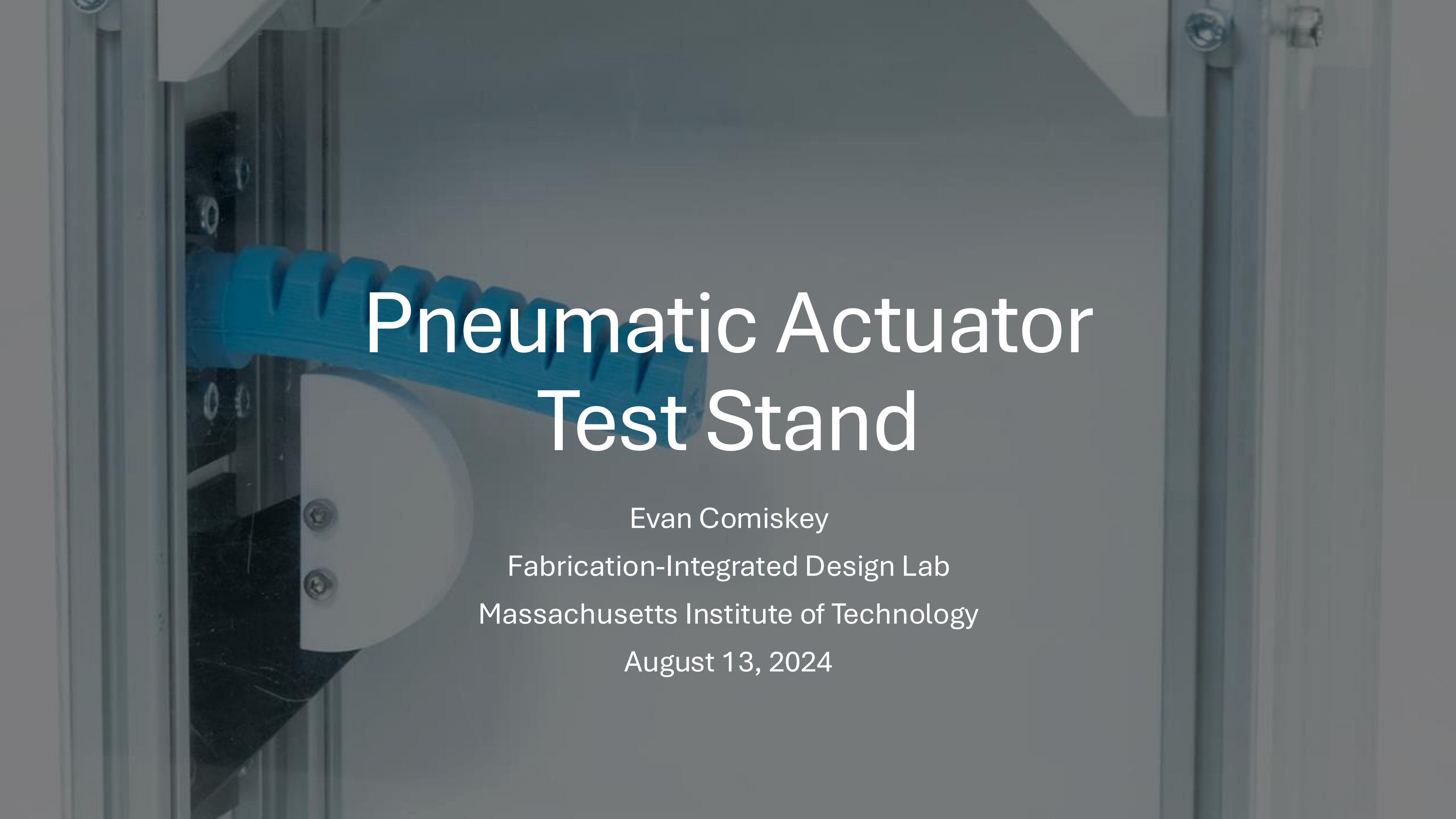


Pneumatic Actuator Test Stand

Evan Comiskey
Fabrication-Integrated Design Lab
Massachusetts Institute of Technology

August 13, 2024

Agenda

Motivation

Hardware

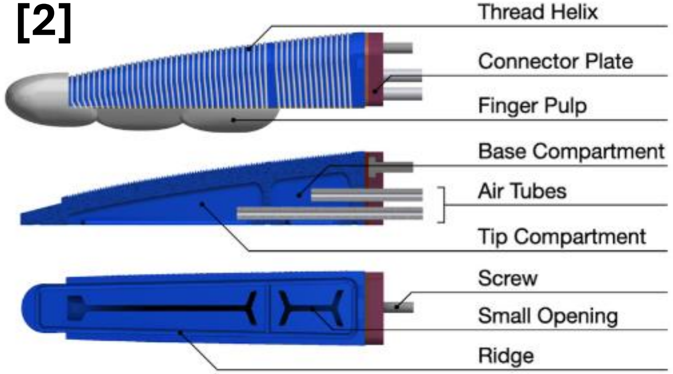
Control Software

Many new actuator designs...

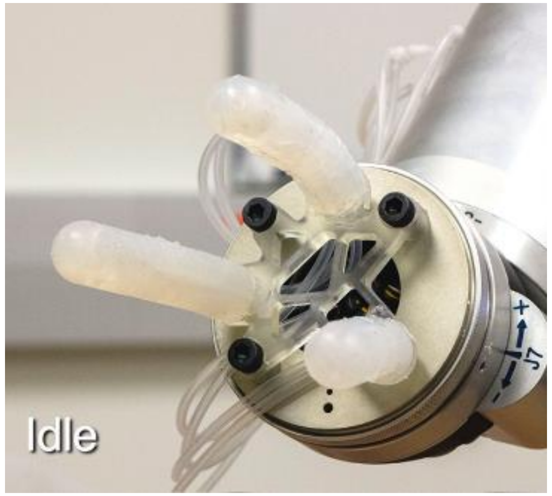
[1]

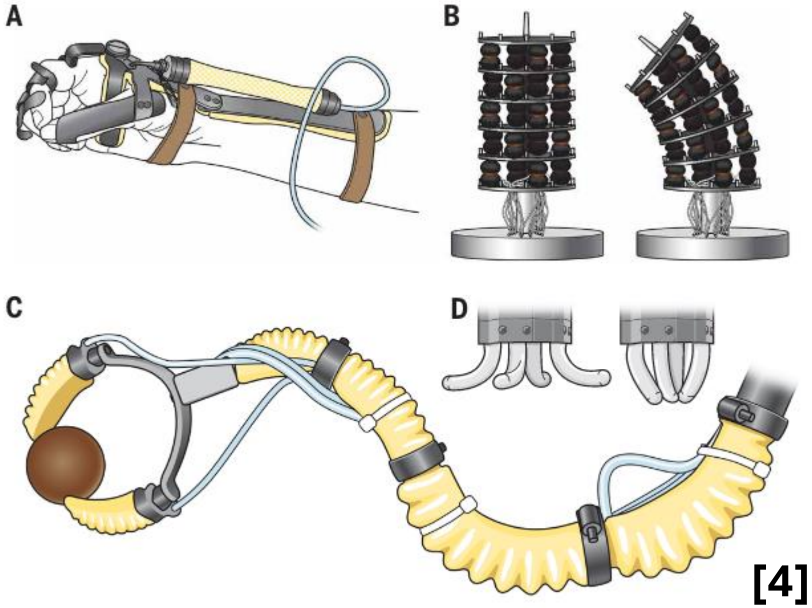
Candidate	Extrusion Cross Section	Isometric View	Parameter Values	Target Curvature Actuation Volume $[mm^3]$	Target Curvature Max Strain [-]
Baseline			Baseline Parameters	18747	1.23
DC5			T = 1.5 mm HV = 1.0 HB = 3.0	8995	0.49
DC6			T = 1.5 mm HV = 2.0 HB = 2.0	8462	0.50
DC7			T = 1.5 mm HV = 1.0 HB = 2.0	7605	0.62
DC8			T = 1.5 mm HV = 2.0 mm HB = 1.0 mm	7115	0.67
DC9			N = 2	1175	0.81

[2]



[3]





[4]

References listed on last slide

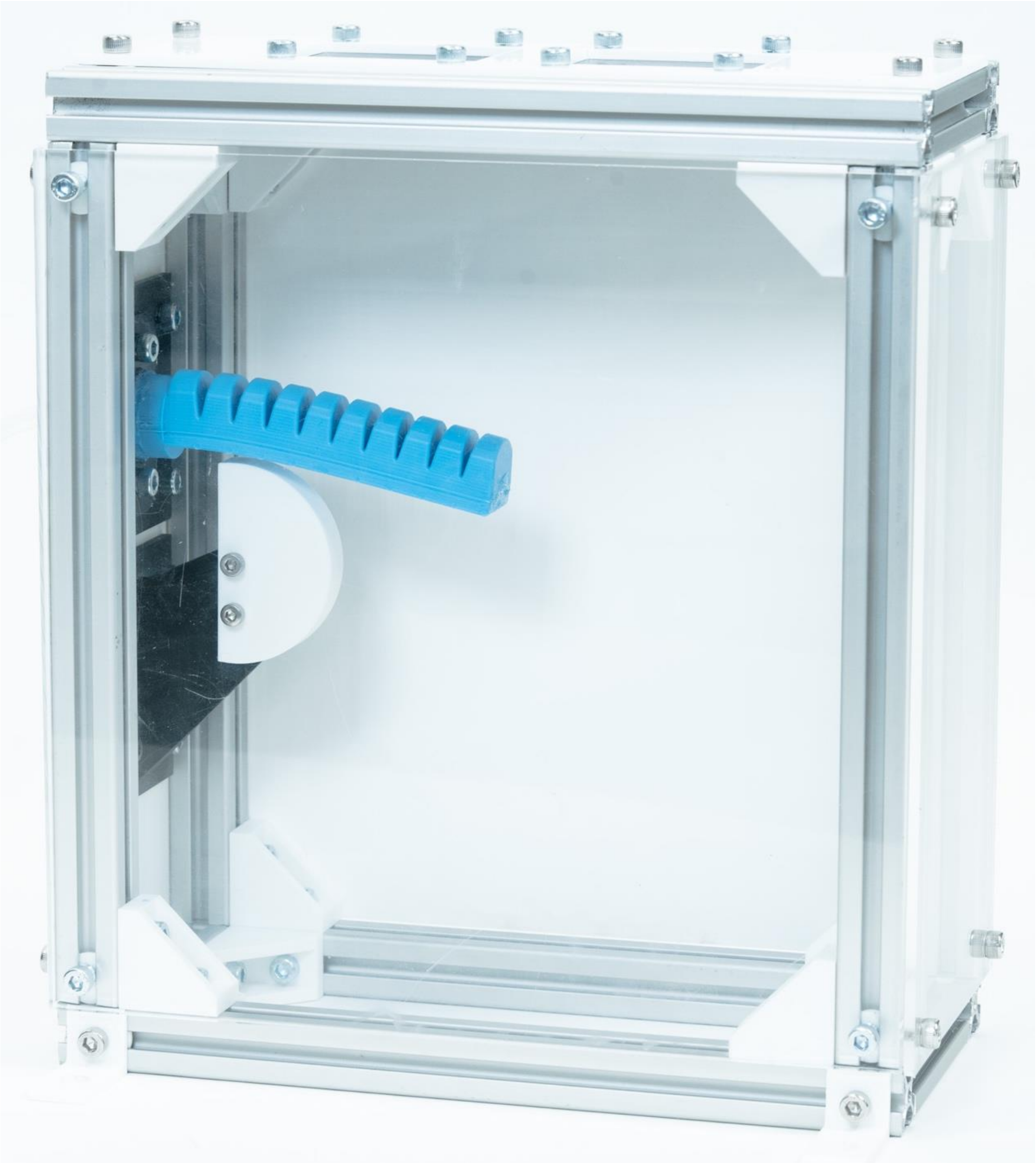
…But lack of unified testing standards

[5] One key issue seen in the literature is incomplete reporting of test methods and testing equipment. This makes it **difficult to fully understand and trust** the reported results, to **replicate** the tests, or to **extend or advance the research**

[6] **Soft robotics lacks standard** benchmarks, metrics, data sets, measurements and characterization workflows, and manufacturing recipes

References listed on last slide

This work: hardware and software for testing pneumatic actuators



Goal: an **open-source**, **easy-to-use** test stand with **repeatable** data collection for **demonstrating** actuator designs decisions

Goal: an **open-source**, **easy-to-use** test stand with **repeatable** data collection for **demonstrating** actuator designs decisions

Make all aspects publicly available so others may use, modify, and expand upon

Goal: an **open-source**, **easy-to-use** test stand with **repeatable** data collection for **demonstrating** actuator designs decisions

Allows reconfiguration without redesign for new actuators, valves, etc.

