that the system in (5) has no hyperbolic equilibrium point, equivalently, the function L does not have any saddle point, Proposition 2 guarantees the convergence of any trajectory of the system to a local minimum of L.

In practice, it is common to minimize L under some penalties R_i, $i = 1, \ldots, m$, e.g., for improving the generalizability of the learning algorithm (Goodfellow, Bengio, and Courville 2016b). In this case, we can choose the adapted ensemble system to be in the control-affine form as

$$\frac{d}{dt}\theta(t) = -\nabla L(\theta(t)) + \sum_{i=1}^{m} u_i(t) \nabla R_i(\theta(t)), \tag{6}$$

in which $u_1, \ldots, u_m$ are the control inputs representing the weights of the penalties. This further leads to a unique advantage of formulating a learning algorithm as a dynamical system as in (6), that is, real-time tuning of the weights of the penalty terms for the algorithm to achieve better performance. Furthermore, as a special case of the general ensemble system in (1), the equivalence between convergence of learning algorithms and controllability also holds for the system in (6).

Remark 4 *Note that, as in* (6)*, the gradient of the cost function represents the natural drift of the system, which is inherent to many of the existing gradient-based learning approaches. However, what is interesting is the role of the control vector fields play as regulators or exploratory signals. Contrary to regular learning algorithms in which these terms combat with gradient terms resulting in some sacrifice for algorithm performance, our results reveal that they, serving as control vector fields, tend to make the ensemble system adapted to the learning algorithm controllable, which in turn leads to global convergence of the algorithm. Geometrically, with these penalties, the ensemble system, equivalently, the learning algorithm, can reach more functions (any functions if controllable), in addition to those along the gradient direction of the cost function.*

Examples and Simulations

In this section, we use the curve fitting problem to illustrate the applicability of the proposed ensemble control-theoretic function learning approach. In particular, to verify the proposed approach, we compare it with classical function learning methods, such as linear regression.

Curve fitting

The purpose of curve fitting is to find a function having the best fit to an input-output dataset. Let $X = \{x_1, \ldots, x_N\} \subset \mathbb{R}^n$ and $Y = \{y_1, \ldots, y_N\} \subset \mathbb{R}$ be the input and output data, respectively, and $\mathcal{F}$ denote the space containing functions from $\mathbb{R}^n$ to $\mathbb{R}$, then the curve fitting problem can be formulated as $\min_{h\in\mathcal{F}} L(h)$ for some loss function $L : \mathcal{F} \to \mathbb{R}$. In general, $\mathcal{F}$ is chosen to be a vector space, then it has a basis $\{\varphi_i\}_{i=0}^{\infty}$ so that any $h \in \mathcal{F}$ can be represented as $h = \sum_{i=0}^{\infty} \theta_i \varphi_i$ for some $\theta_i \in \mathbb{R}$. As a result, $\mathcal{F}$ can be paramterized by real-valued sequences $\theta = (\theta_0, \theta_1, \ldots)$, and the function learning problem can be tackled by using this paramterization as $\min_\theta L(\theta)$.

To illustrate the main idea, we pick $\mathcal{F}$ to be the finite-dimensional vector space spanned by the first r basis functions $\varphi_0, \ldots, \varphi_{r-1}$ and $L(\theta) = \frac{1}{2}\sum_{i=1}^{N} |y_i - \sum_{j=0}^{r-1} \theta_j \varphi_j(x_i)|^2$, which reduces the curve fitting to a regression problem. To find a concrete representation of the dynamical system adapted to this problem in the form of (5), let $Y = (y_1, \ldots, y_N)' \in \mathbb{R}^N$ denote the N-dimensional column vector consisting of the output data, and $H \in \mathbb{R}^{N\times r}$ be the regressor matrix with the (i, j)-entry defined by $H_{ij} = \Phi_j(x_i)$,then the loss function admit the matrix form $L(\theta) = \frac{1}{2}(Y - H\theta)'(Y - H\theta)$, whose gradient is given by $\nabla L(\theta) = H'H\theta - H'Y$. This immediately leads to the

