Universal approximation power of deep residual neural networks through the lens of control theory

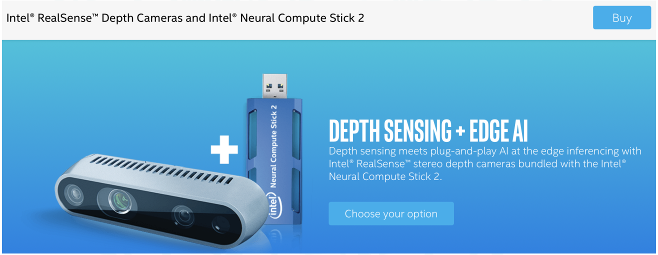
Bahman Gharesifard
UCLA

Séminaire CAS
École des Mines de Paris
May 2022

This is joint work with Paulo Tabuada (UCLA)

Motivation: neural networks in the loop

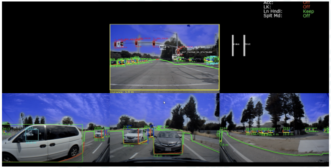
- LiDAR and depth cameras have become the key source of estimation in robotics:



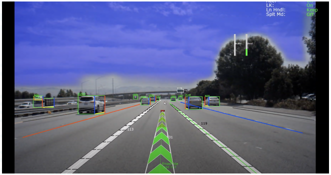
- Perception pipelines are supposed to transform LiDAR outputs into state estimates

Motivation: neural networks in the loop

- There is a widespread use of deep Neural Networks (NNs) for processing vision and LiDAR data; example below is from NVIDIA:



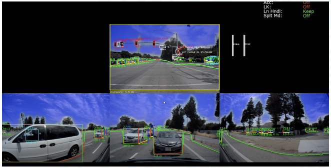
Perception using Lidar



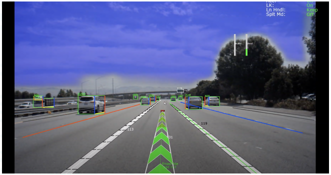
Self-driving using Lidar data

Motivation: neural networks in the loop

- There is a widespread use of deep Neural Networks (NNs) for processing vision and LiDAR data; example below is from NVIDIA:



Perception using Lidar



Self-driving using Lidar data

A key starting point is how to use this perception pipeline in generating control inputs with stability guarantees

Neural Networks

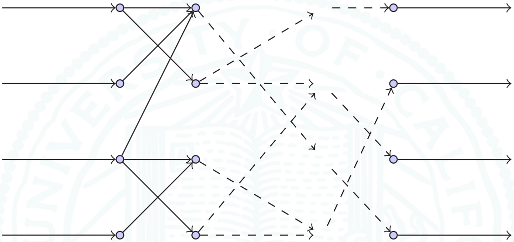


Figure: Neural Network

A feedforward neural network consists of

- **input and output layers**, and a number of **hidden layers**
- Each layer ℓ consists of a set of **nodes**
- **Edges** from nodes in layer $\ell - 1$ to nodes in layer ℓ, each equipped with a **weight** w_{jk}^{ℓ} on the edge *into* the jth node in layer ℓ from the kth node in layer $\ell - 1$
- The output of the jth node in layer ℓ will be denoted by $\hat{x}_j^{\ell}$

Neural Networks

In each layer, the neural network performs the following update:

$$\hat{x}_j^\ell = \sigma\left(\sum_{k=1} w_{jk}^\ell \hat{x}_k^{\ell-1} + b_j^\ell\right)$$

where b_j^ℓ is a constant, and σ is the **activation function**.

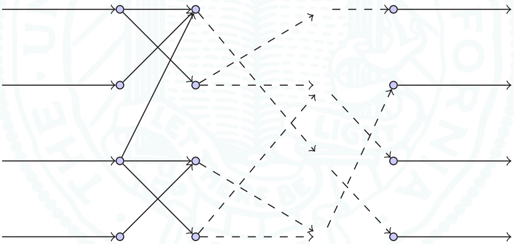


Figure: Neural Network

Residual neural network

We work with residual neural network, where there is a possibility of a so-called skip connection:

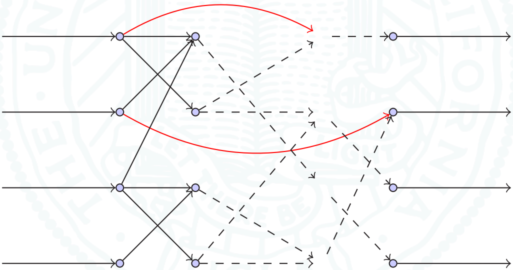


Figure: Neural Network

Neural Networks

- Given a set of data $\{(x^i_{\text{samples}}, y^i_{\text{samples}})\}_{i=1}^N$

Neural Networks

- Given a set of data $\{(x^i_{\text{samples}}, y^i_{\text{samples}})\}_{i=1}^N$

Typical objective: solve

$$\inf_{w,b} \frac{1}{N} \sum_{i=1}^{N} \|y^i_{\text{samples}} - \hat{x}^\ell(x^i_{\text{samples}})\|_2,$$

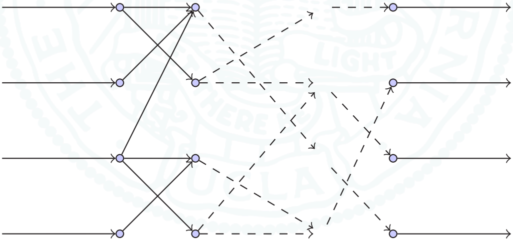
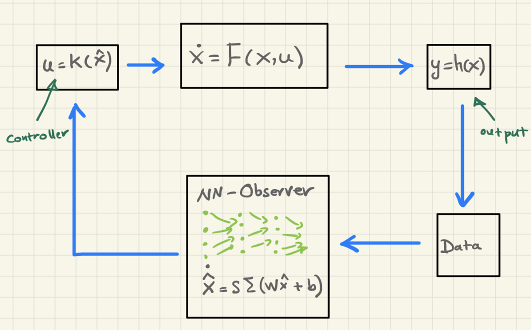


Figure: Neural Network

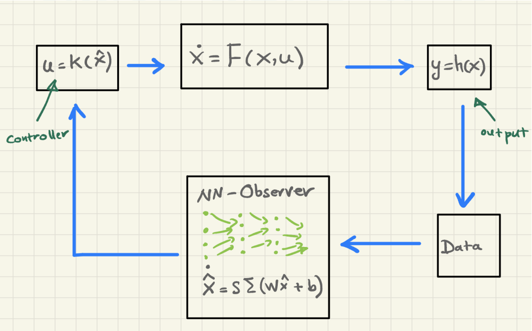
Neural networks for perception

Use neural network for estimation in a control-loop, while verifying "performance"



Neural networks for perception

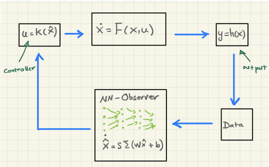
Use neural network for estimation in a control-loop, while verifying "performance"



we do not wish for inaccuracies in estimation to lead to huge spikes (some ISS property in needed)

Neural networks for perception

- We hope to train a residual neural network to **learn** the map **from output measurements** y to the state x



- The main question then is:

How good a (training algorithm for) neural network can approximate a given function?

Outline

Classical universal approximation question

A more delicate objective (written here informally) is

Function approximation: Given

- a continuous function $f : \mathbb{R}^n \to \mathbb{R}^m$,

Classical universal approximation question

A more delicate objective (written here informally) is

Function approximation: Given

- a continuous function $f : \mathbb{R}^n \to \mathbb{R}^m$,
- $E \subset \mathbb{R}^n$ a compact set, and

Classical universal approximation question

A more delicate objective (written here informally) is

Function approximation: Given

- a continuous function $f : \mathbb{R}^n \to \mathbb{R}^m$,
- $E \subset \mathbb{R}^n$ a compact set, and
- $\varepsilon \in \mathbb{R}^+$ be the desired approximation accuracy

Classical universal approximation question

A more delicate objective (written here informally) is

Function approximation: Given

- a continuous function $f : \mathbb{R}^n \to \mathbb{R}^m$,
- $E \subset \mathbb{R}^n$ a compact set, and
- $\varepsilon \in \mathbb{R}^+$ be the desired approximation accuracy

Does there exists a neural network such the inputs can be "trained" on a large enough finite sample $E_{\text{samples}} \subset E$ such that the output, denoted by $g : \mathbb{R}^n \to \mathbb{R}^m$, satisfies

$$\|f - g\|_{L^p(E)} \leq \varepsilon,$$

or better

$$\|f - g\|_{L^\infty(E)} \leq \varepsilon?$$

Brief history of function approximation

Neural networks with **arbitrary width**

Brief history of function approximation

Neural networks with **arbitrary width**

Some key classical work (many others omitted here):

