Learning to control unknown feedback linearizable systems from expert demonstrations

Alimzhan Sultangazin

March 7, 2022

Introduction

- We want to design a controller for an autonomous car that prioritizes comfort of its passengers
- Express comfort with a cost function and use optimal control?

[1] D. Vogt et al. "A system for learning continuous human-robot interactions from human-human demonstrations". In: *ICRA*. 2017.

[2] J. van den Berg et al. "Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations". In: *ICRA*. 2010.

[3] Pieter Abbeel and Andrew Y. Ng. "Apprenticeship Learning via Inverse Reinforcement Learning". In: *ICML*. 2004.

[4] Harish Ravichandar et al. "Recent Advances in Robot Learning from Demonstration". In: *Annual Review of Control, Robotics, and Autonomous Systems* 3.1 (2020), pp. 297–330.

Introduction

- We want to design a controller for an autonomous car that prioritizes comfort of its passengers
- Express comfort with a cost function and use optimal control?
  What is the cost function? How to mathematically express comfort?

[1] D. Vogt et al. "A system for learning continuous human-robot interactions from human-human demonstrations". In: *ICRA*. 2017.

[2] J. van den Berg et al. "Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations". In: *ICRA*. 2010.

[3] Pieter Abbeel and Andrew Y. Ng. "Apprenticeship Learning via Inverse Reinforcement Learning". In: *ICML*. 2004.

[4] Harish Ravichandar et al. "Recent Advances in Robot Learning from Demonstration". In: *Annual Review of Control, Robotics, and Autonomous Systems* 3.1 (2020), pp. 297–330.

Introduction

- We want to design a controller for an autonomous car that prioritizes comfort of its passengers
- Express comfort with a cost function and use optimal control?
  What is the cost function? How to mathematically express comfort?
- We believe it is easier to collect demonstrations of "comfortable driving" — learning from demonstrations (LfD).

[1] D. Vogt et al. "A system for learning continuous human-robot interactions from human-human demonstrations". In: *ICRA*. 2017.

[2] J. van den Berg et al. "Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations". In: *ICRA*. 2010.

[3] Pieter Abbeel and Andrew Y. Ng. "Apprenticeship Learning via Inverse Reinforcement Learning". In: *ICML*. 2004.

[4] Harish Ravichandar et al. "Recent Advances in Robot Learning from Demonstration". In: *Annual Review of Control, Robotics, and Autonomous Systems* 3.1 (2020), pp. 297–330.

Introduction

- We want to design a controller for an autonomous car that prioritizes comfort of its passengers
- Express comfort with a cost function and use optimal control?
  What is the cost function? How to mathematically express comfort?
- We believe it is easier to collect demonstrations of "comfortable driving" — learning from demonstrations (LfD).
- Many other control tasks benefit from LfD, e.g., manufacturing [1]; healthcare [2]; robotics [3]. The growing research interest in LfD [4] reflects the need for a well-defined design methodology.

---

[1] D. Vogt et al. "A system for learning continuous human-robot interactions from human-human demonstrations". In: *ICRA*. 2017.

[2] J. van den Berg et al. "Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations". In: *ICRA*. 2010.

[3] Pieter Abbeel and Andrew Y. Ng. "Apprenticeship Learning via Inverse Reinforcement Learning". In: *ICML*. 2004.

[4] Harish Ravichandar et al. "Recent Advances in Robot Learning from Demonstration". In: *Annual Review of Control, Robotics, and Autonomous Systems* 3.1 (2020), pp. 297–330.

Bird's eye view of this work

- In this work, we combine two methodologies: the LfD methodology from [5] and the data-driven control methodology from [6].

---

[5] Alimzhan Sultangazin et al. *Watch and Learn: Learning to control feedback linearizable systems from expert demonstrations*. Tech. rep. UCLA, 2021. URL: http://www.cyphylab.ee.ucla.edu/Home/publications/UCLA-CyPhyLab-2021-03.pdf.

[6] Lucas Fraile, Matteo Marchi, and Paulo Tabuada. "Data-driven Stabilization of SISO Feedback Linearizable Systems". In: *arXiv e-prints* (Mar. 2020). URL: https://arxiv.org/abs/2003.14240.

Bird's eye view of this work

- In this work, we combine two methodologies: the LfD methodology from [5] and the data-driven control methodology from [6].
- The LfD methodology leverages the fact that the solution set of a linear system is a vector space to construct a stabilizing control law.