In-depth documentation and code commenting makes pneumatic controller customization accessible to non-programmers

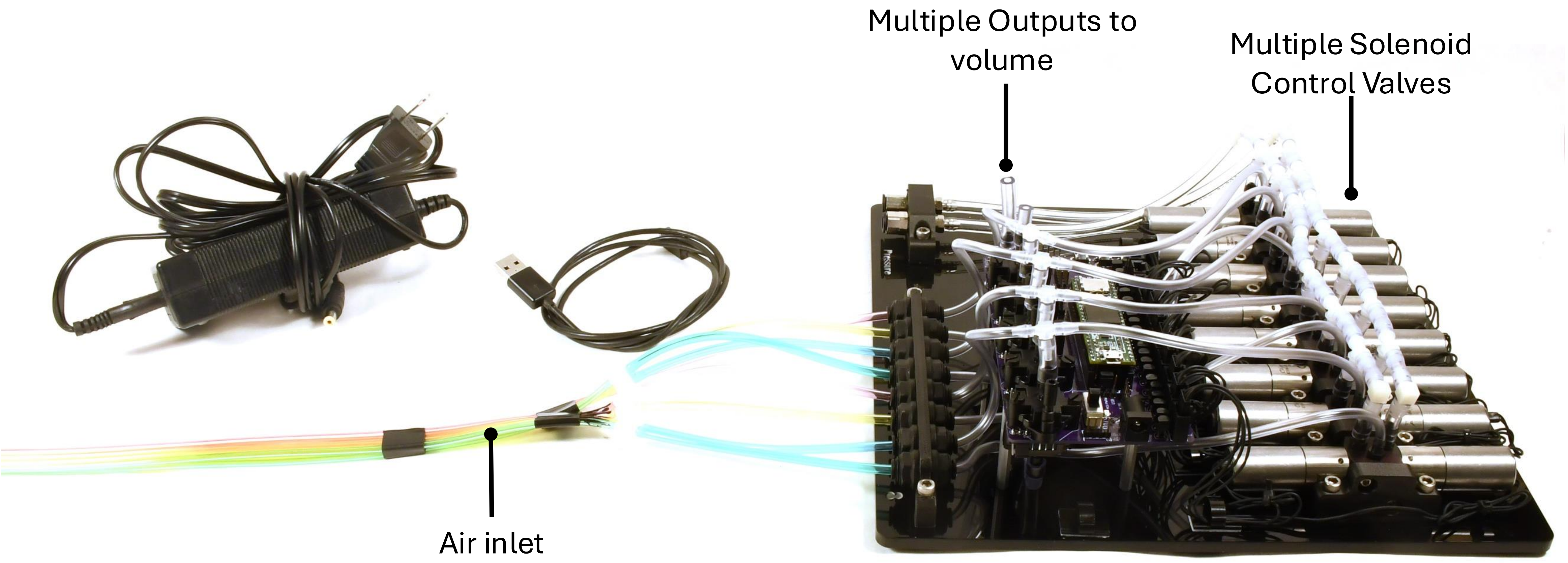
Goal: an **open-source**, **easy-to-use** test stand with **repeatable** data collection for **demonstrating** actuator designs decisions

Previous two points grant ability for different labs to recreate the same data

Goal: an **open-source**, **easy-to-use** test stand with **repeatable** data collection for **demonstrating** actuator designs decisions

Setup allows for comparing and validating various actuator design decisions

Prior work



Citation: Clarke Teeple, https://ctrl-p.cbteeple.com/latest/index.html

Agenda

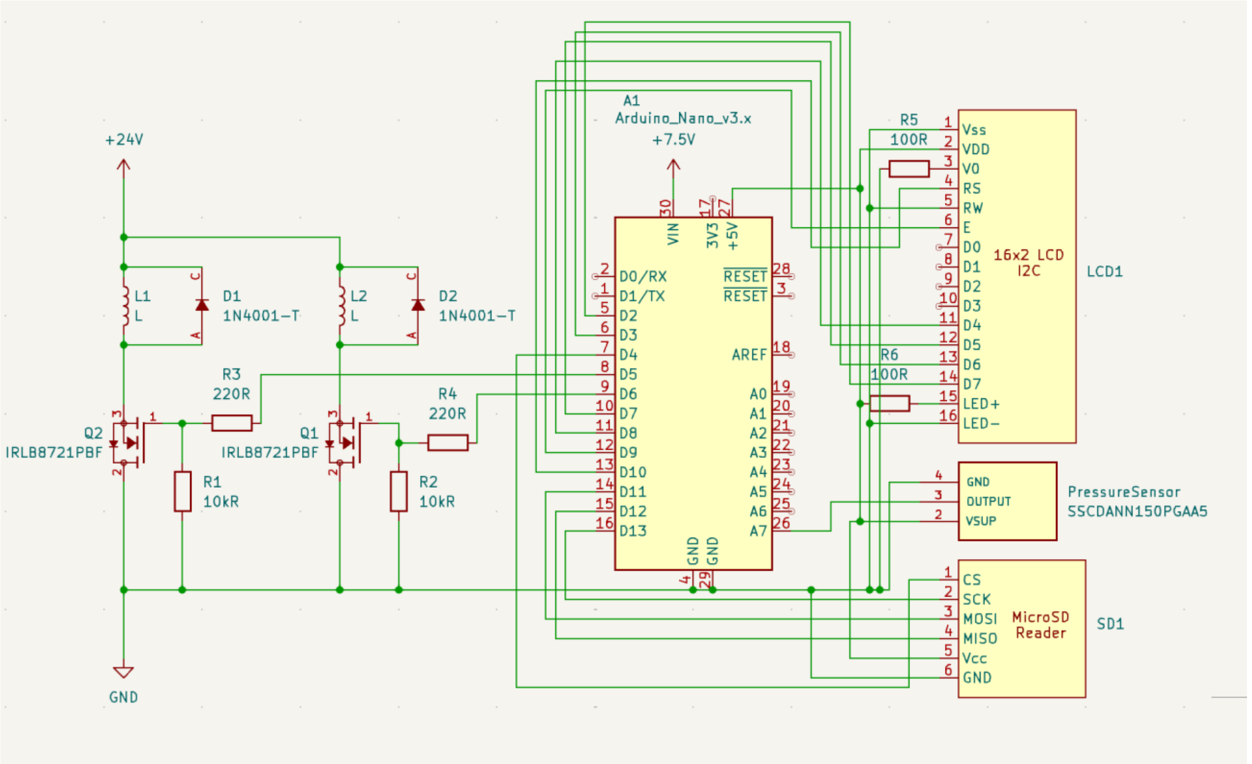
Motivation

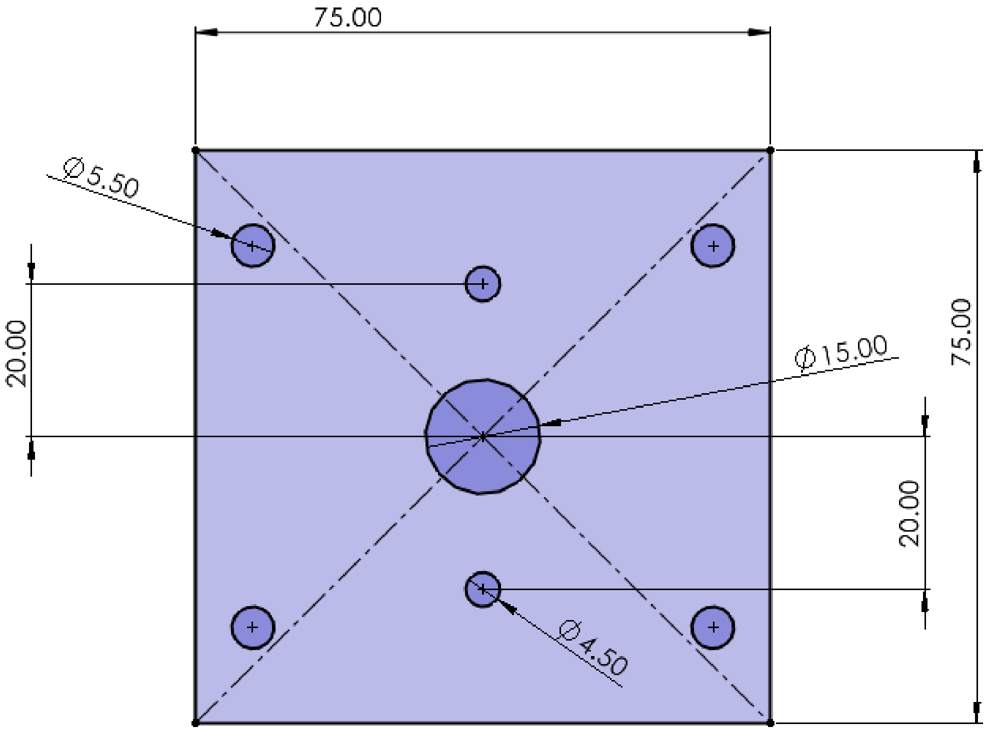
Hardware

Control Software

Open-source hardware

Schematics of all hardware and electronics used to be publicly available for others to recreate and adapt for their own experiments

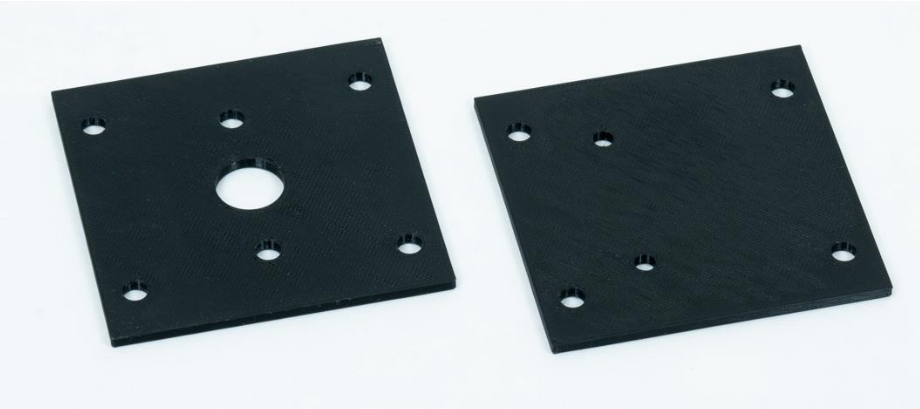


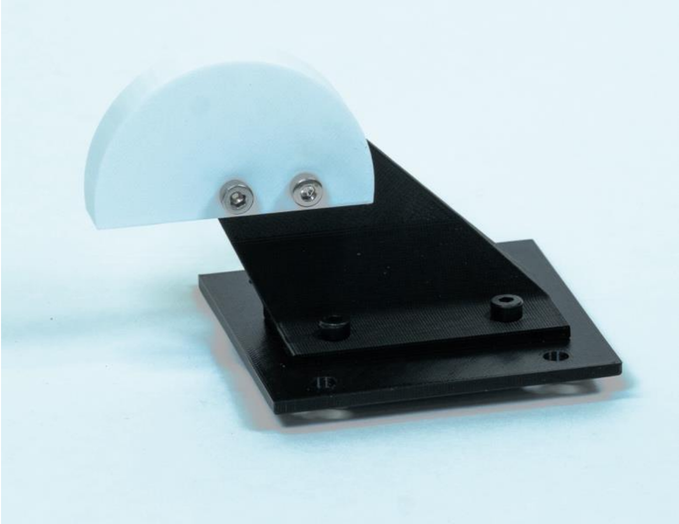


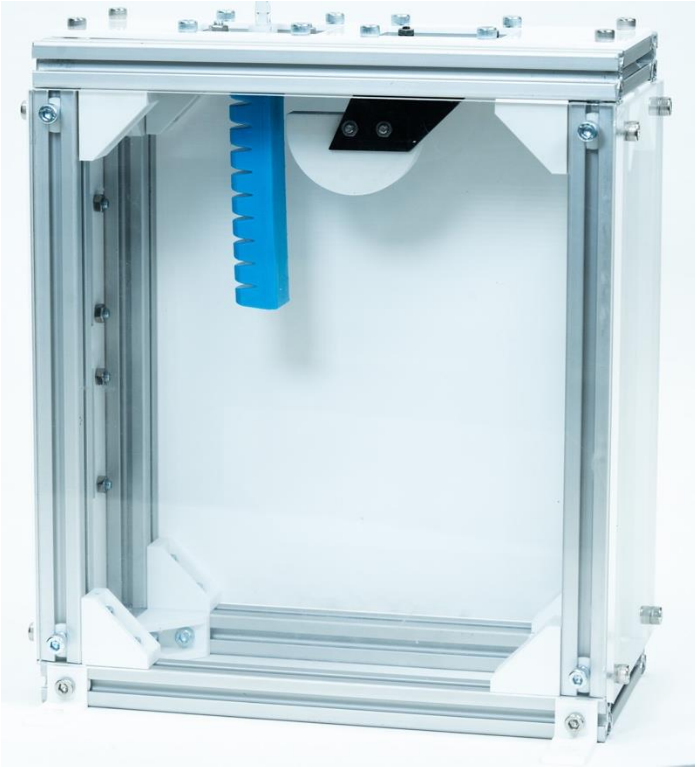
Modularity

All mounting equipment for sample objects and actuators can be fabricated using 3d printing