- **Cybenko**, *Approximation by superpositions of a sigmoidal function*, Mathematics of Control, Signals, and Systems, 1989
- **Hornik**, *Approximation capabilities of multilayer feedforward networks*, Neural Networks, 1991
- **Pinkus**, *Approximation theory of the MLP model in neural networks*, Acta Numerica, 1999

The results above, even though applicable to any depth, **rely on the fact that there is no bound on the width**

History of function approximation

Deep narrow neural networks, **bounded width**

History of function approximation

Deep narrow neural networks, **bounded width**

Reason for interest: much easier to train

History of function approximation

Deep narrow neural networks, **bounded width**

Reason for interest: much easier to train

The literature on this subject is less classical, and obtaining results in uniform norm, rather than L^p, are particularly difficult

History of function approximation

Deep narrow neural networks, **bounded width**

Reason for interest: much easier to train

The literature on this subject is less classical, and obtaining results in uniform norm, rather than L^p, are particularly difficult

For $f : \mathbb{R}^n \to \mathbb{R}^m$ (mostly for feedforward networks):

- **Lu et. al.:** *The expressive power of neural networks: A view from the width, Advances in Neural Information Processing Systems, 2017* L^1 results for ReLU with $m = 1$ $(n + 1 \leq w_{\text{min}} \leq n + 4)$
- **Hanin and and Sellke:** *Approximating continuous functions by ReLU nets of minimal width, 2017* uniform results for ReLU $(n + 1 \leq w_{\text{min}} \leq n + m)$

History of function approximation

Deep narrow neural networks, **bounded width**

Reason for interest: much easier to train

The literature on this subject is less classical, and obtaining results in uniform norm, rather than L^p, are particularly difficult

For $f : \mathbb{R}^n \to \mathbb{R}^m$ (mostly for feedforward networks):

- **Lu et. al.:** *The expressive power of neural networks: A view from the width, Advances in Neural Information Processing Systems, 2017* L^1 results for ReLU with $m = 1$ $(n + 1 \leq w_{\text{min}} \leq n + 4)$
- **Hanin and and Sellke:** *Approximating continuous functions by ReLU nets of minimal width, 2017* uniform results for ReLU $(n + 1 \leq w_{\text{min}} \leq n + m)$
- **Kidger and Lyons:** *Universal approximation with deep narrow networks, Conference on Learning Theory, 2020* uniform results for very general class of activation functions $(w_{\text{min}} \leq n + m + 1)$

History of function approximation

Deep narrow neural networks

Closing the gap:

History of function approximation

Deep narrow neural networks

Closing the gap:

- **Park, Yun, Lee, and Shin**, Minimum width for universal approximation, **International Conference on Learning Representations, 2021**
  - uniform results for "ReLU" and feedforward networks ($w_{\text{min}} = \max\{n+1, m\}$)
  - uniform results for more general feedforward networks ($w_{\text{min}} = \max\{n+2, m\}$)

- **P. Tabuada and BG**, Universal approximation power of deep residual neural networks via nonlinear control theory, **International Conference on Learning Representations, 2021**
  - uniform results for **a large class of activation functions** with
    - $n \geq m$ ($w_{\text{min}} = n+1$)
    - $n < m$ ($w_{\text{min}} = m+1$)
  - **Result notably apply to residual neural networks**

Outline

Neural networks in perception pipeline

Brief history of universal approximation

Learning capabilities of residual networks on finite samples through ensemble controllability

A geometric approach

Universal approximation power of residual networks

Outlook

Control-theoretic view of residual networks

Control system perspective on residual neural networks:[1] [2]

$$\boxed{x(k+1) = x(k) + s(k)\Sigma(W(k)x(k) + b(k))}$$

- the layer k is viewed as indexing time
- $(s(k), W(k), b(k)) \in \mathbb{R}^{n\times n} \times \mathbb{R} \times \mathbb{R}^n$ are the control inputs
- $\Sigma(x) = (\sigma(x_1), \sigma(x_2), \ldots, \sigma(x_n))$ with σ the **activation function**

[1] E. Haber and L. Ruthotto. Stable architectures for deep neural networks. Inverse Problems, 2017

[2] E. Weinan. A proposal on machine learning via dynamical systems. Communications in Mathematics and Statistics, 2017

Control-theoretic view of residual networks

Control system perspective on residual neural networks:[1] [2]

$$\boxed{x(k+1) = x(k) + s(k)\Sigma(W(k)x(k) + b(k))}$$

- the layer k is viewed as indexing time
- $(s(k), W(k), b(k)) \in \mathbb{R}^{n\times n} \times \mathbb{R} \times \mathbb{R}^n$ are the control inputs
- $\Sigma(x) = (\sigma(x_1), \sigma(x_2), \ldots, \sigma(x_n))$ with σ the **activation function**