[5] Alimzhan Sultangazin et al. *Watch and Learn: Learning to control feedback linearizable systems from expert demonstrations.* Tech. rep. UCLA, 2021. URL: `http://www.cyphylab.ee.ucla.edu/Home/publications/UCLA-CyPhyLab-2021-03.pdf`.

[6] Lucas Fraile, Matteo Marchi, and Paulo Tabuada. "Data-driven Stabilization of SISO Feedback Linearizable Systems". In: *arXiv e-prints* (Mar. 2020). URL: `https://arxiv.org/abs/2003.14240`.

Bird's eye view of this work

- In this work, we combine two methodologies: the LfD methodology from [5] and the data-driven control methodology from [6].
- The LfD methodology leverages the fact that the solution set of a linear system is a vector space to construct a stabilizing control law.
  - Results in [5] rely on the assumption that we have complete knowledge of the system.

---

[5] Alimzhan Sultangazin et al. *Watch and Learn: Learning to control feedback linearizable systems from expert demonstrations*. Tech. rep. UCLA, 2021. URL: http://www.cyphylab.ee.ucla.edu/Home/publications/UCLA-CyPhyLab-2021-03.pdf.

[6] Lucas Fraile, Matteo Marchi, and Paulo Tabuada. "Data-driven Stabilization of SISO Feedback Linearizable Systems". In: *arXiv e-prints* (Mar. 2020). URL: https://arxiv.org/abs/2003.14240.

Bird's eye view of this work

- In this work, we combine two methodologies: the LfD methodology from [5] and the data-driven control methodology from [6].
- The LfD methodology leverages the fact that the solution set of a linear system is a vector space to construct a stabilizing control law.
  - Results in [5] rely on the assumption that we have complete knowledge of the system.
- The data-driven control methodology provides a method for stabilizing unknown feedback linearizable SISO systems with standard linear control techniques and sufficiently fast sampling rates.
  - Data-driven control allows us to relax the assumption on the knowledge of the system.

---

[5] Alimzhan Sultangazin et al. *Watch and Learn: Learning to control feedback linearizable systems from expert demonstrations*. Tech. rep. UCLA, 2021. URL: http://www.cyphylab.ee.ucla.edu/Home/publications/UCLA-CyPhyLab-2021-03.pdf.

[6] Lucas Fraile, Matteo Marchi, and Paulo Tabuada. "Data-driven Stabilization of SISO Feedback Linearizable Systems". In: *arXiv e-prints* (Mar. 2020). URL: https://arxiv.org/abs/2003.14240.

Related work (Policy-learning algorithms)

- Many LfD methods assume there exists a mapping from state to control input that dictates the expert's behaviour, i.e., the expert's policy.

[7] Dean A. Pomerleau. "ALVINN: An Autonomous Land Vehicle in a Neural Network". In: *NIPS '88*. San Francisco, CA, USA, 1989.

[8] Pieter Abbeel, Adam Coates, and Andrew Y. Ng. "Autonomous Helicopter Aerobatics through Apprenticeship Learning". In: *The International Journal of Robotics Research* 29.13 (2010), pp. 1608–1639. DOI: 10.1177/0278364910371999.

[9] Elia Kaufmann et al. "Deep Drone Acrobatics". In: *arXiv e-prints* (2020). URL: https://arxiv.org/abs/2006.05768.

[10] F. Codevilla et al. "End-to-End Driving Via Conditional Imitation Learning". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 4693–4700. DOI: 10.1109/ICRA.2018.8460487.

[11] Malayandi Palan et al. "Fitting a Linear Control Policy to Demonstrations with a Kalman Constraint". In: *arXiv e-prints,* arXiv:2001.07572 (Jan. 2020).

Related work (Policy-learning algorithms)

- Many LfD methods assume there exists a mapping from state to control input that dictates the expert's behaviour, i.e., the expert's policy.
- In many ML-based LfD methods, policy learning is viewed as a supervised-learning problem (e.g., [7], [8], [9])

[7] Dean A. Pomerleau. "ALVINN: An Autonomous Land Vehicle in a Neural Network". In: *NIPS '88*. San Francisco, CA, USA, 1989.

[8] Pieter Abbeel, Adam Coates, and Andrew Y. Ng. "Autonomous Helicopter Aerobatics through Apprenticeship Learning". In: *The International Journal of Robotics Research* 29.13 (2010), pp. 1608–1639. DOI: 10.1177/0278364910371999.