Allows stand to accommodate any actuator design or desired test set-up in either horizontal or vertical orientation

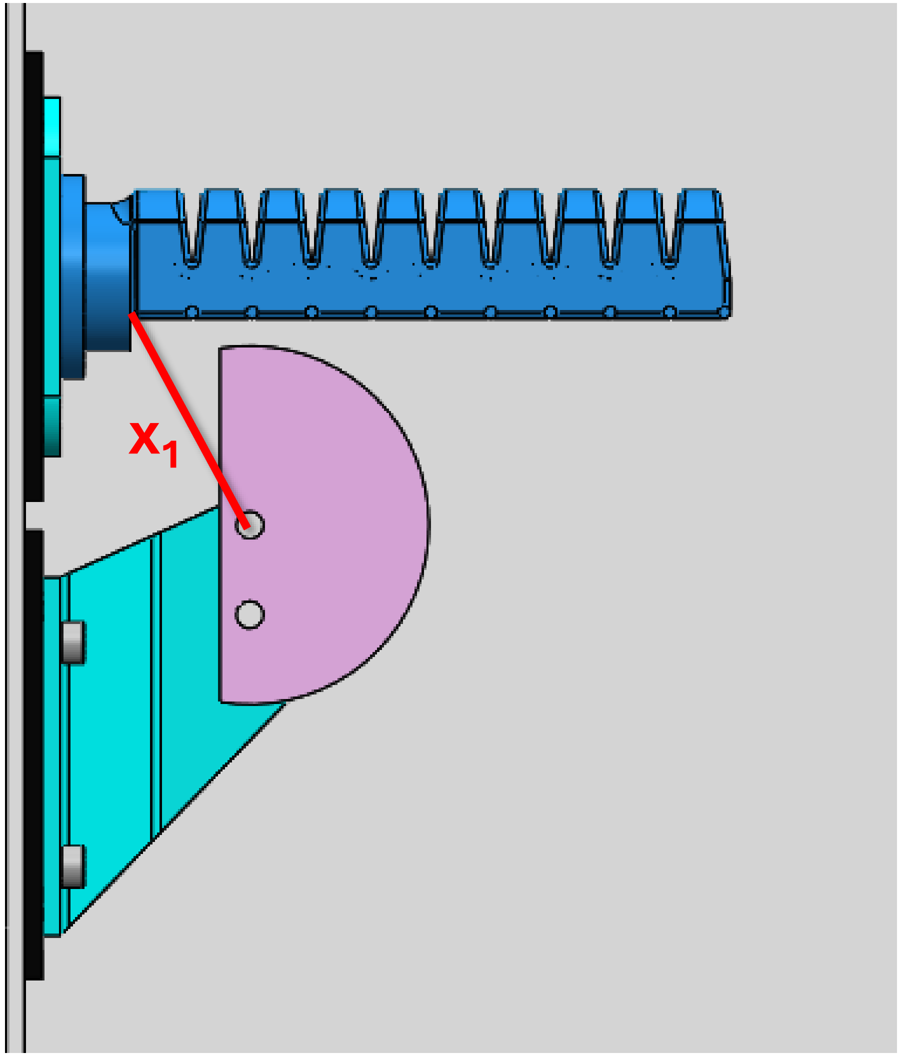






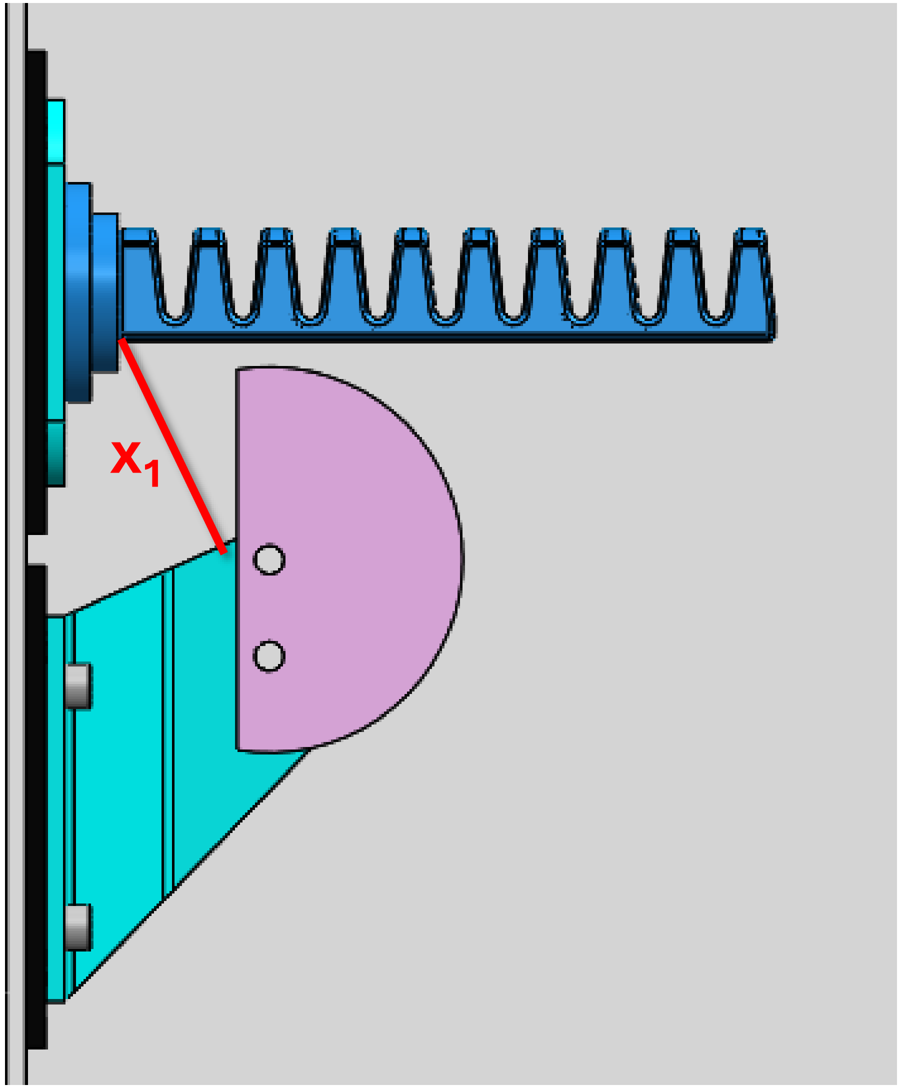
Easy to work with variables of interest

Flexibility in modularity makes isolating any physical variable between different test conditions simple



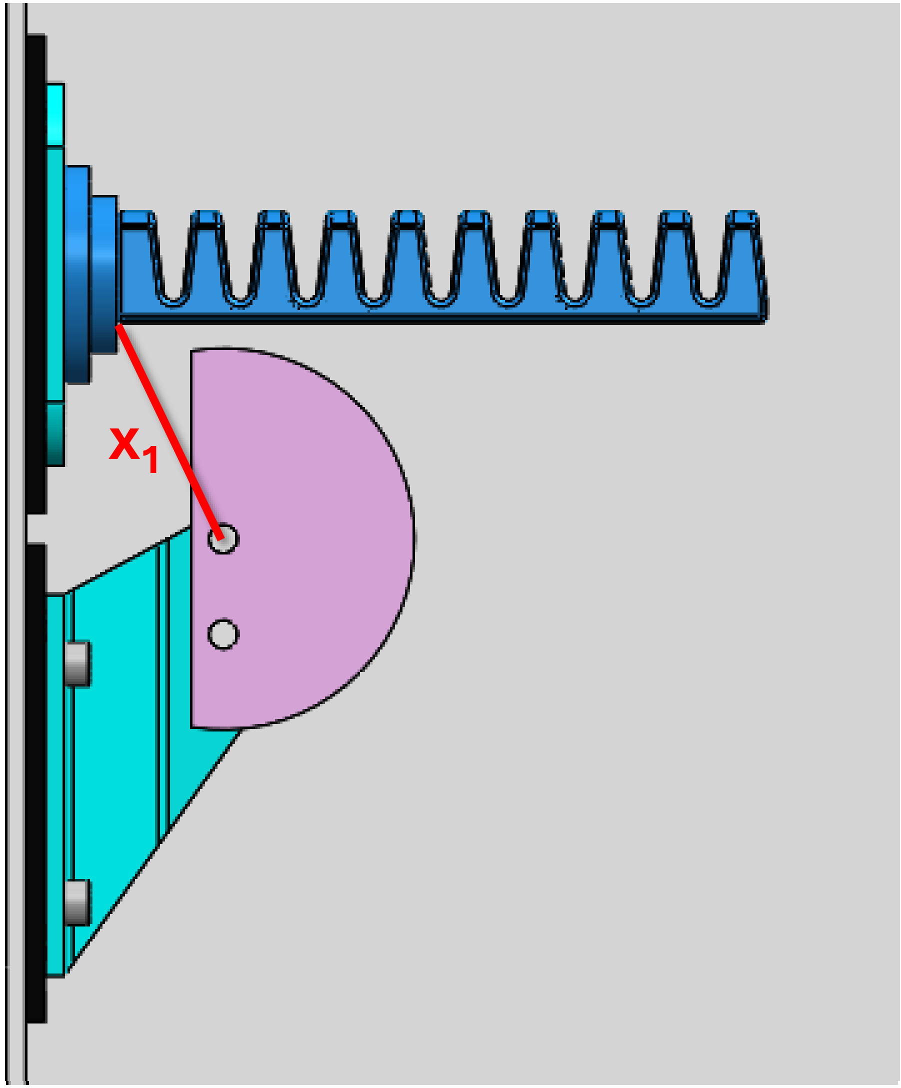
Easy to work with variables of interest cont.

So, if an engineer wants to change some physical design parameter…

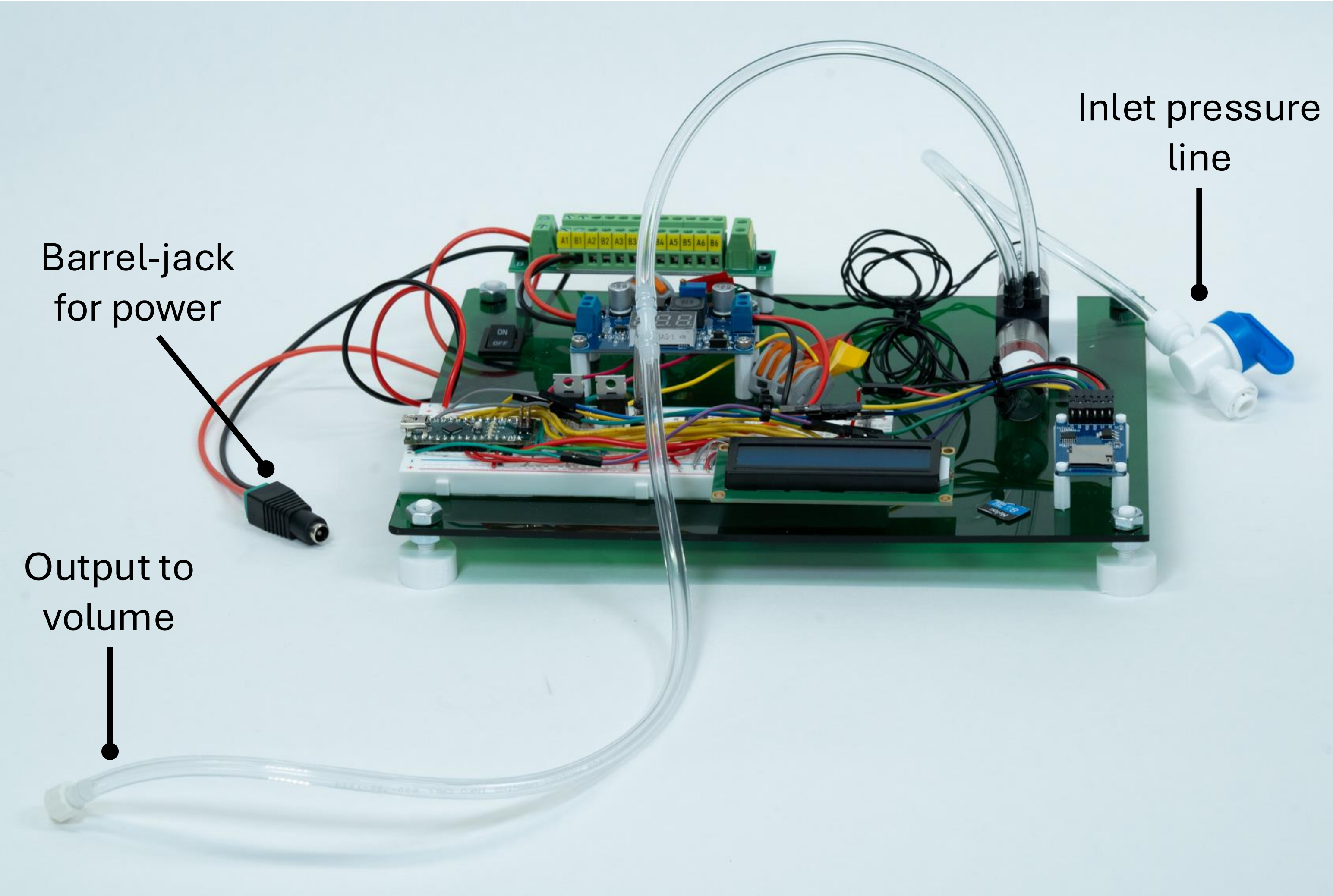


Easy to work with variables of interest cont.