In continuous-time, this reads as

$$\boxed{\dot{x}(t) = s(t)\Sigma(W(t)x(t) + b(t))}$$

[1] E. Haber and L. Ruthotto. Stable architectures for deep neural networks. Inverse Problems, 2017

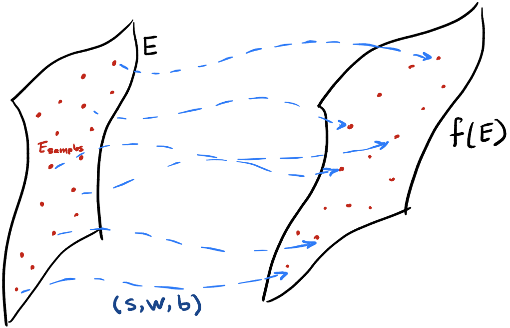
[2] E. Weinan. A proposal on machine learning via dynamical systems. Communications in Mathematics and Statistics, 2017

Function approximation, reformulated

Given

- a function $f : \mathbb{R}^n \to \mathbb{R}^n$
- a finite set of samples $E_{\text{samples}} \subset \mathbb{R}^n$,

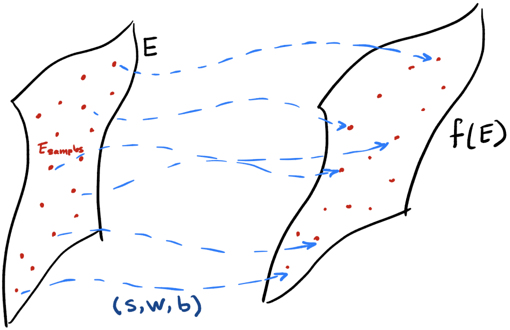
Construct an open-loop control input (S, W, b) to take $x \in E_{\text{samples}}$ to the states $f(x)$



Function approximation, reformulated

Punchline

- We need to **drive an ensemble** of samples **with one controller**
- This is different from the classical framework of ensemble control[3]



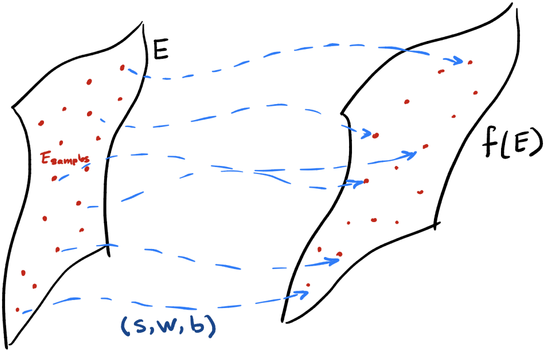
[3] This is related to A. Agrachev and A. Sarychev. Control in the spaces of ensembles of points. SICON, 2020, and also A.A. Agrachev and M. Caponigro. Controllability on the group of diffeomorphisms. Annales de l'Institut Henri Poincare, 2009.

Function approximation, reformulated

Ensemble system: $d = |E_{\text{samples}}|$ copies given by:

$$\dot{X}(t) = [S(t)\Sigma(W(t)X_{\bullet 1}(t) + b(t)) \mid \dots \mid S(t)\Sigma(WX_{\bullet d}(t) + b(t)))]$$

where $X(t) \in \mathbb{R}^{n \times d}$ and $X_{\bullet i}(t)$ is the solution of the ith copy in the ensemble, and $X^{\text{init}} = [x^1|x^2|\dots|x^d]$ and $X^{\text{fin}} = [f(x^1)|f(x^2)|\dots|f(x^d)]$,



Outline

Neural networks in perception pipeline

Brief history of universal approximation

Learning capabilities of residual networks on finite samples through ensemble controllability

A geometric approach

Universal approximation power of residual networks

Outlook

Geometric control in one slide

Consider the control system:

$$\dot{x} = u_1 Z_1(x) + u_2 Z_2(x),$$

Think of Z_1 and Z_2 as direction that you can travel along directly, and u_1 and u_2 as the controls you can apply

Geometric control in one slide

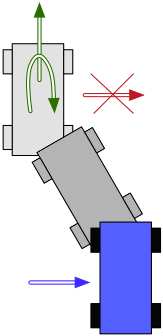
Consider the control system:

$$\dot{x} = u_1 Z_1(x) + u_2 Z_2(x),$$

Think of Z_1 and Z_2 as direction that you can travel along directly, and u_1 and u_2 as the controls you can apply