[9] Elia Kaufmann et al. "Deep Drone Acrobatics". In: *arXiv e-prints* (2020). URL: https://arxiv.org/abs/2006.05768.

[10] F. Codevilla et al. "End-to-End Driving Via Conditional Imitation Learning". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 4693–4700. DOI: 10.1109/ICRA.2018.8460487.

[11] Malayandi Palan et al. "Fitting a Linear Control Policy to Demonstrations with a Kalman Constraint". In: *arXiv e-prints*, arXiv:2001.07572 (Jan. 2020).

Related work (Policy-learning algorithms)

- Many LfD methods assume there exists a mapping from state to control input that dictates the expert's behaviour, i.e., the expert's policy.
- In many ML-based LfD methods, policy learning is viewed as a supervised-learning problem (e.g., [7], [8], [9])
- Issues with ML-based approaches: need many demonstrations, cannot recover from disturbances [10], few formal stability guarantees.

---

[7] Dean A. Pomerleau. "ALVINN: An Autonomous Land Vehicle in a Neural Network". In: *NIPS '88*. San Francisco, CA, USA, 1989.

[8] Pieter Abbeel, Adam Coates, and Andrew Y. Ng. "Autonomous Helicopter Aerobatics through Apprenticeship Learning". In: *The International Journal of Robotics Research* 29.13 (2010), pp. 1608–1639. DOI: 10.1177/0278364910371999.

[9] Elia Kaufmann et al. "Deep Drone Acrobatics". In: *arXiv e-prints* (2020). URL: https://arxiv.org/abs/2006.05768.

[10] F. Codevilla et al. "End-to-End Driving Via Conditional Imitation Learning". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 4693–4700. DOI: 10.1109/ICRA.2018.8460487.

[11] Malayandi Palan et al. "Fitting a Linear Control Policy to Demonstrations with a Kalman Constraint". In: *arXiv e-prints,* arXiv:2001.07572 (Jan. 2020).

Related work (Policy-learning algorithms)

- Many LfD methods assume there exists a mapping from state to control input that dictates the expert's behaviour, i.e., the expert's policy.
- In many ML-based LfD methods, policy learning is viewed as a supervised-learning problem (e.g., [7], [8], [9])
- Issues with ML-based approaches: need many demonstrations, cannot recover from disturbances [10], few formal stability guarantees.
- Control-theoretic approaches: the work in [11] is conceptually the closest to ours, but we do not assume the expert's policy to be linear.

[7] Dean A. Pomerleau. "ALVINN: An Autonomous Land Vehicle in a Neural Network". In: *NIPS '88*. San Francisco, CA, USA, 1989.

[8] Pieter Abbeel, Adam Coates, and Andrew Y. Ng. "Autonomous Helicopter Aerobatics through Apprenticeship Learning". In: *The International Journal of Robotics Research* 29.13 (2010), pp. 1608–1639. DOI: 10.1177/0278364910371999.

[9] Elia Kaufmann et al. "Deep Drone Acrobatics". In: *arXiv e-prints* (2020). URL: https://arxiv.org/abs/2006.05768.

[10] F. Codevilla et al. "End-to-End Driving Via Conditional Imitation Learning". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 4693–4700. DOI: 10.1109/ICRA.2018.8460487.

[11] Malayandi Palan et al. "Fitting a Linear Control Policy to Demonstrations with a Kalman Constraint". In: *arXiv e-prints*, arXiv:2001.07572 (Jan. 2020).

Problem Statement and Preliminaries

Consider a continuous-time control-affine system:

$$\Sigma : \quad \dot{x} = f(x) + g(x)u, \quad y = h(x). \tag{1}$$

Problem Statement and Preliminaries

Consider a continuous-time control-affine system:

$$\Sigma: \quad \dot{x} = f(x) + g(x)u, \quad y = h(x). \tag{1}$$

- Triple $(x, u, y) : \mathbb{R}_0^+ \to \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}$: a solution of the system (1).
- Functions x, u, and y are trajectory, control input, and output of (1).

Problem Statement and Preliminaries

Consider a continuous-time control-affine system:

$$\Sigma : \quad \dot{x} = f(x) + g(x)u, \quad y = h(x). \tag{1}$$

- Triple $(x, u, y) : \mathbb{R}_0^+ \to \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}$: a solution of the system (1).
- Functions x, u, and y are trajectory, control input, and output of (1).
- Given a sampling period $T > 0$, assume u is constant over sampling intervals and y is measured at the sampling instants.