…they can hold relevant variables constant through software adjustments, and quickly implement corresponding hardware changes via 3D printing



Electronics & Pneumatics



Agenda

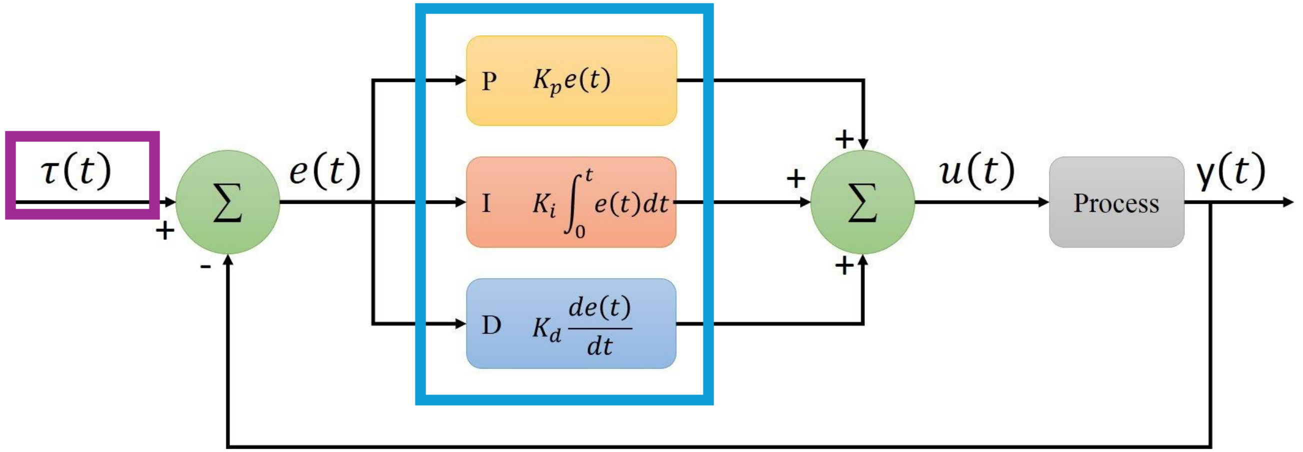
Motivation

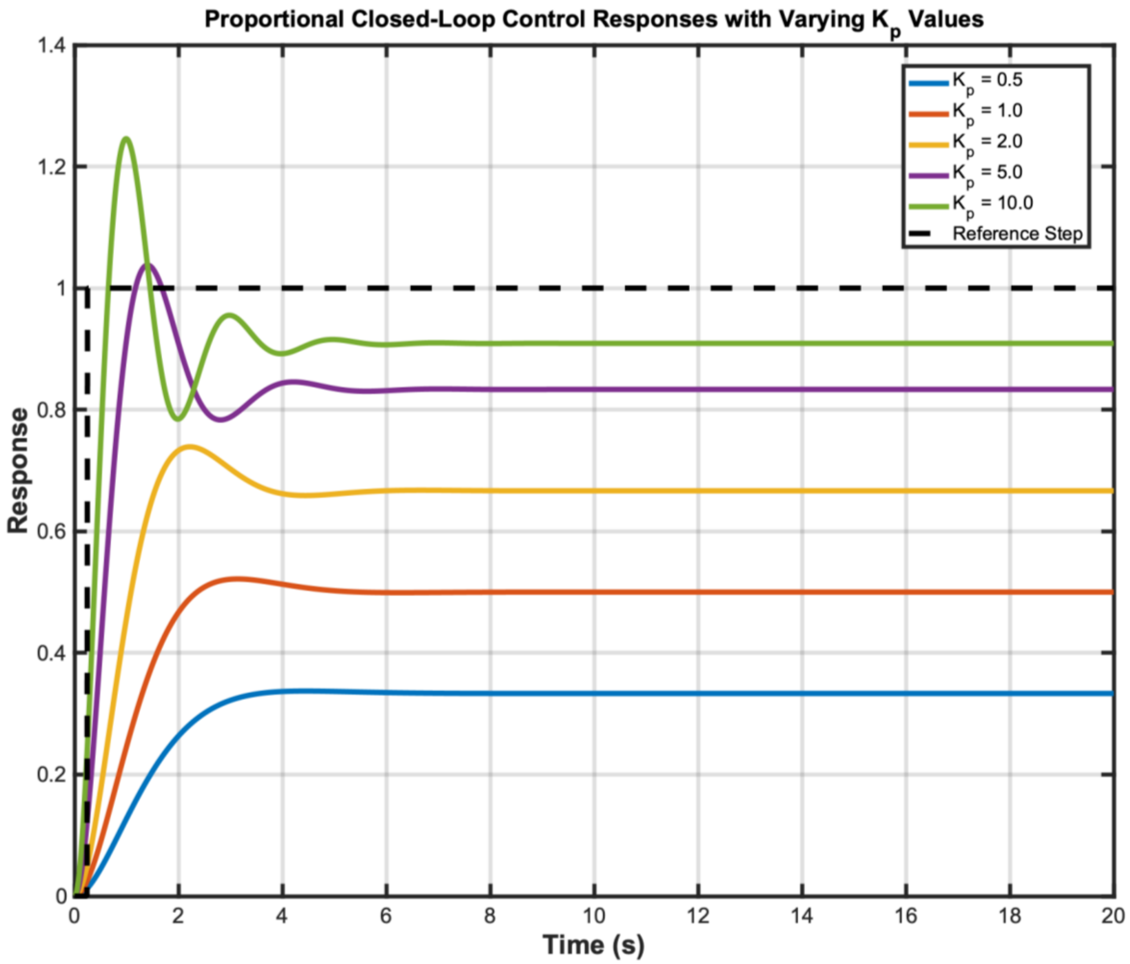
Hardware

Control Software

PID control overview

Where $e(t) = \textbf{reference signal} - \textbf{measured signal}$

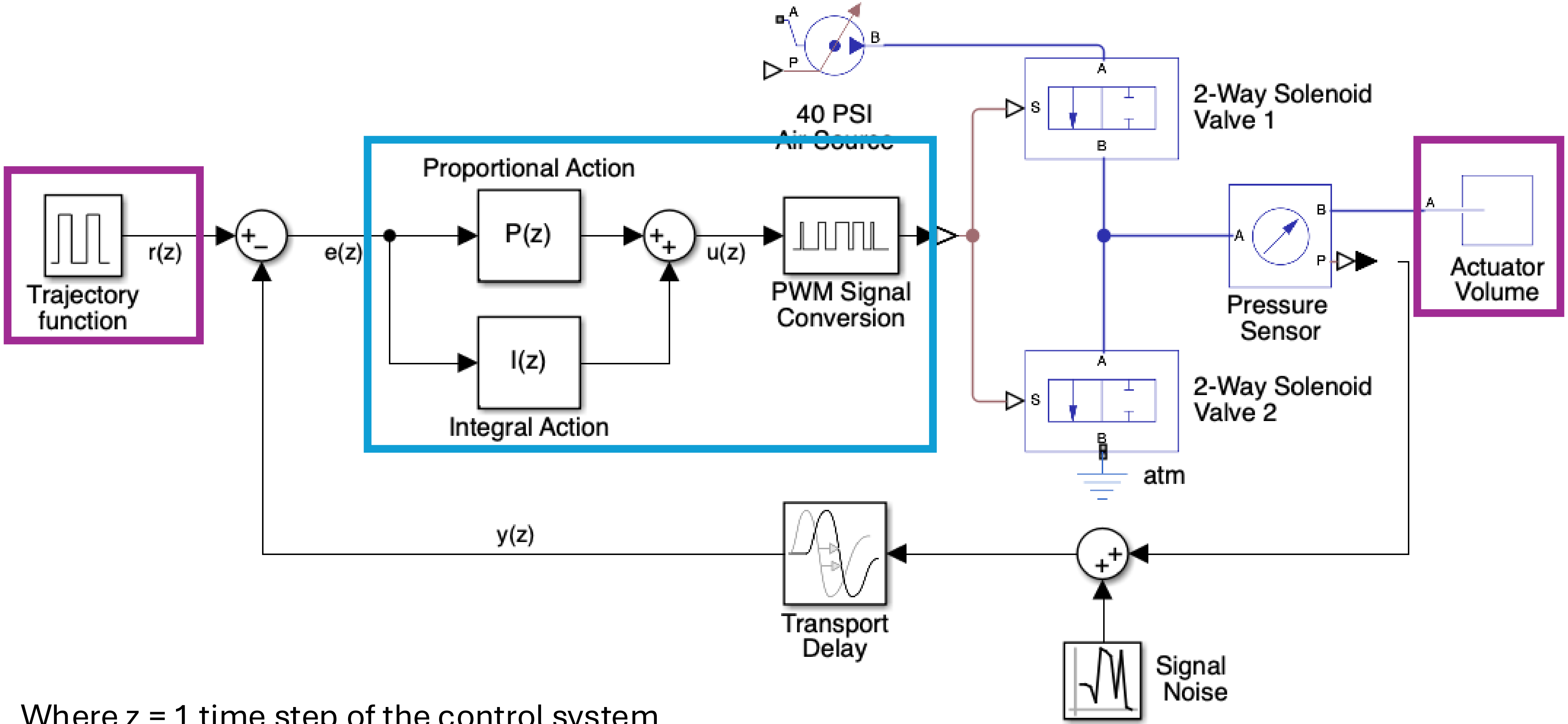




An engineer chooses the gain (K) values

Design criteria inform $\tau(t)$

This stand's pneumatic PID control system



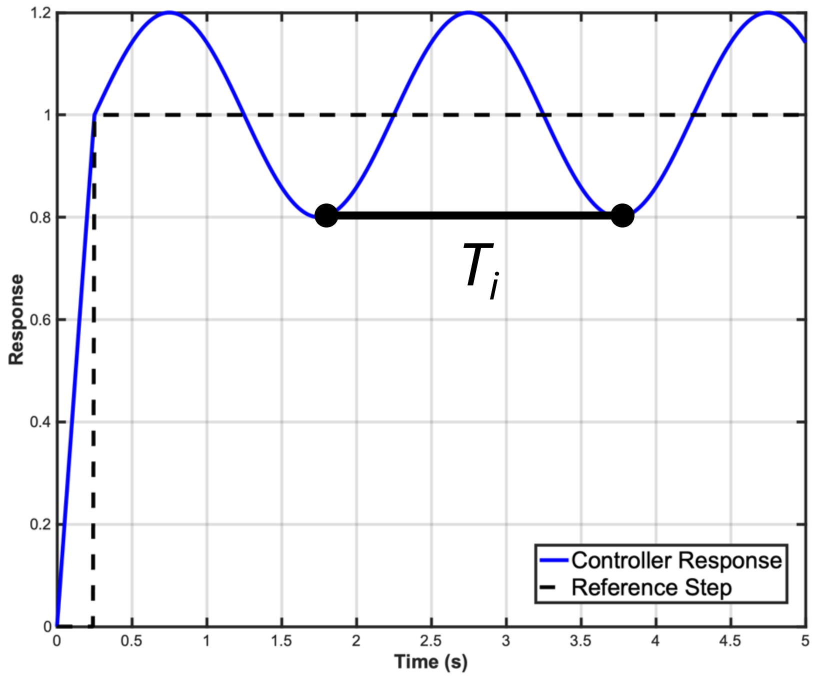
Where z = 1 time step of the control system

Ease of set-up

Zeigler-Nichols Tuning Process

1. Increment K_P until oscillations occur at a steady state
2. Set this K_p as K_u
3. Measure the steady state oscillation period, T_i

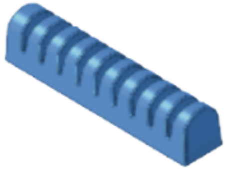
*Expanded, pictographic instructions to be included in the documentation