Key idea in control theory: we can obtain new directions **by "*concatenating*" the two direction**

Example: Parallel parking



Geometric control in one slide

Consider the control system:

$$\dot{x} = u_1 Z_1(x) + u_2 Z_2(x),$$

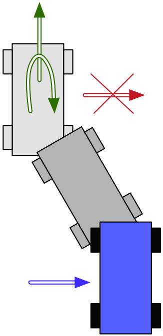
Think of Z_1 and Z_2 as direction that you can travel along directly, and u_1 and u_2 as the controls you can apply

The "extra control directions" are mathematically characterized by **Lie brackets:**

$$\boxed{[Z_1, Z_2](x) := \frac{\partial Z_2}{\partial x} Z_1(x) - \frac{\partial Z_1}{\partial x} Z_2(x)}$$

In this sense, the reachable set of the system above is equivalent to the one of

$$\dot{x} = u_1 Z_1(x) + u_2 Z_2(x) + u_3 [Z_1, Z_2](x)$$



Geometric control in one slide

Consider the control system:

$$\dot{x} = u_1 Z_1(x) + u_2 Z_2(x),$$

Think of Z_1 and Z_2 as direction that you can travel along directly, and u_1 and u_2 as the controls you can apply

Iterating on this, the Lie algebra generated by the vector fields in a set $\mathcal{F}$ is denoted by $\mathrm{Lie}(\mathcal{F})$

- e.g., for $\mathcal{F} = \{Z_1, Z_2\}$,

$$\mathrm{Lie}(\mathcal{F}) = \{Z_1, Z_2, [Z_1, Z_2], [Z_1, [Z_1, Z_2]], \cdots\}$$

- The celebrated **Chow-Rashevsky Theorem** implies that the driftless control affine system above is **controllable** if

$$\mathrm{Lie}(\mathcal{F})(x) = \mathbb{R}^n$$

  for every point $x \in \mathbb{R}^n \backslash \{0\}$.

Ensemble controllability

Although I am skipping some important technical points, you should except to see higher order derivatives of σ when describing the Lie algebra

Ensemble controllability

Although I am skipping some important technical points, you should except to see higher order derivatives of σ when describing the Lie algebra

Assuming that the number of sample points d is larger than n here, it is enough to have that

$$\begin{bmatrix} 1 & \sigma(A_{1\ell}) & D\sigma(A_{1\ell}) & \cdots & D^{d-2}\sigma(A_{1\ell}) \\ 1 & \sigma(A_{2\ell}) & D\sigma(A_{2\ell}) & \cdots & D^{d-2}\sigma(A_{2\ell}) \\ \vdots & \vdots & & \vdots & \\ 1 & \sigma(A_{n\ell}) & D\sigma(A_{n\ell}) & \cdots & D^{d-2}\sigma(A_{n\ell}) \end{bmatrix},$$

where $A \in \mathbb{R}^{n\times d}$ and $\ell \in \{1,\ldots,n\}$, **has rank** n

Ensemble controllability

The next key observation

Proposition. Suppose $\xi : \mathbb{R} \to \mathbb{R}$ satisfies the **quadratic differential equation**:

$$D\xi(x) = a_0 + a_1\xi(x) + a_2\xi^2(x),$$

where $a_0, a_1, a_2 \in \mathbb{R}$, with $a_2 \neq 0$.

Ensemble controllability

The next key observation

Proposition. Suppose $\xi : \mathbb{R} \to \mathbb{R}$ satisfies the **quadratic differential equation**:

$$D\xi(x) = a_0 + a_1\xi(x) + a_2\xi^2(x),$$

where $a_0, a_1, a_2 \in \mathbb{R}$, with $a_2 \neq 0$. Then, the **determinant of the matrix:**

$$L(x_1, x_2, \ldots, x_\ell) = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \xi(x_1) & \xi(x_2) & \ldots & \xi(x_\ell) \\ D\xi(x_1) & D\xi(x_2) & \ldots & D\xi(x_\ell) \\ \vdots & \vdots & \ddots & \vdots \\ D^{\ell-2}\xi(x_1) & D^{\ell-2}\xi(x_2) & \ldots & D^{\ell-2}\xi(x_\ell) \end{bmatrix},$$

is non-zero if and only if

$$\prod_{1 \leq i < j \leq \ell} (\xi(x_i) - \xi(x_j)) \neq 0$$

Class of activation functions

Interestingly, a large class of activation functions $\sigma : \mathbb{R} \to \mathbb{R}$ satisfy:

$$\boxed{D\xi = a_0 + a_1\xi + a_2\xi^2}$$

with $a_1, a_2, a_3 \in \mathbb{R}$, $a_2 \neq 0$, and $\xi = D^j\sigma$ for some $j \in \mathbb{N}_0$.