Problem Statement and Preliminaries

Consider a continuous-time control-affine system:

$$\Sigma : \quad \dot{x} = f(x) + g(x)u, \quad y = h(x). \tag{1}$$

- Triple $(x, u, y) : \mathbb{R}_0^+ \to \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}$: a solution of the system (1).
- Functions x, u, and y are trajectory, control input, and output of (1).
- Given a sampling period $T > 0$, assume u is constant over sampling intervals and y is measured at the sampling instants.
- For a continuous-time signal v, its corresponding sampled-data signal v_s is defined by $v_s(k) \triangleq v(kT)$.

Problem Statement and Preliminaries

Consider a continuous-time control-affine system:

$$\Sigma: \quad \dot{x} = f(x) + g(x)u, \quad y = h(x). \tag{1}$$

- Triple $(x, u, y) : \mathbb{R}_0^+ \to \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}$: a solution of the system (1).
- Functions x, u, and y are trajectory, control input, and output of (1).
- Given a sampling period $T > 0$, assume u is constant over sampling intervals and y is measured at the sampling instants.
- For a continuous-time signal v, its corresponding sampled-data signal v_s is defined by $v_s(k) \triangleq v(kT)$.

Definition

A controller $\kappa : \mathbb{R}^n \to \mathbb{R}^m$ is *asymptotically stabilizing* for system (1) if the origin is uniformly asymptotically stable[1] for the system (1) with $u = \kappa(x)$.

[] [1] that is, there exists a class $\mathcal{KL}$ function[1] β such that $\|x(t)\| \leq \beta(\|x(t_0)\|, t - t_0)$, $\forall t \geq t_0 \geq 0$.

Problem Statement and Preliminaries

- Suppose there exists an unknown asymptotically stabilizing controller $u = \kappa(x)$ — the expert controller.

Problem Statement and Preliminaries

- Suppose there exists an unknown asymptotically stabilizing controller $u = \kappa(x)$ — the expert controller.
- Assume the expert produces $n+1$ expert solutions of system (1) with length $\tau \in \mathbb{R}$:

$$\mathcal{D} = \{(x^i, u^i, y^i)\}_{i=1}^{n+1}, \text{ with } (x^i, u^i, y^i) : [0, \tau] \to \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}.$$

  We collect the corresponding set of measurement samples:

$$\mathcal{D}_s = \{y_s^i\}_{i=1}^{n+1}, \quad y_s^i(k) \triangleq y^i(kT). \tag{2}$$

Problem Statement and Preliminaries

- Suppose there exists an unknown asymptotically stabilizing controller $u = \kappa(x)$ — the expert controller.
- Assume the expert produces $n+1$ expert solutions of system (1) with length $\tau \in \mathbb{R}$:

$$\mathcal{D} = \{(x^i, u^i, y^i)\}_{i=1}^{n+1}, \text{ with } (x^i, u^i, y^i) : [0, \tau] \to \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}.$$

  We collect the corresponding set of measurement samples:

$$\mathcal{D}_s = \{y_s^i\}_{i=1}^{n+1}, \quad y_s^i(k) \triangleq y^i(kT). \tag{2}$$

- Assume the system (1) is feedback linearizable with h having a relative degree n, i.e., for all $x \in \mathbb{R}^n$:

$$L_g L_f^i h(x) = 0, \quad i = 0, \ldots, n-2, \tag{3}$$

$$L_g L_f^{n-1} h(x) \neq 0. \tag{4}$$

Problem Statement and Preliminaries

- Suppose there exists an unknown asymptotically stabilizing controller $u = \kappa(x)$ — the expert controller.
- Assume the expert produces $n+1$ expert solutions of system (1) with length $\tau \in \mathbb{R}$:

$$\mathcal{D} = \{(x^i, u^i, y^i)\}_{i=1}^{n+1}, \text{ with } (x^i, u^i, y^i) : [0, \tau] \to \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}.$$

  We collect the corresponding set of measurement samples:

$$\mathcal{D}_s = \{y_s^i\}_{i=1}^{n+1}, \quad y_s^i(k) \triangleq y^i(kT). \tag{2}$$

- Assume the system (1) is feedback linearizable with h having a relative degree n, i.e., for all $x \in \mathbb{R}^n$:

$$L_g L_f^i h(x) = 0, \quad i = 0, \ldots, n-2, \tag{3}$$

$$L_g L_f^{n-1} h(x) \neq 0. \tag{4}$$

- In addition, assume w.l.o.g. that $L_g L_f^{n-1} h > 0$.