system adapted to the regression problem as

$$\frac{d}{dt}\theta(t) = -H'H\theta + H'Y, \tag{7}$$

which characterizes the dynamics of the adapted ensemble system on $\mathcal{F}$ under the parameterization $\theta = (\theta_0, \ldots, \theta_{r-1}) \in \mathbb{R}^r$. Moreover, the system in (7) is a linear system whose solution is given by

$$\theta(t) = e^{-tH'H}\theta(0) + \int_0^t e^{(s-t)H'H}H'Y dt,$$

where $\theta(0)$ is the initial guess. In general, the regressor matrix H is full rank (unless there are redundant data) so that $-H'H$ is negative-definite, and hence $e^{-tH'H} \to 0$ as $t \to 0$. This implies that the solution of the adapted system in (7) converges to the solution of the regression problem regardless of the initial guess. Moreover, the invertibility of $H'H$ gives a more concrete representation of the solution

$$\theta(t) = e^{-tH'H}\theta(0) + (I - e^{-tH'H})(H'H)^{-1}H'Y,$$

where $I \in \mathbb{R}^{r\times r}$ denotes the identity matrix, and we use the commutativity of $(H'H)^{-1}$ and $e^{tH'H}$. When $t \to \infty$, $\theta(t) \to (H'H)^{-1}H'Y$, which exactly coincides the solution θ^* of the linear regression problem, theoretically verifying the proposed ensemble system-theoretic approach to function learning.

To demonstrate the applicability of this novel approach, we would like to learn the nonlinear function $h : [-1, 1] \to \mathbb{R}$, $x \mapsto \cos(1.15\pi x) + \sin(1.15\pi x)$ by using polynomial functions up to order 4, i.e., the function space $\mathcal{F}$ is the 5-dimensional vector space spanned by $\varphi_i(x) = x^i$ for $i = 0, 1, \ldots, 4$. To this end, we draw 20 samples $x_1, \ldots, x_{20}$ from the uniform distribution on [-1,1] as the input data, then noise the values of h evaluated at these points by a 0 mean and 0.05 variance Gaussian noise δ as the output data $y_i = h(x_i) + \delta$, $i = 1, \ldots, 20$. In this case, the regressor matrix H is a 20-by-5 matrix with the (i, j)-entry given by x_i^j for each $i = 1, \ldots, 20$ and $j = 0, \ldots, 4$. Specifically, we solve the ordinary differential equation system in (7) numerically for the time duration $[0, 100]$. The ℓ^2-error between $\theta(t)$ and θ^* and the cost with respect to time are shown in Figure 1, which rapidly converge to 0 and the minimum cost, respectively. Moreover, in Figure 2, we show the polynomials with coefficients $\theta(t)$ for $t = 10, 20, \ldots, 100$, which clearly converge to the least square solution h^* of the regression problem.

Remark 5 (Dynamic function learning and early stopping) *It is well-known in the machine learning society that early stopping is one of the most effective ways to improve the generalizability of learning algorithms. In the proposed ensemble system-theoretic function learning approach, early stopping can be simply realized by choosing a relatively small final time for the ensemble system adapted to a learning algorithm. In addition, compared with the classical "discrete-time" learning algorithms, the stopping criterion for this "continuous-time" algorithm does not restrict to integer time, which demonstrates a great potential to reach better generalizability.*