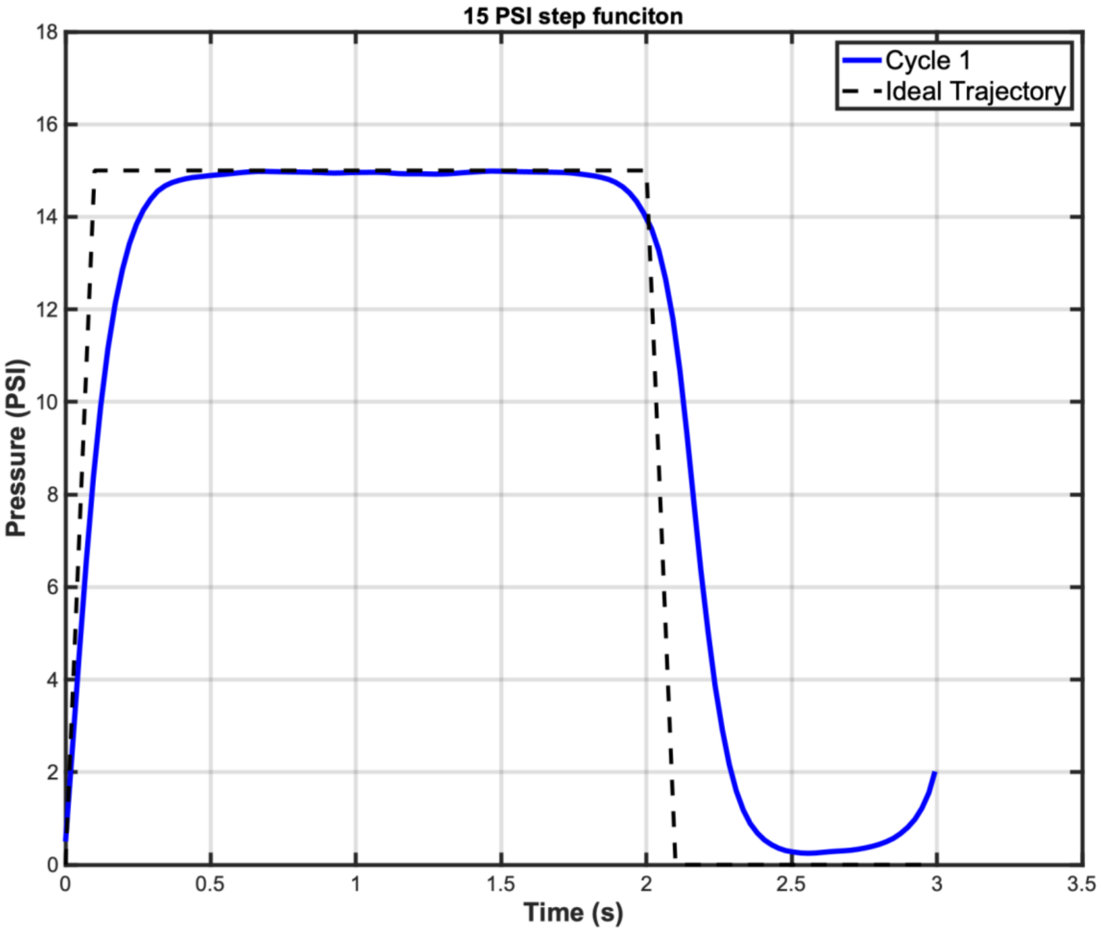
Ziegler–Nichols method[1]

Control Type	K_p	T_i	T_d	K_i	K_d
P	$0.5K_u$	–	–	–	–
PI	$0.45K_u$	$0.8\overline{3}T_u$	–	$0.54K_u/T_u$	–
PD	$0.8K_u$	–	$0.125T_u$	–	$0.10K_uT_u$
classic PID[2]	$0.6K_u$	$0.5T_u$	$0.125T_u$	$1.2K_u/T_u$	$0.075K_uT_u$
Pessen Integral Rule[2]	$0.7K_u$	$0.4T_u$	$0.15T_u$	$1.75K_u/T_u$	$0.105K_uT_u$
some overshoot[2]	$0.3\overline{3}K_u$	$0.50T_u$	$0.3\overline{3}T_u$	$0.6\overline{6}K_u/T_u$	$0.1\overline{1}K_uT_u$
no overshoot[2]	$0.20K_u$	$0.50T_u$	$0.3\overline{3}T_u$	$0.40K_u/T_u$	$0.06\overline{6}K_uT_u$

What data looks like

Candidate	Side Cross-Sectional View	Isometric View
Baseline		 [1]

```
// Step function trajectory
const float TIMES[] = {0, 100, 2000, 2100, 3000}; //milliseconds
const double PRESSURES[] = {0, 15, 15, 0, 0}; // PSI
```



Zeigler-Nichols & ideal gains

- This controller has the capacity to follow programmed trajectories with high accuracy
- Zeigler-Nichols tuning:

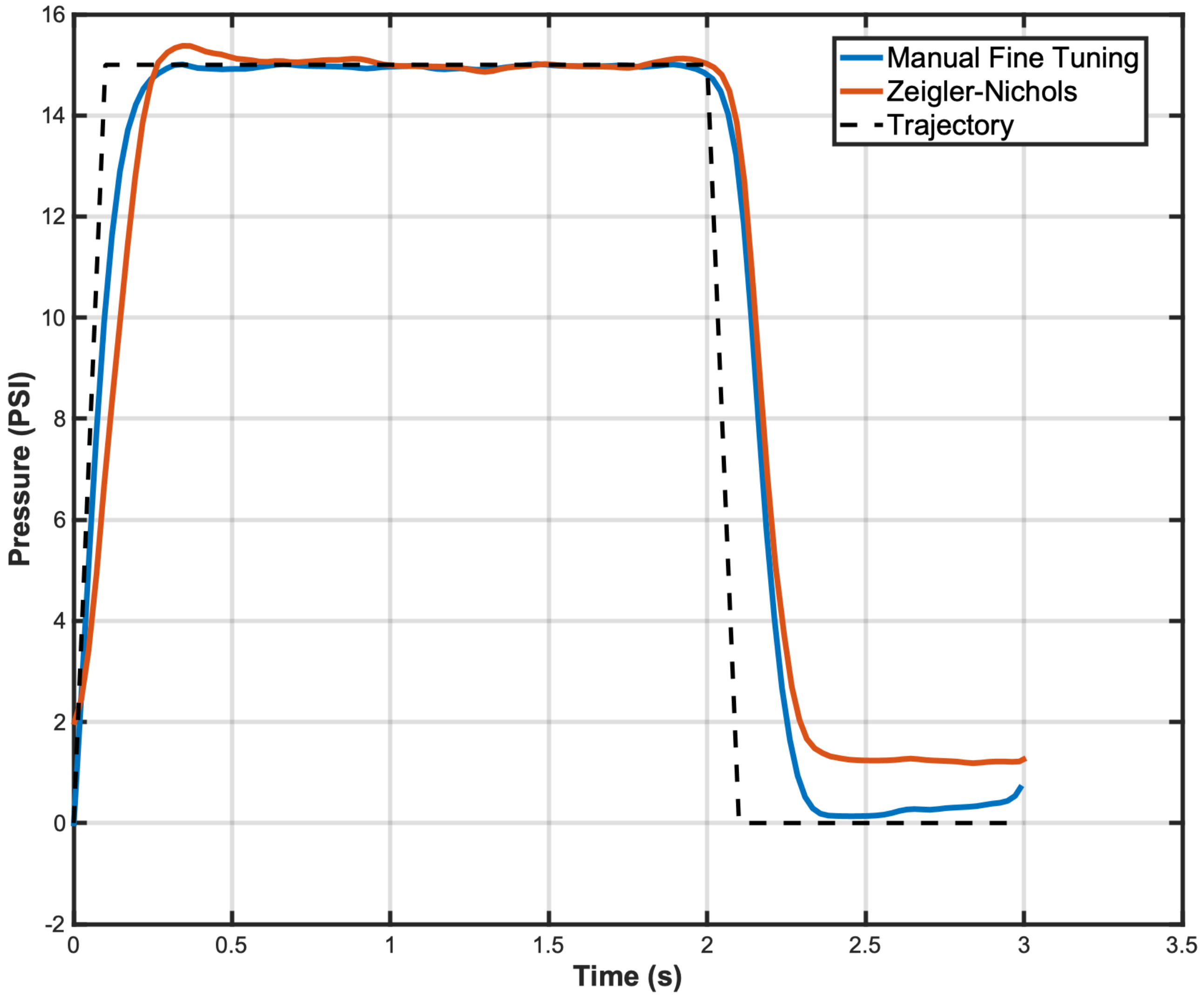
  $K_p = 0.675$

  $K_i = 0.00324$

- Fine manual tuning:

  $K_p = 0.4$

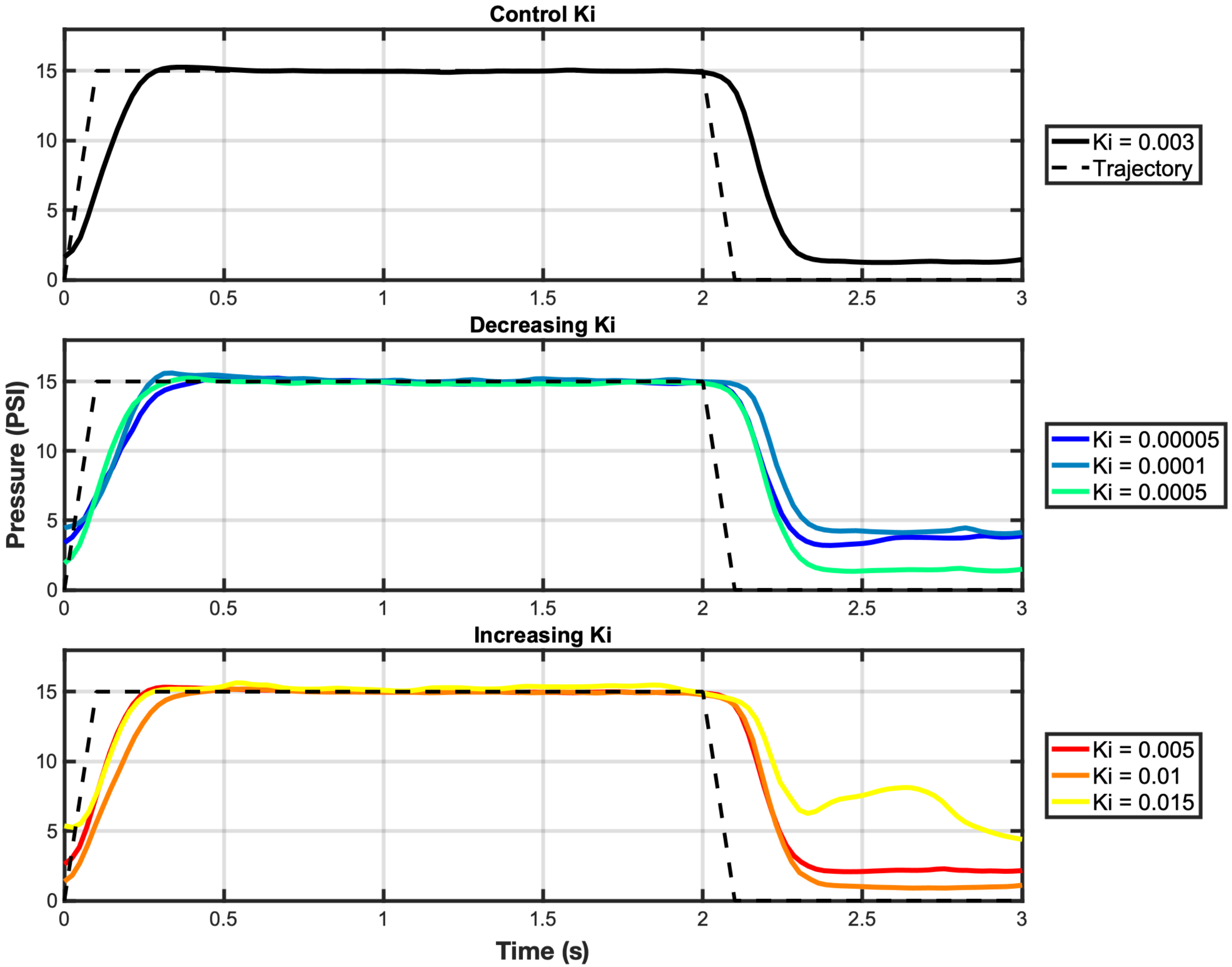
  $K_i = 0.00115$



Adjusting gains

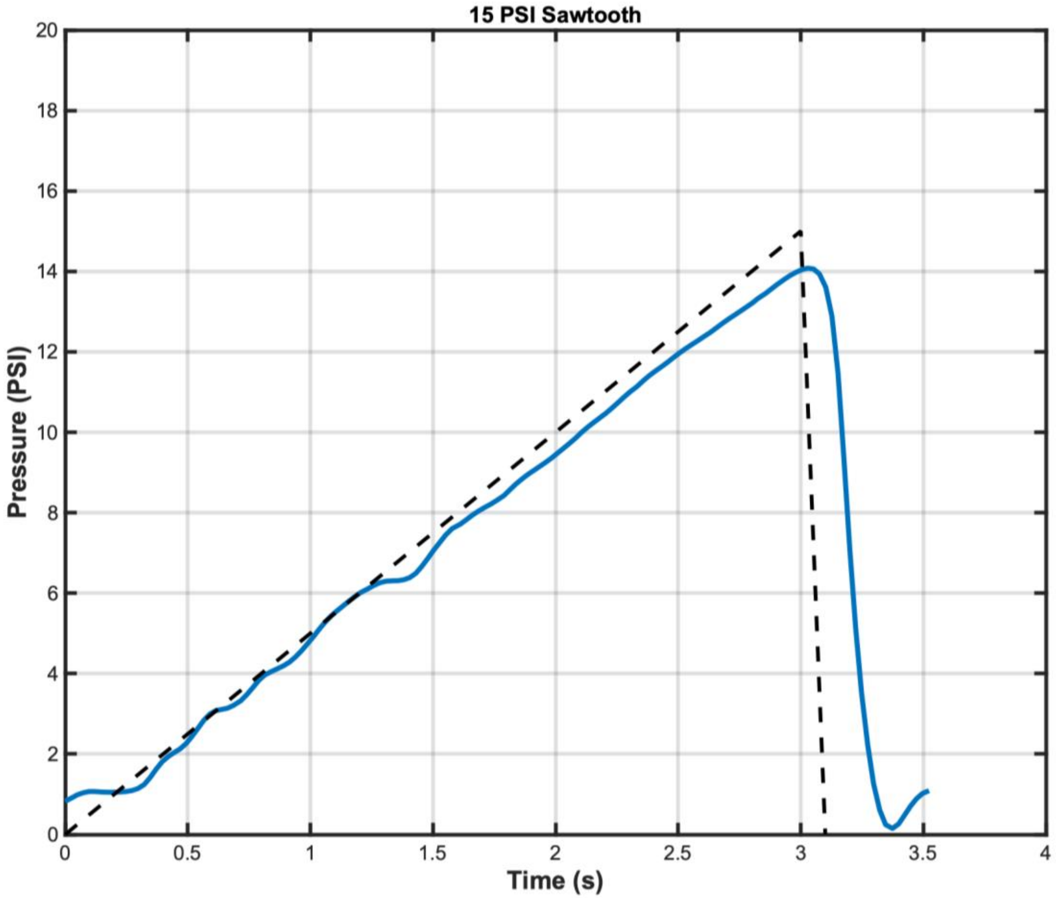
With high flow rates, system gains are small in magnitude:

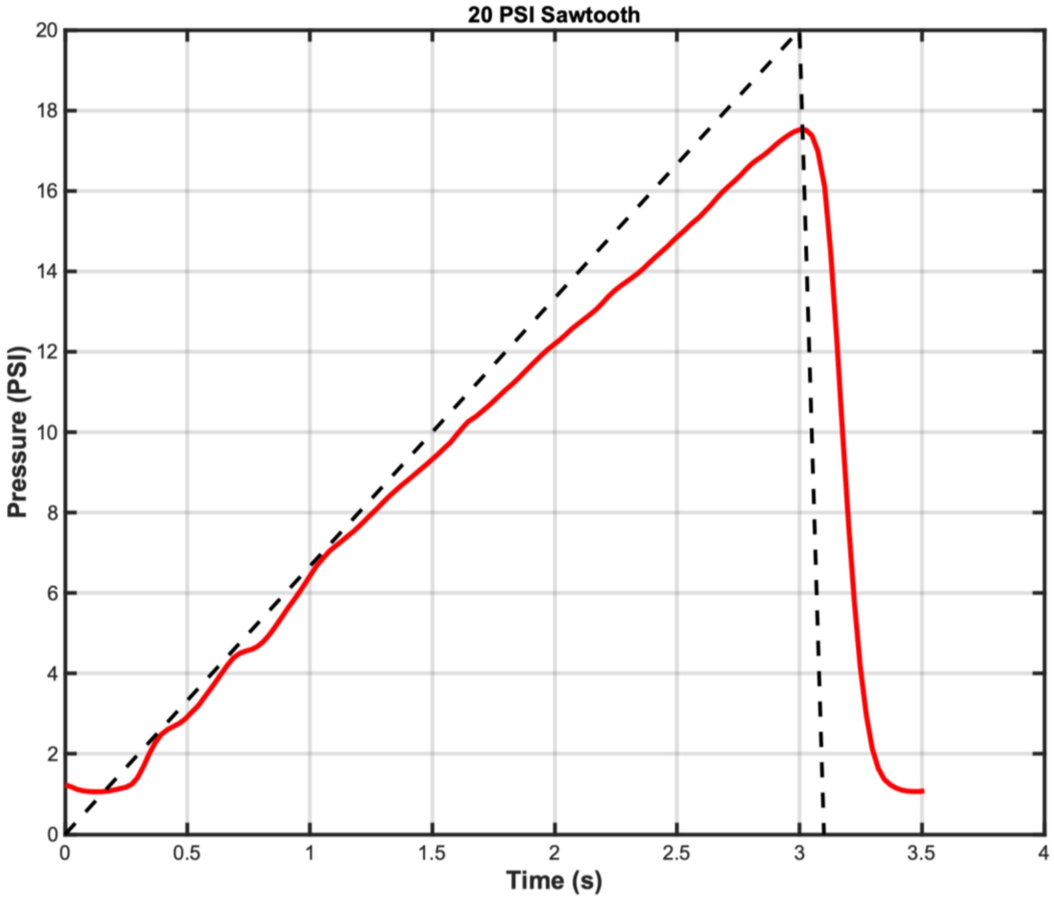
- K_p on the order of 10^{-1}
- K_i on the order of 10^{-3}
- K_d unnecessary given system response speed



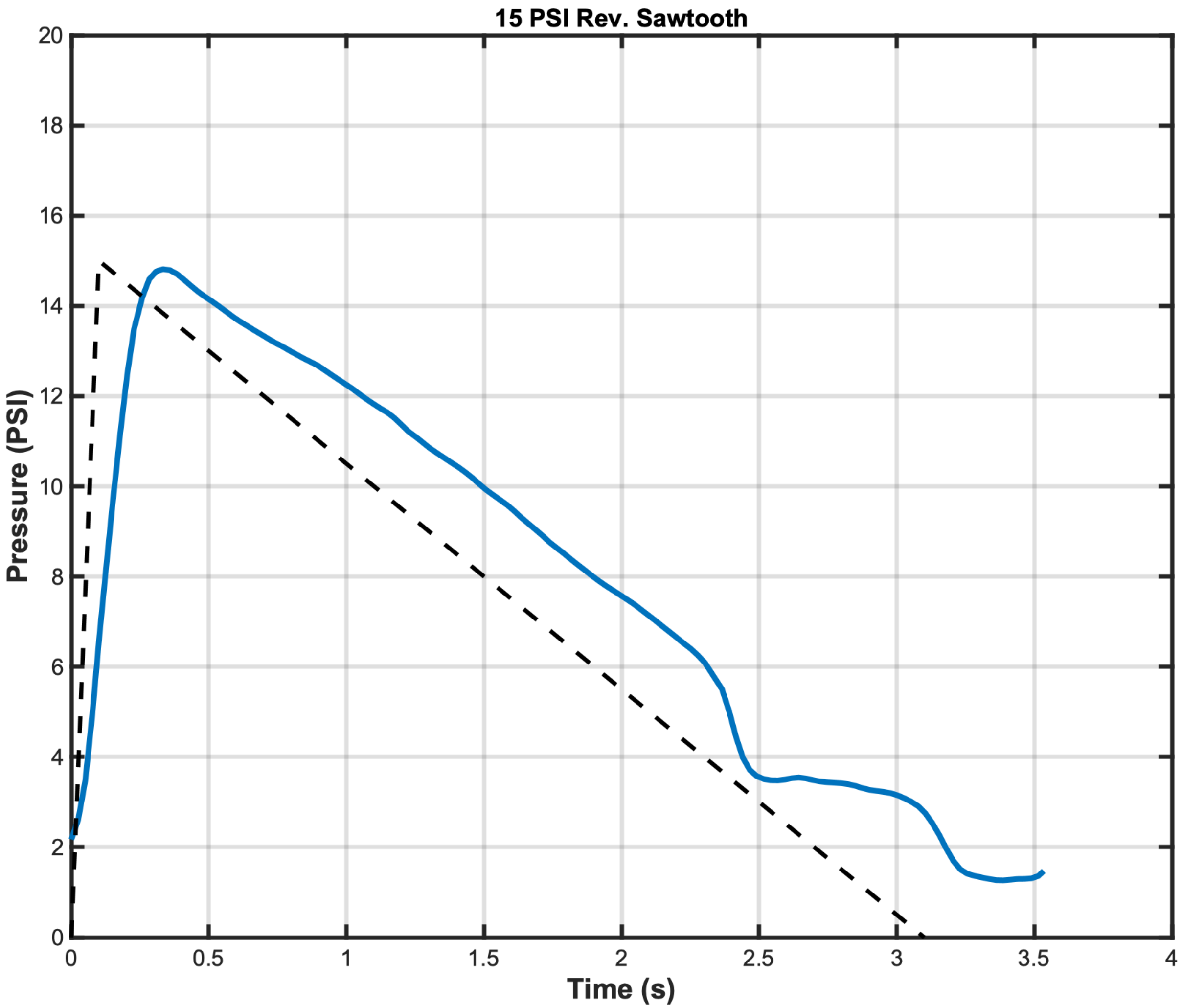
Versatile trajectory following

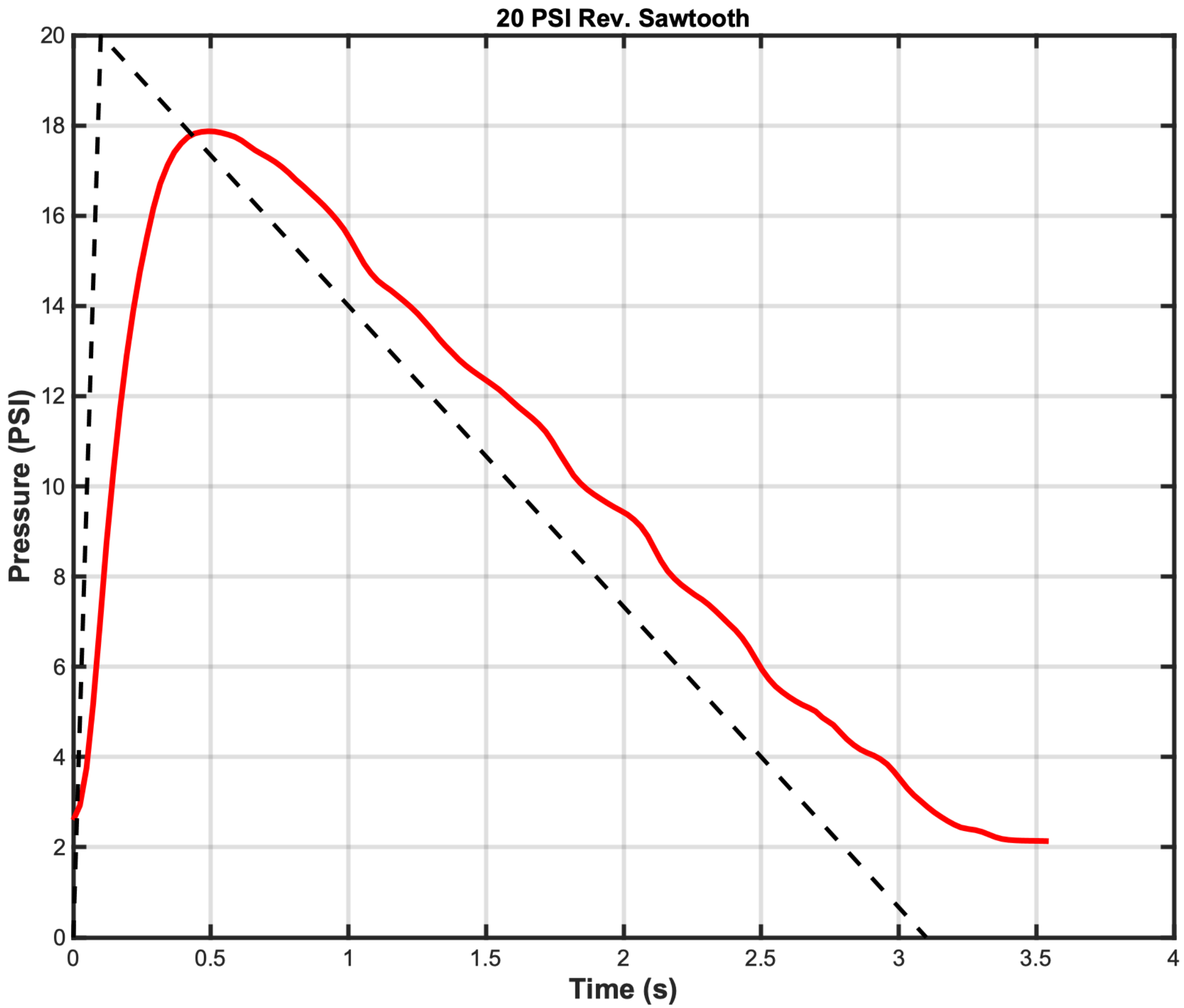
```
// Sawtooth Trajectory
const float TIMES[] = {0, 3000, 3100, 3500}; // milliseconds
const double PRESSURES[] = {0, 20, 0, 0};     // PSI
```