Problem Statement and Preliminaries

- Our goal is to use the set of measurement samples $\mathcal{D}_s$ to construct a controller that is guaranteed to stabilize the system (1).

Problem Statement and Preliminaries

- Our goal is to use the set of measurement samples $\mathcal{D}_s$ to construct a controller that is guaranteed to stabilize the system (1).
- For clarity of exposition, in what follows we assume $n = 2$.

Problem Statement and Preliminaries

- Our goal is to use the set of measurement samples $\mathcal{D}_s$ to construct a controller that is guaranteed to stabilize the system (1).
- For clarity of exposition, in what follows we assume $n = 2$.
- We can rewrite the nonlinear system dynamics (1) in the coordinates:

$$z = \Phi(x) = \begin{bmatrix} h(x) & L_f h(x) \end{bmatrix}^T, \tag{5}$$

and get:

$$\begin{aligned} \dot{z}_1 &= z_2, \\ \dot{z}_2 &= \alpha(z) + \beta(z)u = w, \\ y &= z_1 \end{aligned} \tag{6}$$

where $\alpha = (L_f^2 h) \circ \Phi^{-1}$, $\beta = (L_g L_f h) \circ \Phi^{-1}$, and $w \triangleq \alpha(z) + \beta(z)u$.

Learning from demonstrations when dynamics are known

Constructing the learned controller

- For now, assume that we know functions α and β and are given the set $\mathcal{D}_e = \{(z^i, w^i)\}$, where $z^i = \Phi(x^i)$ and $w^i = \alpha(z^i) + \beta(z^i)u^i$.

Constructing the learned controller

- For now, assume that we know functions α and β and are given the set $\mathcal{D}_e = \{(z^i, w^i)\}$, where $z^i = \Phi(x^i)$ and $w^i = \alpha(z^i) + \beta(z^i)u^i$.
- Construct the following matrices:

$$Z(t) \triangleq \left[z^2(t) - z^1(t) \,\middle|\, \cdots \,\middle|\, z^{n+1}(t) - z^1(t)\right] \tag{7}$$

$$W(t) \triangleq \left[w^2(t) - w^1(t) \,\middle|\, \cdots \,\middle|\, w^{n+1}(t) - w^1(t)\right]. \tag{8}$$

Constructing the learned controller

- For now, assume that we know functions α and β and are given the set $\mathcal{D}_e = \{(z^i, w^i)\}$, where $z^i = \Phi(x^i)$ and $w^i = \alpha(z^i) + \beta(z^i)u^i$.
- Construct the following matrices:

$$Z(t) \triangleq \left[z^2(t) - z^1(t) \,\middle|\, \cdots \,\middle|\, z^{n+1}(t) - z^1(t)\right] \tag{7}$$

$$W(t) \triangleq \left[w^2(t) - w^1(t) \,\middle|\, \cdots \,\middle|\, w^{n+1}(t) - w^1(t)\right]. \tag{8}$$

Lemma (Affine comb. of inputs ⇒ Affine comb. of trajectories)

- *Suppose we are given a set of finite-length solutions* $\{(z^i, w^i)\}_{i=1}^{n+1}$ *of the system (6), where each* (z^i, w^i) *is defined for* $0 \leq t \leq \tau$, $\tau \in \mathbb{R}$.
- *Assume that* $\{z^i(0)\}_{i=1}^{n+1}$ *is an affinely independent set.*

Constructing the learned controller

- For now, assume that we know functions α and β and are given the set $\mathcal{D}_e = \{(z^i, w^i)\}$, where $z^i = \Phi(x^i)$ and $w^i = \alpha(z^i) + \beta(z^i)u^i$.
- Construct the following matrices:

$$Z(t) \triangleq \left[z^2(t) - z^1(t) \,\middle|\, \cdots \,\middle|\, z^{n+1}(t) - z^1(t)\right] \tag{7}$$

$$W(t) \triangleq \left[w^2(t) - w^1(t) \,\middle|\, \cdots \,\middle|\, w^{n+1}(t) - w^1(t)\right]. \tag{8}$$