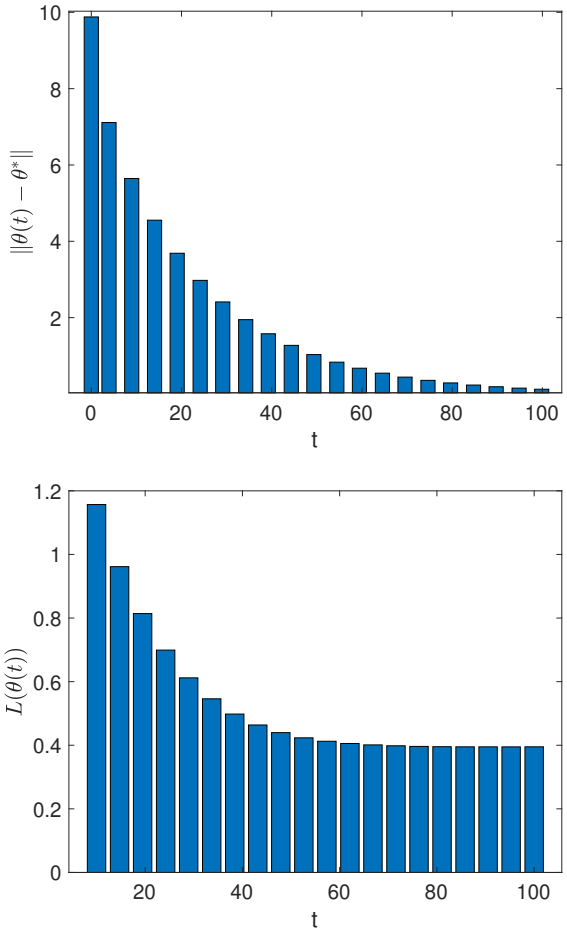

Figure 1: The time evolutions of the ℓ^2-error between $\theta(t)$ and θ^* (the top figure), and the cost $L(\theta(t))$ (the bottom figure), where $\theta(t)$ is the solution of the system in (7) adapted to the curve fitting problem $\min_{\theta\in\mathbb{R}^4} \sum_{i=1}^{20} |y_i - \sum_{j=0}^{4} \theta_i x_i^j|^2$, and θ^* is the least square solution.

Conclusions

In this paper, we propose a novel viewpoint for function learning problems through the lens of ensemble control theory. The core idea is to draw a parallel between the process of generating a sequence of functions by a iterative learning algorithm and the propagation of an ensemble system defined on a function space. Specifically, we establish an equivalence between convergence of function learning algorithms and asymptotic stability of ensemble systems, in which we particularly emphasize the importance of ensemble controllability of a system to the global convergence of the function learning algorithm, in the sense of being robust to initial guesses, generated by it. Moreover, the proposed framework is also applicable to parameterized model learning given prior knowledge about the targets, as well as learning problems with penalties. In addition, this ensemble system-theoretic framework further gives rise to a systematic approach to the design of "continuous-time" function

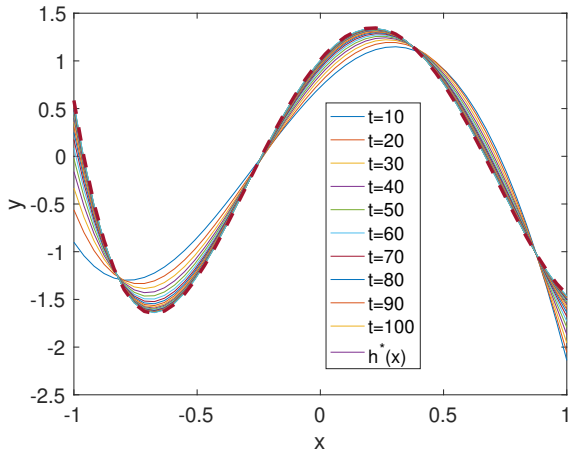

Figure 2: Convergence of the polynomial functions (the solid curves) with coefficients $\theta(t)$, $t = 10, 20, \ldots, 100$ to the least square solution of the curve fitting problem $\min_{\theta\in\mathbb{R}^4} \sum_{i=1}^{20} |y_i - \sum_{j=0}^{4} \theta_i x_i^j|^2$ (the dashed curve).

learning algorithms, which have a great potential to reach better generalizability than classical "discrete-time" algorithms.

Due to the nature of function learning problems, the developed framework requires data consisting both of the values of a function and the corresponding preimages in the domain. This implies learning algorithms generated by ensemble systems only works for supervised learning tasks, which indicates a limitation of this work. In the future, we would like to include unsupervised learning tasks into the ensemble control-theoretic learning framework, and on the other hand reverse the engineering to study ensemble control problems by using function leaning techniques, which promises to expand the scope of both control theory and machine learning.