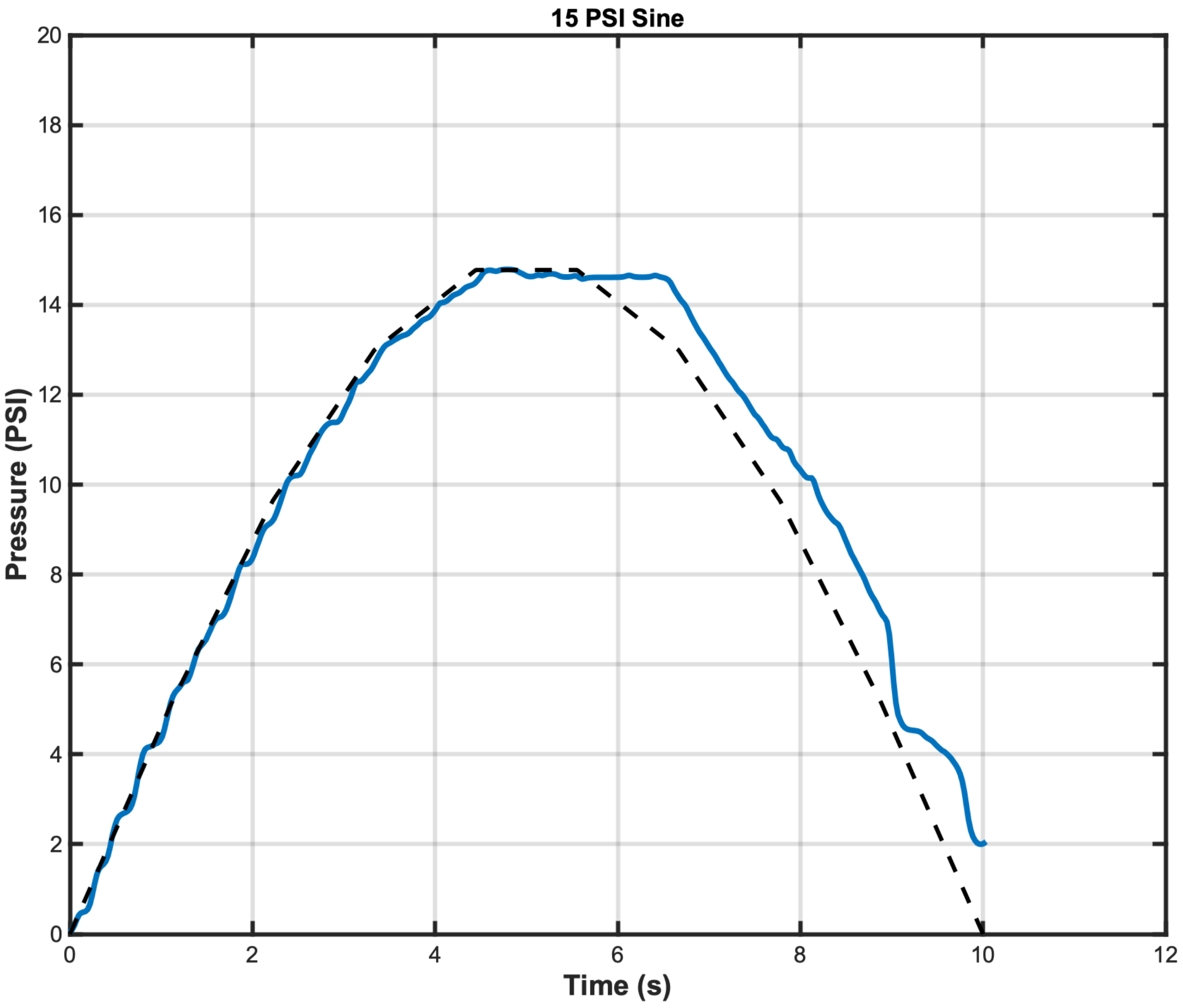


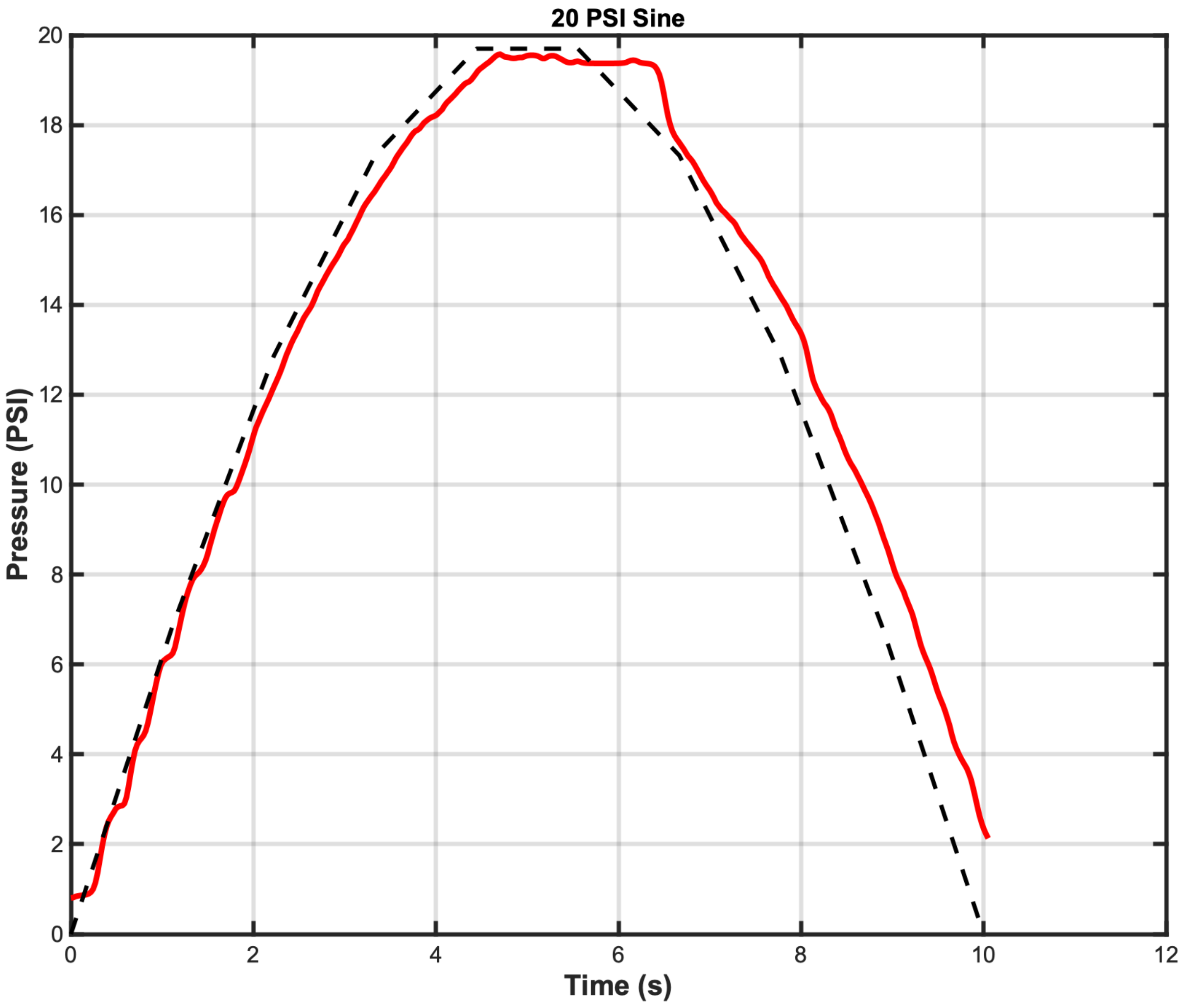
Versatile trajectory following cont.





Versatile trajectory following cont.





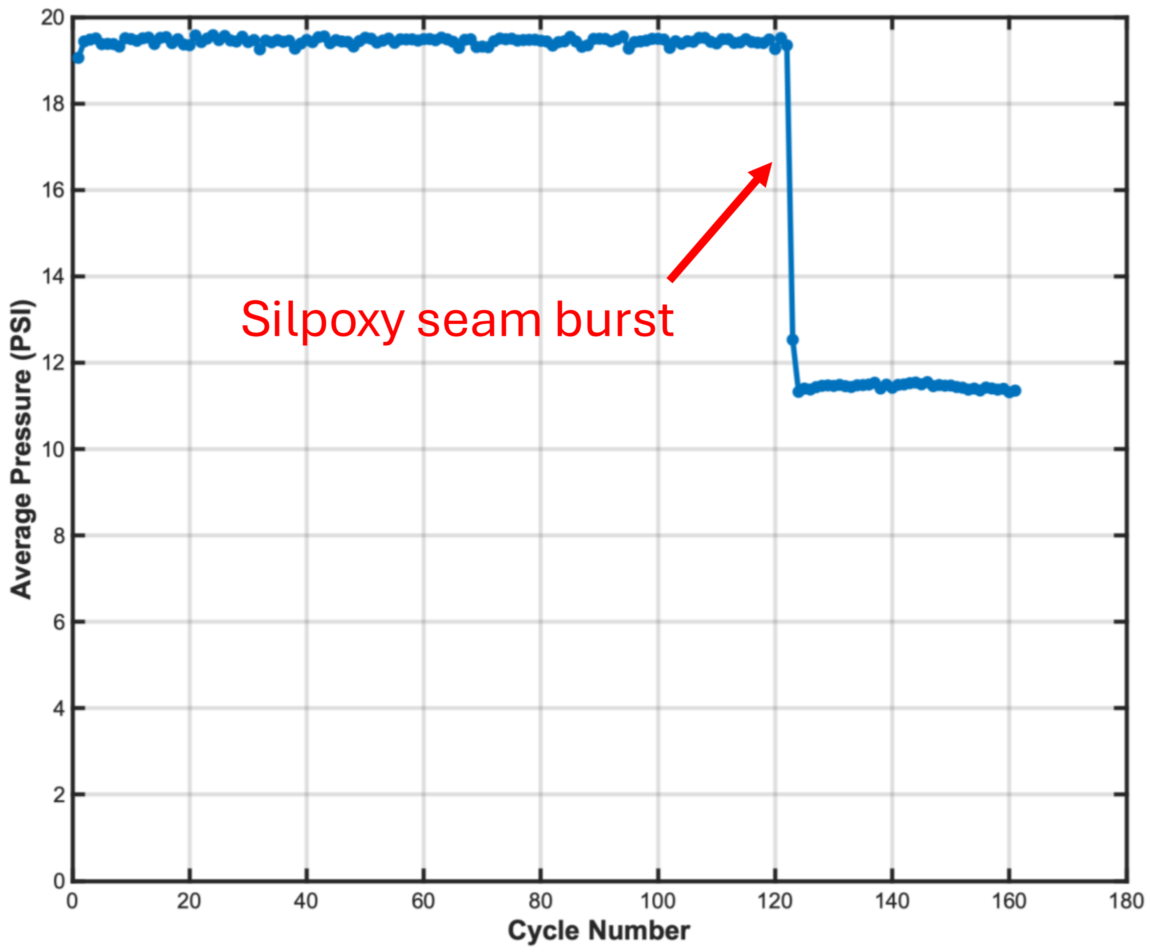
Cycle testing for actuator lifetime characterization

- Each dot on the graph represents the average pressure during the ”top” of each step function from one cycle