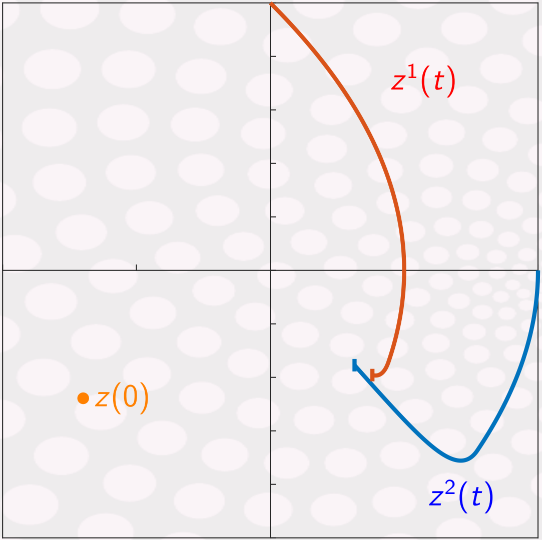
Lemma (Affine comb. of inputs ⇒ Affine comb. of trajectories)

- *Suppose we are given a set of finite-length solutions $\{(z^i, w^i)\}_{i=1}^{n+1}$ of the system (6), where each (z^i, w^i) is defined for $0 \leq t \leq \tau$, $\tau \in \mathbb{R}$.*
- *Assume that $\{z^i(0)\}_{i=1}^{n+1}$ is an affinely independent set.*

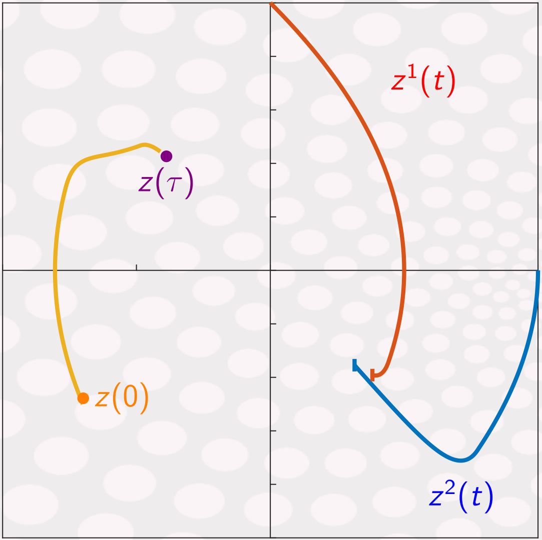
Then, for $t_0 \leq t \leq \tau + t_0$, under the input $w(t) = W(t - t_0)\zeta$ with $\zeta = Z^{-1}(0)z_0$, the solution of the system (6) with $z(t_0) = z_0$ is:

$$z(t) = Z(t - t_0)\zeta.$$

Constructing the learned controller

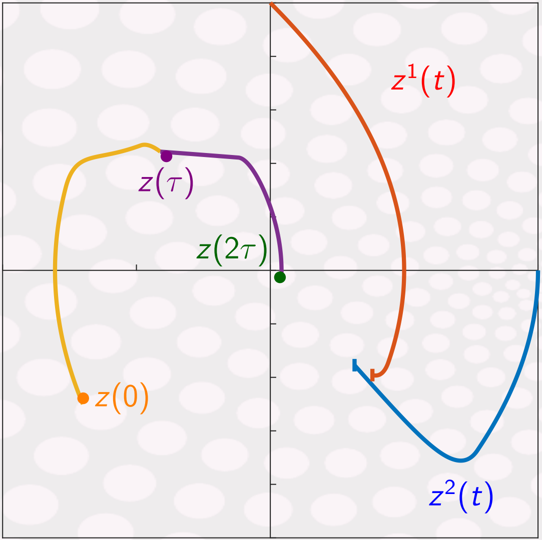


Constructing the learned controller



1. For $t \in [0, \tau)$,
$w(t) = W(t)\zeta(0)$ with
$\zeta(0) = Z^{-1}(0)z(0)$

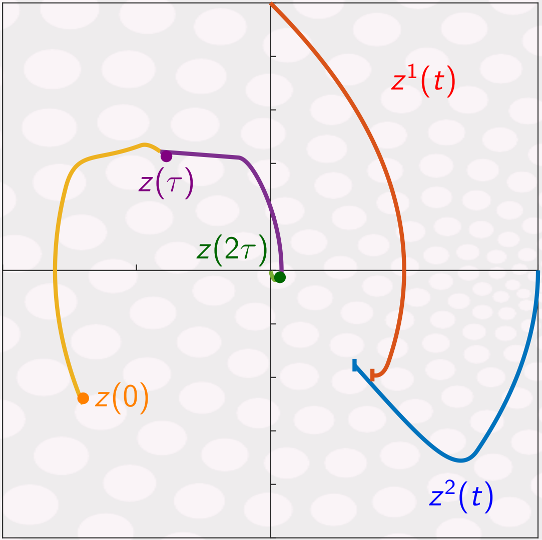
Constructing the learned controller



1. For $t \in [0, \tau)$,
$w(t) = W(t)\zeta(0)$ with
$\zeta(0) = Z^{-1}(0)z(0)$

2. For $t \in [\tau, 2\tau)$,
$w(t) = W(t-\tau)\zeta(1)$ with
$\zeta(1) = Z^{-1}(0)z(\tau)$

Constructing the learned controller



1. For $t \in [0, \tau)$,
$w(t) = W(t)\zeta(0)$ with
$\zeta(0) = Z^{-1}(0)z(0)$

2. For $t \in [\tau, 2\tau)$,
$w(t) = W(t-\tau)\zeta(1)$ with
$\zeta(1) = Z^{-1}(0)z(\tau)$

3. For $t \in [2\tau, 3\tau)$,
$w(t) = W(t-2\tau)\zeta(2)$ with
$\zeta(2) = Z^{-1}(0)z(2\tau)$

Constructing the learned controller

We apply the following preliminary controller:

$$u(t) = \beta^{-1}(z(t))(-\alpha(z(t)) + w(t, z(t))), \tag{9}$$

to bring the system (1) to the form $y^{(n)} = w$ and use the control law:

$$w(t, z(t)) = W(t - p\tau)Z^{-1}(t - p\tau)z(t) = K(t)z(t), \tag{10}$$

for all $t \in [p\tau, (p+1)\tau)$ and $p \in \mathbb{N}_0$.

Constructing the learned controller

We apply the following preliminary controller:

$$u(t) = \beta^{-1}(z(t))(-\alpha(z(t)) + w(t, z(t))), \tag{9}$$

to bring the system (1) to the form $y^{(n)} = w$ and use the control law:

$$w(t, z(t)) = W(t - p\tau)Z^{-1}(t - p\tau)z(t) = K(t)z(t), \tag{10}$$

for all $t \in [p\tau, (p+1)\tau)$ and $p \in \mathbb{N}_0$.

Lemma (Minimal length of demonstrations for stability)

- *Suppose a set $\mathcal{D}_e = \{(z^i, w^i)\}_{i=1}^{n+1}$ of solutions with length $\tau \in \mathbb{R}^+$ is generated by the system (6) in closed loop with an asymptotically stabilizing controller $u = \kappa(z)$.*
- *Assume that $\{z^i(t)\}_{i=1}^{n+1}$ is affinely independent for all $t \in [0, \tau]$.*

Then, there is $\bar{\tau} \in \mathbb{R}^+$ such that for all $\tau \geq \bar{\tau}$, the origin of the system (6) in closed loop with the controller in (10) is uniformly exponentially stable.

Learning from demonstrations when dynamics are unknown (using data-driven control)

Data-driven control and LfD for unknown systems

- Previous assumptions: know $\mathcal{D}_e = \{(z^i, w^i)\}_{i=1}^{n+1}$ and functions α and β.

[6] Fraile, Marchi, and Tabuada, 2020, "Data-driven Stabilization of SISO Feedback Linearizable Systems"

Data-driven control and LfD for unknown systems

- Previous assumptions: know $\mathcal{D}_e = \{(z^i, w^i)\}_{i=1}^{n+1}$ and functions α and β.
- The data-driven control from [6] consists of two components: state estimator and dynamic controller.