Appendix

Proof of Proposition 1

Because the sequence $\{h_k\}_{k\in\mathbb{N}}$ is convergent, it must be a Cauchy sequence, i.e., $h_{k+1} - h_k = \Delta h_k \to 0$. By the definition of the ensemble system in (3), we also have $f(x(t_k, \cdot), \cdot) = \Delta h_k(\cdot)/(t_{k+1} - t_k) \to 0$, which particularly implies that f can be chosen to be a bounded function with bounded derivative. In order to show that this choice of f yields $f(h(\cdot), \cdot) = 0$, because of the continuity of f, it is suffices to prove that we can choose x_{t_k}, $k \in \mathbb{N}$ such that $x_{t_k} \to h$ as $k \to \infty$.

Note that because convergence is a long-term behavior of a dynamical system, we can let the system in (3) start from $x_0 = h_N$ for some N large enough so that $\|h_{k+1} - h_k\| = \|\Delta h_k\| < \varepsilon/2^k$ for all $k \geq N$ and some $\varepsilon > 0$, after passing h_k to a subsequence if necessary. In addition, without loss of generality, we also choose the sampling rate of the ensemble system in (3) to be uniform, that is, $t_{k+1} - t_k = \tau$ for all $k \in \mathbb{N}$. Following these assumptions, the error of the approximation of the sequence $\{h_k\}_{k\geq N}$ by the sampled trajectory $\{x_{t_k}\}_{k\geq 0}$ can be obtained iteratively as follows. We first apply Taylor's theorem to $x(t_1, \beta)$ up to the second order

$$\begin{aligned} x(t_1,\beta) &\approx x(0,\beta) + \tau\frac{d}{dt}x(0,\beta) + \frac{\tau^2}{2}\frac{d^2}{dt^2}x(0,\beta) \\ &= h_N(\beta) + \tau f(x(0,\beta),\beta) + \frac{\tau^2}{2}\frac{d}{dt}f(x(0,\beta),\beta) \\ &= h_{N+1}(\beta) + \frac{\tau^2}{2}Df(x(0,\beta),\beta)\cdot f(x(0,\beta),\beta) \\ &= h_{N+1}(\beta) + \frac{\tau}{2}Df(x(0,\beta),\beta)\cdot \Delta h_N(\beta), \end{aligned}$$

where Df denote the differential, i.e., the Jacobian matrix, of $f(x, \beta)$ with respect to x, and $\tau_0 = t_1 - t_0 = t_1$. Similarly, applying Taylor's theorem to $x(t_1, \beta)$ up to the second order yields

$$\begin{aligned} x(t_2,\beta) &\approx x(t_1,\beta) + \tau f(x(t_1,\beta),\beta) \\ &\quad + \frac{\tau}{2}Df(x(t_1,\beta),\beta)\cdot\Delta h_{N+1}(\beta) \\ &\approx h_{N+1}(\beta) + \tau f(x(t_1,\beta),\beta) \\ &\quad + \frac{\tau}{2}Df(x(t_1,\beta),\beta)\cdot\Delta h_{N+1}(\beta) \\ &\quad + \frac{\tau}{2}Df(x(0,\beta),\beta)\cdot\Delta h_N(\beta) \\ &= h_{N+2}(\beta) + \frac{\tau}{2}\big(Df(x(t_1,\beta),\beta)\cdot\Delta h_{N+1}(\beta) \\ &\quad + Df(x(0,\beta),\beta)\cdot\Delta h_N(\beta)\big). \end{aligned}$$

and inductively, we obtain

$$x(t_k,\beta) \approx h_{N+k}(\beta) + \frac{\tau}{2}\sum_{i=0}^{k-1} Df(x(t_i,\beta),\beta)\cdot\Delta h_n(\beta).$$

Let C be the upper bound of $\|Df\|$, then we have the following estimate

$$\begin{aligned} \|x(t_k,\cdot) - h_{N+k}\| &\leq \sum_{i=0}^{k-1}\frac{1}{2^{i+1}}\|Df(x(t_i,\cdot),\cdot)\|\|\Delta h_n\| \\ &\leq C(1-2^{-k})\varepsilon, \end{aligned}$$

which then yields $\|x(t_k, \cdot) - h_{N+k}\| \to 0$ as $k \to \infty$ since ε is arbitrary. Consequently, we obtain $\|x_{t_k} - h\| \leq \|x_{t_k} - h_{N+k}\| + \|h_{N+k} - h\| \to 0$ because both of the two terms on the right hand side approach 0 when $k \to \infty$, which also concludes the proof.