Class of activation functions

Interestingly, a large class of activation functions $\sigma : \mathbb{R} \to \mathbb{R}$ satisfy:

$$\boxed{D\xi = a_0 + a_1\xi + a_2\xi^2}$$

with $a_1, a_2, a_3 \in \mathbb{R}$, $a_2 \neq 0$, and $\xi = D^j\sigma$ for some $j \in \mathbb{N}_0$.

Here are some examples:

Function name	**Definition**	**Satisfied differential equation**
Logistic function	$\sigma(x) = \frac{1}{1+e^{-x}}$	$D\sigma - \sigma + \sigma^2 = 0$
Hyperbolic tangent	$\sigma(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$	$D\sigma - 1 + \sigma^2 = 0$
Leaky ReLU	$\sigma(x) = x$ for $x \geq 0$ and	$D^2\sigma - k(1+r)D\sigma + k(D\sigma)^2 + kr = 0$
	$\sigma(x) = rx$ for $x < 0$	as $k \to \infty$
Soft plus	$\sigma(x) = \frac{1}{r}\log(1 + e^{rx})$	$D^2\sigma - rD\sigma + r(D\sigma)^2 = 0$

Table: Some activation functions and the differential equations they satisfy

Controllability result

Theorem. Let $N \subset \mathbb{R}^{n\times d}$ be the set defined by:

$$N = \{A \in \mathbb{R}^{n\times d} \mid \prod_{1\leq i<j\leq d} (A_{\ell i} - A_{\ell j}) = 0,\ \ell \in \{1, \ldots, n\}\}.$$

Let $n > 1$ and suppose that the activation function satisfies the mentioned assumption. Then the ensemble control system is controllable on the submanifold $M = \mathbb{R}^{n\times d} \backslash N$.

Note that for $n \neq 1$ this submanifold is connected, open dense subset of $\mathbb{R}^{n\times d} \Rightarrow$

As long as E_{samples}, and $f(E_{\text{samples}})$ are in M, we have ensemble controllability

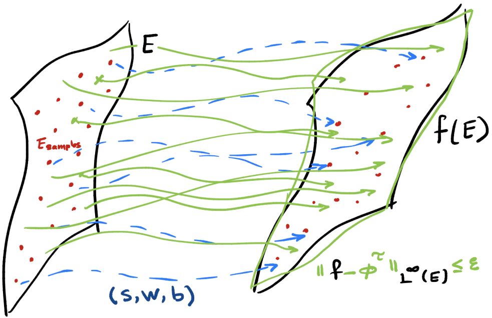
Some final remarks on ensemble controllability

Very key ingredient in our understanding the structure of the Lie algebra is utilizing the fact that the **activation function** satisfies

$$\boxed{D\xi = a_0 + a_1\xi + a_2\xi^2}$$

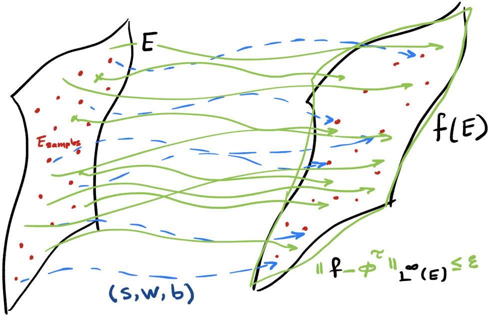
Function approximation

There is still a major step to ensure function approximation:



Function approximation

There is still a major step to ensure function approximation:



How do we ensure that the points in between samples are mapped in a way that they guarantee uniform approximation?

Outline

Neural networks in perception pipeline

Brief history of universal approximation

Learning capabilities of residual networks on finite samples through ensemble controllability

A geometric approach

Universal approximation power of residual networks

Outlook

Key idea: Monotonicity

The key idea that helps us in this step is **monotonicity**:

On $\mathbb{R}^n$, with the ordering relation $x \preceq x'$ defined by $x_i \leq x'_i$ for all $i \in \{1, \ldots, n\}$ and $x, x' \in \mathbb{R}^n$

A map $f : \mathbb{R}^n \to \mathbb{R}^n$ is said to be a **monotone map** when $x \preceq x'$ implies $f(x) \preceq f(x')$.