[6] Fraile, Marchi, and Tabuada, 2020, "Data-driven Stabilization of SISO Feedback Linearizable Systems"

Data-driven control and LfD for unknown systems

- Previous assumptions: know $\mathcal{D}_e = \{(z^i, w^i)\}_{i=1}^{n+1}$ and functions α and β.
- The data-driven control from [6] consists of two components: state estimator and dynamic controller.
- We use the state estimator from [6] to:
  - estimate the set $\widehat{\mathcal{D}}_e = \{(\widehat{z}_s^i, \widehat{w}_s^i)\}_{i=1}^{n+1}$ from the given data $\mathcal{D}_s = \{y_s^i\}_{i=1}^{n+1}$;

[6] Fraile, Marchi, and Tabuada, 2020, "Data-driven Stabilization of SISO Feedback Linearizable Systems"

Data-driven control and LfD for unknown systems

- Previous assumptions: know $\mathcal{D}_e = \{(z^i, w^i)\}_{i=1}^{n+1}$ and functions α and β.
- The data-driven control from [6] consists of two components: state estimator and dynamic controller.
- We use the state estimator from [6] to:
  - estimate the set $\widehat{\mathcal{D}}_e = \{(\widehat{z}_s^i, \widehat{w}_s^i)\}_{i=1}^{n+1}$ from the given data $\mathcal{D}_s = \{y_s^i\}_{i=1}^{n+1}$;
  - provide estimates of $\widehat{z}_s$ and $\widehat{w}_s$ to the dynamic controller.

[6] Fraile, Marchi, and Tabuada, 2020, "Data-driven Stabilization of SISO Feedback Linearizable Systems"

Data-driven control and LfD for unknown systems

- Previous assumptions: know $\mathcal{D}_e = \{(z^i, w^i)\}_{i=1}^{n+1}$ and functions α and β.
- The data-driven control from [6] consists of two components: state estimator and dynamic controller.
- We use the state estimator from [6] to:
  - estimate the set $\widehat{\mathcal{D}}_e = \{(\widehat{z}_s^i, \widehat{w}_s^i)\}_{i=1}^{n+1}$ from the given data $\mathcal{D}_s = \{y_s^i\}_{i=1}^{n+1}$;
  - provide estimates of $\widehat{z}_s$ and $\widehat{w}_s$ to the dynamic controller.
- The dynamic controller from [6] tracks the virtual input w from the learned controller (10) without knowing α and β.

[6] Fraile, Marchi, and Tabuada, 2020, "Data-driven Stabilization of SISO Feedback Linearizable Systems"

The main result

Theorem

Consider an unknown SISO system (1) *where h has a relative degree n.*

- *Let T be the sampling time and τ be the demonstration length.*
- *Suppose we are given $\mathcal{D}_s = \{y_s^i\}_{i=1}^{n+1}$ generated by the system* (1) *in closed loop with a stabilizing expert. and the state estimator from [6] is used to construct $\widehat{\mathcal{D}}_e = \left\{\left(\widehat{z}_s^i, \widehat{w}_s^i\right)\right\}_{i=1}^{n+1}$.*

Then, there exist $\bar{T} \in \mathbb{R}^+$ and $\bar{\tau} \in \mathbb{R}^+$ so that, for any $T \in [0, \bar{T}]$ and any $\tau \geq \bar{\tau}$, the dynamic controller, based on the learned controller (10), *renders the closed-loop solutions bounded and $\lim_{t\to\infty} x(t) = 0$.*

Conclusion

- We have extended a methodology in [5] for constructing a stabilizing controller from expert demonstrations to unknown SISO systems.

[5] Sultangazin et al., 2021, *Watch and Learn: Learning to control feedback linearizable systems from expert demonstrations*

Conclusion

- We have extended a methodology in [5] for constructing a stabilizing controller from expert demonstrations to unknown SISO systems.
- Compared to machine-learning approaches, this methodology requires few demonstrations (i.e., the minimal number of demonstrations is $n + 1$) and provides formal stability guarantees.

[5] Sultangazin et al., 2021, *Watch and Learn: Learning to control feedback linearizable systems from expert demonstrations*

Conclusion

- We have extended a methodology in [5] for constructing a stabilizing controller from expert demonstrations to unknown SISO systems.
- Compared to machine-learning approaches, this methodology requires few demonstrations (i.e., the minimal number of demonstrations is $n + 1$) and provides formal stability guarantees.
- As part of future work, we plan to:
  - apply a similar methodology to learn control of unknown MIMO systems;
  - experimentally verify this methodology using the testbed with Crazyflie quadrotors we have in our laboratory.

[5] Sultangazin et al., 2021, *Watch and Learn: Learning to control feedback linearizable systems from expert demonstrations*