Proof of Proposition 2

Because $h \in \mathcal{F}(\Omega, \mathbb{R}^n)$ is an equilibrium point of the ensemble system in (3), we have $f(h(\cdot), \cdot) = 0$, the zero function. Together with the condition that $h_0 \in \mathcal{F}(\Omega, \mathbb{R}^n)$ is in the region of attraction of h, the solution of the system with the initial condition h_0 satisfies $x(t, \cdot) \to h(\cdot)$ as $t \to \infty$. We pick a sequence of time t_k, $k \in \mathbb{N}$ such that $t_k \to \infty$ as $k \to \infty$. Taylor's theorem implies that for each k

$$\begin{aligned} x(t_{k+1},\beta) &= x(t_k,\beta) + (t_{k+1}-t_k)\frac{d}{dt}x(\tau_k,\beta) \\ &= x(t_k,\beta) + (t_{k+1}-t_k)f(x(\tau_k,\beta),\beta) \end{aligned}$$

for some $t_k \leq \tau_k \leq t_{k+1}$. Then, we evolve the system from the initial condition $x_{t_0} = h_0$ and define $\Delta h_k(\cdot) = (t_{k+1} - t_k) f(x(\tau_k, \cdot), \cdot)$, which gives $h_k = x_{t_k}$ for all $k \in \mathbb{N}$, yielding the convergence of the learning algorithm $h_k \to h$ as $k \to \infty$.

Proof of Theorem 1

Necessity: Suppose that the system in (1) is ensemble controllable on $\mathcal{F}(\Omega, \mathbb{R}^n)$, then for any $\varepsilon > 0$ and any initial condition $x_0 \in \mathcal{F}(\Omega, \mathbb{R}^n)$, there is a control input $u(t)$ steering the system to a function $x(T, \cdot) \in \mathcal{F}(\Omega, \mathbb{R}^n)$ in a finite time T such that $\|x_T - h\| < \varepsilon$. Then, we design a function learning algorithm with the initial guess $h_0 = x_0$ as in the proof of Proposition 2, which will converge to h since ε is arbitrary.

Sufficiency: Given arbitrary h and h_0 in $\mathcal{F}(\Omega, \mathbb{R}^n)$, suppose that there is a function learning algorithm as in (2), generated by the ensemble system in (1) driven by a control input $u(t)$, converging to h with the initial guess h_0. Then, following the same proof as Proposition 1, we can show the ensemble system stabilizes to h, i.e., for any $\varepsilon > 0$, there is a finite time T such that the control input $u(t)$ steers the ensemble system to x_T satisfying $\|x_T - h\| < \varepsilon$. Because ε is arbitrary, it concludes ensemble controllability of the system.

References

Abu-Mostafa, Y. S.; Magdon-Ismail, M.; and Lin, H.-T. 2012. *Learning from data*, volume 4. AMLBook New York, NY, USA:.

Agrachev, A.; and Sarychev, A. 2020. Control On the Manifolds Of Mappings As a Setting For Deep Learning. *arXiv preprint arXiv:2008.12702*.

Albertini, F.; and Sontag, E. D. 1993. For neural networks, function determines form. *Neural networks*, 6(7): 975–990.

Bertsekas, D. P.; et al. 2000. *Dynamic programming and optimal control: Vol. 1*. Athena scientific Belmont.

Chen, X. 2020. Ensemble observability of Bloch equations with unknown population density. *Automatica*, 119: 109057.

Doya, K. 2000. Reinforcement learning in continuous time and space. *Neural computation*, 12(1): 219–245.

Goodfellow, I.; Bengio, Y.; and Courville, A. 2016a. *Deep learning*. MIT press.

Goodfellow, I.; Bengio, Y.; and Courville, A. 2016b. Regularization for deep learning. *Deep learning*, 216–261.

Haber, E.; and Ruthotto, L. 2017. Stable architectures for deep neural networks. *Inverse Problems*, 34(1): 014004.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hirsch, M. W. 1989. Convergent activation dynamics in continuous time networks. *Neural networks*, 2(5): 331–349.