- When f is continuous differentiable, monotonicity admits a simple characterization:

$$\frac{\partial f_i}{\partial x_j} \geq 0, \quad \forall i, j \in \{1, \ldots, n\}.$$

Key idea: Monotonicity

The key idea that helps us in this step is **monotonicity**:

On $\mathbb{R}^n$, with the ordering relation $x \preceq x'$ defined by $x_i \leq x'_i$ for all $i \in \{1, \ldots, n\}$ and $x, x' \in \mathbb{R}^n$

A map $f : \mathbb{R}^n \to \mathbb{R}^n$ is said to be a **monotone map** when $x \preceq x'$ implies $f(x) \preceq f(x')$.

- When f is continuous differentiable, monotonicity admits a simple characterization:

$$\frac{\partial f_i}{\partial x_j} \geq 0, \quad \forall i, j \in \{1, \ldots, n\}.$$

A vector field $Z : \mathbb{R}^n \to \mathbb{R}^n$ is said to be monotone when its flow $\phi^\tau : \mathbb{R}^n \to \mathbb{R}^n$ is a monotone map

Function approximation: monotonicity of vector field

An important fact: Suppose that

- $f : \mathbb{R}^n \to \mathbb{R}^n$ is a continuous map on $E \subset \mathbb{R}^n$ a compact set
- Suppose $E_{\mathsf{samples}} \subset \mathbb{R}^n$ **contains fine enough samples** such that

$$\forall x \in E \quad \exists\, \underline{x}, \overline{x} \in E_{\mathsf{samples}},$$
$$|\underline{x} - \overline{x}|_\infty \le \delta \quad \text{and} \quad \underline{x}_i \le x_i \le \overline{x}_i,$$

Function approximation: monotonicity of vector field

An important fact: Suppose that

- $f : \mathbb{R}^n \to \mathbb{R}^n$ is a continuous map on $E \subset \mathbb{R}^n$ a compact set
- Suppose $E_{\text{samples}} \subset \mathbb{R}^n$ **contains fine enough samples** such that

$$\forall x \in E \quad \exists\, \underline{x}, \overline{x} \in E_{\text{samples}}, \\ |\underline{x} - \overline{x}|_\infty \leq \delta \quad \text{and} \quad \underline{x}_i \leq x_i \leq \overline{x}_i,$$

- Suppose that $\phi : \mathbb{R}^n \to \mathbb{R}^n$ is a **monotone map** satisfying:

$$\|f - \phi\|_{L^\infty(E_{\text{samples}})} \leq \zeta,$$

with $\zeta \in \mathbb{R}^+$.

Function approximation: monotonicity of vector field

An important fact: Suppose that

- $f : \mathbb{R}^n \to \mathbb{R}^n$ is a continuous map on $E \subset \mathbb{R}^n$ a compact set
- Suppose $E_{\text{samples}} \subset \mathbb{R}^n$ **contains fine enough samples** such that

$$\forall x \in E \quad \exists\, \underline{x}, \overline{x} \in E_{\text{samples}},$$
$$|\underline{x} - \overline{x}|_\infty \le \delta \quad \text{and} \quad \underline{x}_i \le x_i \le \overline{x}_i,$$

- Suppose that $\phi : \mathbb{R}^n \to \mathbb{R}^n$ is a **monotone map** satisfying:

$$\|f - \phi\|_{L^\infty(E_{\text{samples}})} \le \zeta,$$

  with $\zeta \in \mathbb{R}^+$.

Then,

$$\boxed{\|f - \phi\|_{L^\infty(E)} \le 2\omega_f(\delta) + 3\zeta}$$

where ω_f is the **modulus of continuity** of f.

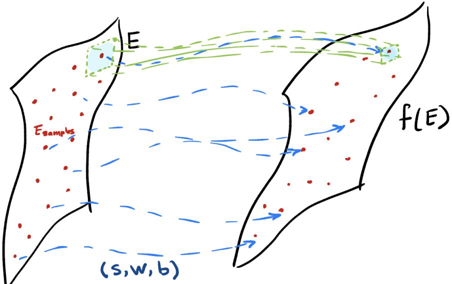
Function approximation: monotone functions

Is it possible to ensure monotonicity of the generated flow?