The “top” of the function

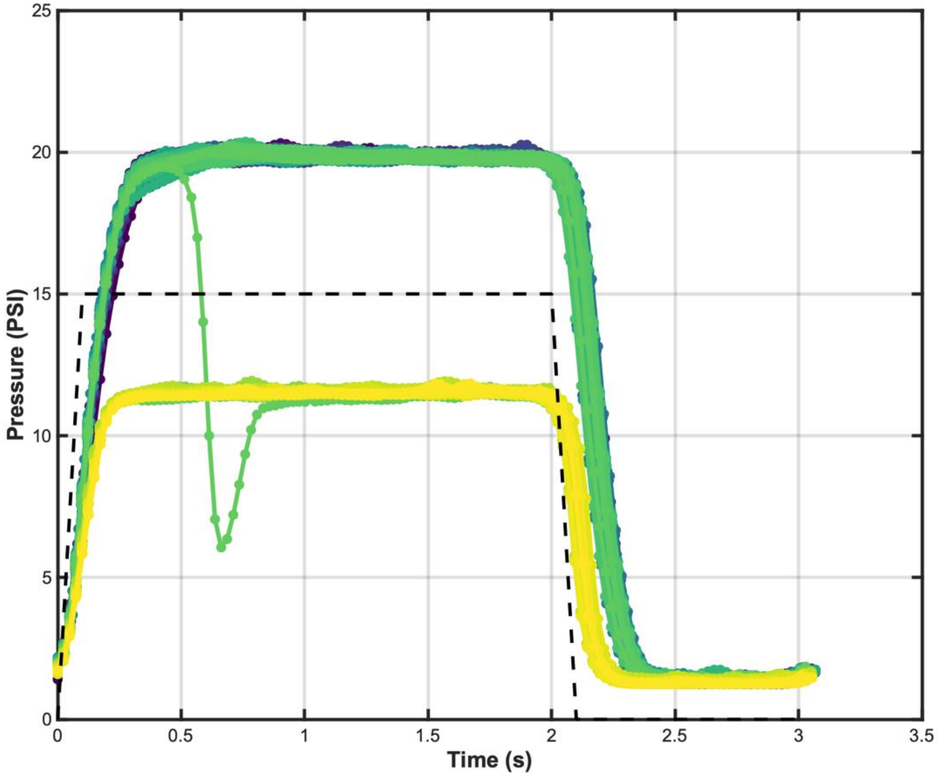
```
// Step function trajectory
const float TIMES[] = {0, 100, 2000, 2100, 3000}; //milliseconds
const double PRESSURES[] = {0, 15, 15, 0, 0}; // PSI
```

- Repeating trajectory cycle functions allows for **infinite test lengths**

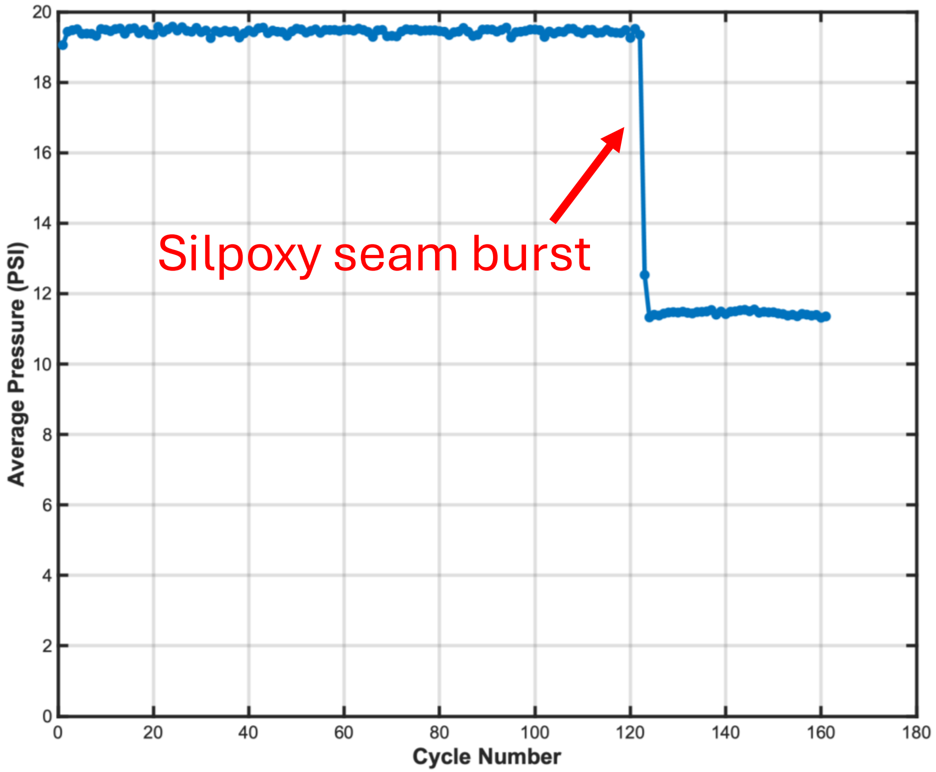


Cycle testing for actuator lifetime characterization

All Raw Data



Processed Version

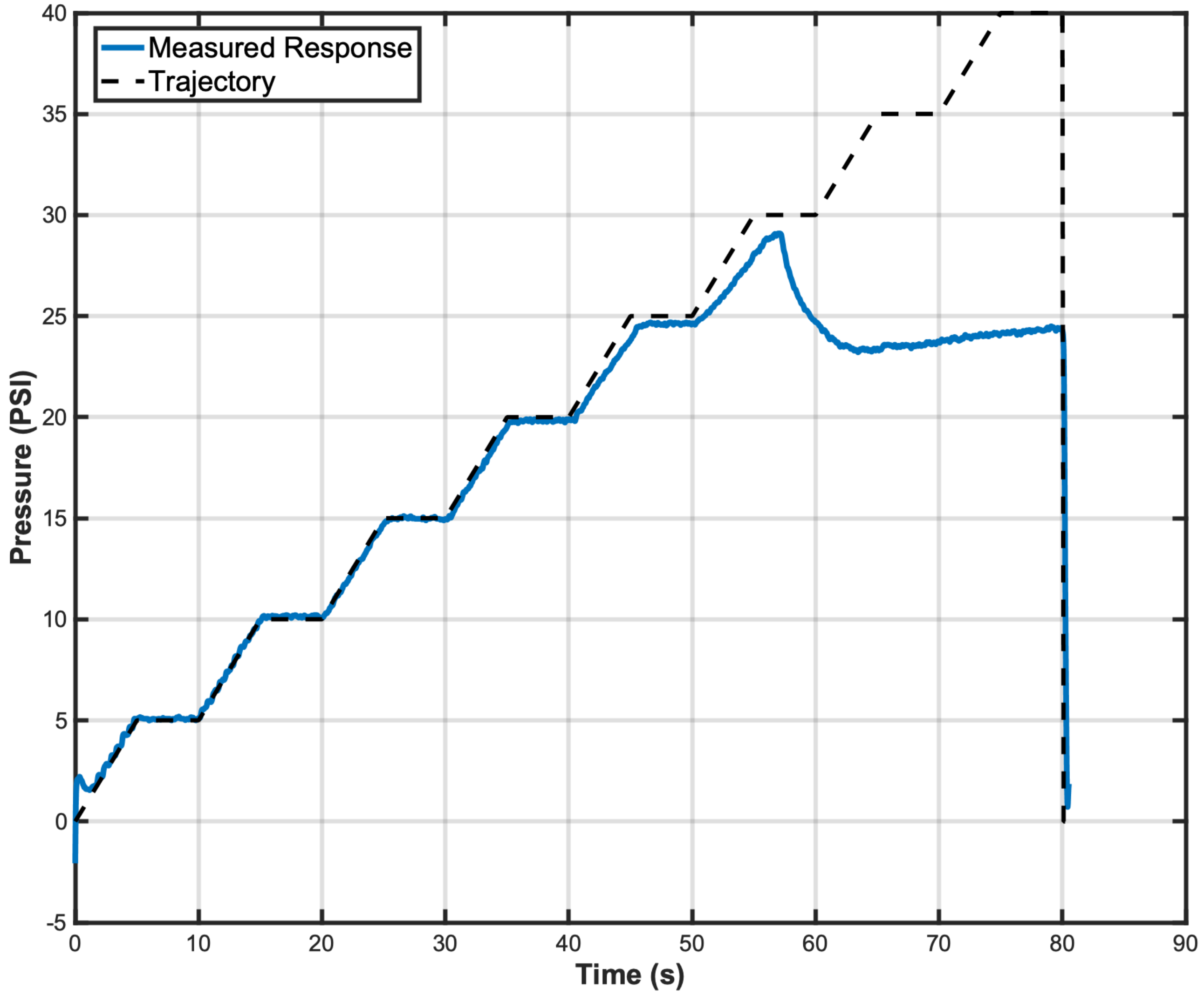


The "top" of the function

```
// Step function trajectory
const float TIMES[] = {0, 100, 2000, 2100, 3000}; //milliseconds
const double PRESSURES[] = {0, 15, 15, 0, 0}; // PSI
```

Burst testing for actuator limit characterization

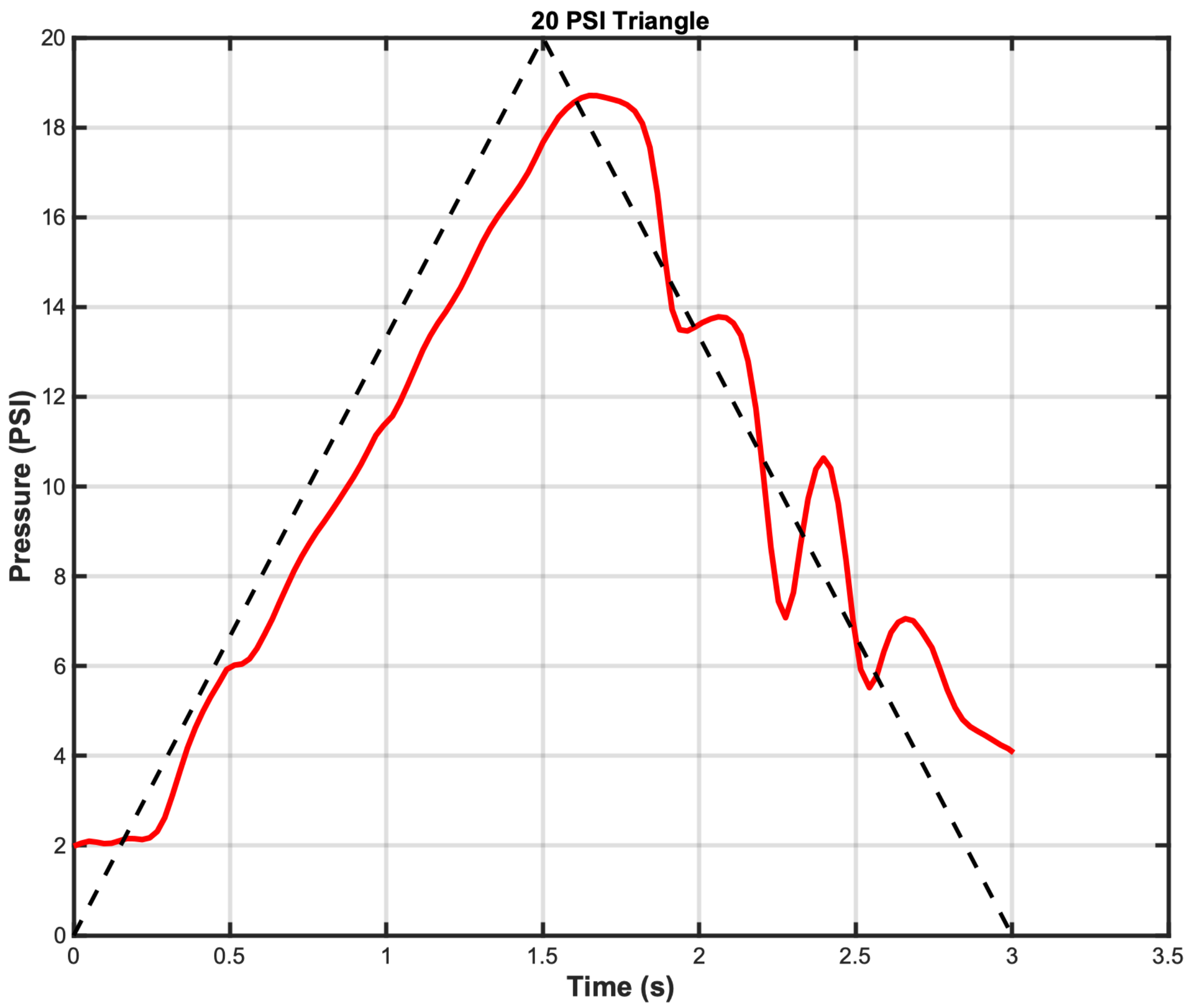
- Can also combine with video analysis to determine actuator's $\theta(P)$
- Safe to be around during this testing thanks to clear acrylic shielding



Controller limitations

Zeigler-Nichols tuning can be a time-intensive process

The controller struggles to follow negative slope trajectories near 10-13 PSI/sec. Currently unsure if that is actuator specific



References

[1] Doris et al. – 2024 – *Automated Computational Design of Soft Robots for Functionality and Durability*

[2] Puhlmann et al. – 2022 – *RBO Hand 3: A Platform for Soft Dexterous Manipulation*

[3] Preechayasomboon and Rombokas – 2020 – *Negshell casting: 3D-printed structured and sacrificial cores for soft robot fabrication*

[4] Hawkes et al. – 2021 – *Hard questions for soft robotics*

[5] Case and Marvel – 2024 – *FY24 Soft Robotics Report*

[6] Baines et al. – 2024 – *The need for reproducible research in soft robotics*