Kuritz, K.; Zeng, S.; and Allgöwer, F. 2018. Ensemble controllability of cellular oscillators. *IEEE Control Systems Letters*, 3(2): 296–301.

Li, J.-S. 2006. *Control of inhomogeneous ensembles*. Ph.D. thesis, Harvard University USA.

Li, J.-S. 2011. Ensemble Control of Finite-Dimensional Time-Varying Linear Systems. *IEEE Transactions on Automatic Control*, 56(2): 345–357.

Li, J.-S.; Dasanayake, I.; and Ruths, J. 2013. Control and Synchronization of Neuron Ensembles. *IEEE Transactions on Automatic Control*, 58(8): 1919–1930.

Li, J.-S.; and Khaneja, N. 2009. Ensemble Control of Bloch Equations. *IEEE Transactions on Automatic Control*, 54(3): 528–536.

Li, J.-S.; Ruths, J.; Yu, T.-Y.; Arthanari, H.; and Wagner, G. 2011. Optimal pulse design in quantum control: A unified computational method. *Proceedings of the National Academy of Sciences*, 108(5): 1879–1884.

Li, J.-S.; Zhang, W.; and Tie, L. 2020. On separating points for ensemble controllability. *SIAM Journal on Control and Optimization*, 58(5): 2740–2764.

Ljung, L. 1999. System identification. *Wiley Encyclopedia of Electrical and Electronics Engineering*, 1–19.

Lu, Y.; Zhong, A.; Li, Q.; and Dong, B. 2018. Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. In *International Conference on Machine Learning*, 3276–3285. PMLR.

Michel, A. N.; Farrell, J. A.; and Porod, W. 1989. Qualitative analysis of neural networks. *IEEE Transactions on Circuits and Systems*, 36(2): 229–243.

Narayanan, V.; Zhang, W.; and Li, J.-S. 2020. Moment-Based Ensemble Control. *arXiv preprint arXiv:2009.02646*.

Narendra, K. S.; and Kannan, P. 1990. Identification and control of dynamical systems using neural networks. *IEEE Transactions on neural networks*, 1(1): 4–27.

Schoukens, J.; and Ljung, L. 2019. Nonlinear System Identification: A User-Oriented Road Map. *IEEE Control Systems Magazine*, 39(6): 28–99.

Sontag, E. D.; and Qiao, Y. 1999. Further results on controllability of recurrent neural networks. *Systems & control letters*, 36(2): 121–129.

Sontag, E. D.; and Sussmann, H. 1997. Complete controllability of continuous-time recurrent neural networks. *Systems & control letters*, 30(4): 177–183.

Sutton, R. S.; and Barto, A. G. 1998. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge.

Tabuada, P.; and Gharesifard, B. 2020. Universal approximation power of deep neural networks via nonlinear control theory. *arXiv preprint arXiv:2007.06007*.

Tarantola, A. 2005. *Inverse problem theory and methods for model parameter estimation*. SIAM.

Wang, S.; and Li, J.-S. 2018. Free-endpoint optimal control of inhomogeneous bilinear ensemble systems. *Automatica*, 95: 306–315.

Weinan, E. 2017. A proposal on machine learning via dynamical systems. *Communications in Mathematics and Statistics*, 5(1): 1–11.

Zeng, S. 2019. Iterative optimal control syntheses illustrated on the Brockett integrator. *IFAC-PapersOnLine*, 52(16): 138–143. 11th IFAC Symposium on Nonlinear Control Systems NOLCOS 2019.

Zeng, S.; and Allgöewer, F. 2016. A moment-based approach to ensemble controllability of linear systems. *Systems & Control Letters*, 98: 49–56.

Zeng, S.; Waldherr, S.; Ebenbauer, C.; and Allgöwer, F. 2016. Ensemble Observability of Linear Systems. *IEEE Transactions on Automatic Control*, 61(6): 1452–1465.

Zhang, W.; and Li, J.-S. 2021. Ensemble Control on Lie Groups. *SIAM Journal on Control and Optimization*, 59(5): 3805–3827.