Function approximation: monotone functions

Theorem. Let $n > 1$ and suppose that the activation function satisfies the mentioned assumption. Then, for every **monotone analytic function** $f : \mathbb{R}^n \to \mathbb{R}^n$, $E \subset \mathbb{R}^n$ compact, and for every $\varepsilon \in \mathbb{R}^+$ there exist a time $\tau \in \mathbb{R}^+$ and an input $(s, W, b) : [0, \tau] \to \mathbb{R} \times \mathbb{R}^{n \times n} \times \mathbb{R}^n$ so that the flow $\phi^\tau : \mathbb{R}^n \to \mathbb{R}^n$ of the corresponding control system satisfies

$$\|f - \phi^\tau\|_{L^\infty(E)} \leq \varepsilon.$$



Function approximation: monotone functions

The proof is technical and relies on:

1. Ensuring that change of orders of the entries of the flow occur only finite number of times (**analyticity helps here**)

Function approximation: monotone functions

The proof is technical and relies on:

1. Ensuring that change of orders of the entries of the flow occur only finite number of times (**analyticity helps here**)

2. Ensuring that the inputs are constructed in a way that can guarantee monotonicity with in subinterval of times, and monotonicity in transitions (**neural networks being overly actuated helps here**)

The general case

What happens when the map to be approximated is not monotone?

The general case

What happens when the map to be approximated is not monotone?

Key idea: Utilize embeddings[4] [5]

> Embedding of a function to a monotone one usually requires **doubling of the state**, however, **we can get away with only adding an extra dimension**, again due to flexibility of the design of the neural network

[4] Fort, The embedding of homeomorphisms in flows, Proc. of AMS, 1955
[5] Utz, The embedding of homeomorphisms in continuous flows, Top. Proc. 1981

Function approximation: the general case

Embed the function f on $\mathbb{R}^n$ into a **monotone function** $\tilde{f}$ on $\mathbb{R}^\kappa$, where $\kappa = n + 1$

[6] α and β are implemented by the first and last layer of the neural network

Function approximation: the general case

Embed the function f on $\mathbb{R}^n$ into a **monotone function** $\tilde{f}$ on $\mathbb{R}^\kappa$, where $\kappa = n + 1$

In particular, we find:

- An injection $\alpha : \mathbb{R}^n \to \mathbb{R}^\kappa$, and
- A projection[6] $\beta : \mathbb{R}^\kappa \to \mathbb{R}^n$ such that

$$f = \beta \circ \tilde{f} \circ \alpha$$

where $\tilde{f}$ is monotone

[6] α and β are implemented by the first and last layer of the neural network

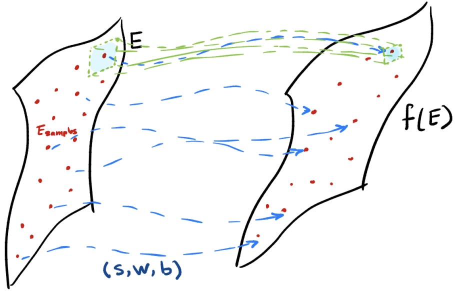
Function approximation: the general case

Theorem. Let $n > 1$ and suppose that the activation function satisfies the mentioned assumption. Then, for every continuous function $f : \mathbb{R}^n \to \mathbb{R}^n$, compact set $E \subset \mathbb{R}^n$, and for every $\varepsilon \in \mathbb{R}^+$ there exist a time $\tau \in \mathbb{R}^+$, an injection $\alpha : \mathbb{R}^n \to \mathbb{R}^\kappa$, $\kappa = n + 1$, a projection $\beta : \mathbb{R}^\kappa \to \mathbb{R}^n$, and an input $(s, W, b) : [0, \tau] \to \mathbb{R} \times \mathbb{R}^{\kappa \times \kappa} \times \mathbb{R}^\kappa$ so that the flow $\phi^\tau : \mathbb{R}^\kappa \to \mathbb{R}^\kappa$ defined by the solution of the corresponding control system

$$\|f - \beta \circ \phi^\tau \circ \alpha\|_{L^\infty(E)} \le \varepsilon.$$

Summary of our approach to uniform approximation

1. **Control-theoretic view of residual networks**
2. **Controllability of sample ensembles** ala geometric control
3. Key **role of monotonicity** in uniform approximation outside samples
4. **Ensuring monotonicity by embedding** into a monotone map



Outlook

Among many things:

- Training neural networks with guarantees for control
- Data-driven control using deep residual networks
- Issues of overfitting and regularization
- Issues with low-dimensional data

Reference

1. **Universal approximation power of deep residual neural networks through the lens of control**, Paulo Tabuada and BG, accepted TAC, 2022

2. Training deep residual networks for uniform approximation guarantees, M. Marchi, BG, P. Tabuada, L4DC 2020

3. Stability guarantees for control loops with deep learning state estimation, M. Marchi, J. Bunton, BG, Tabuada, IEEE Control Systems Letters, 2021