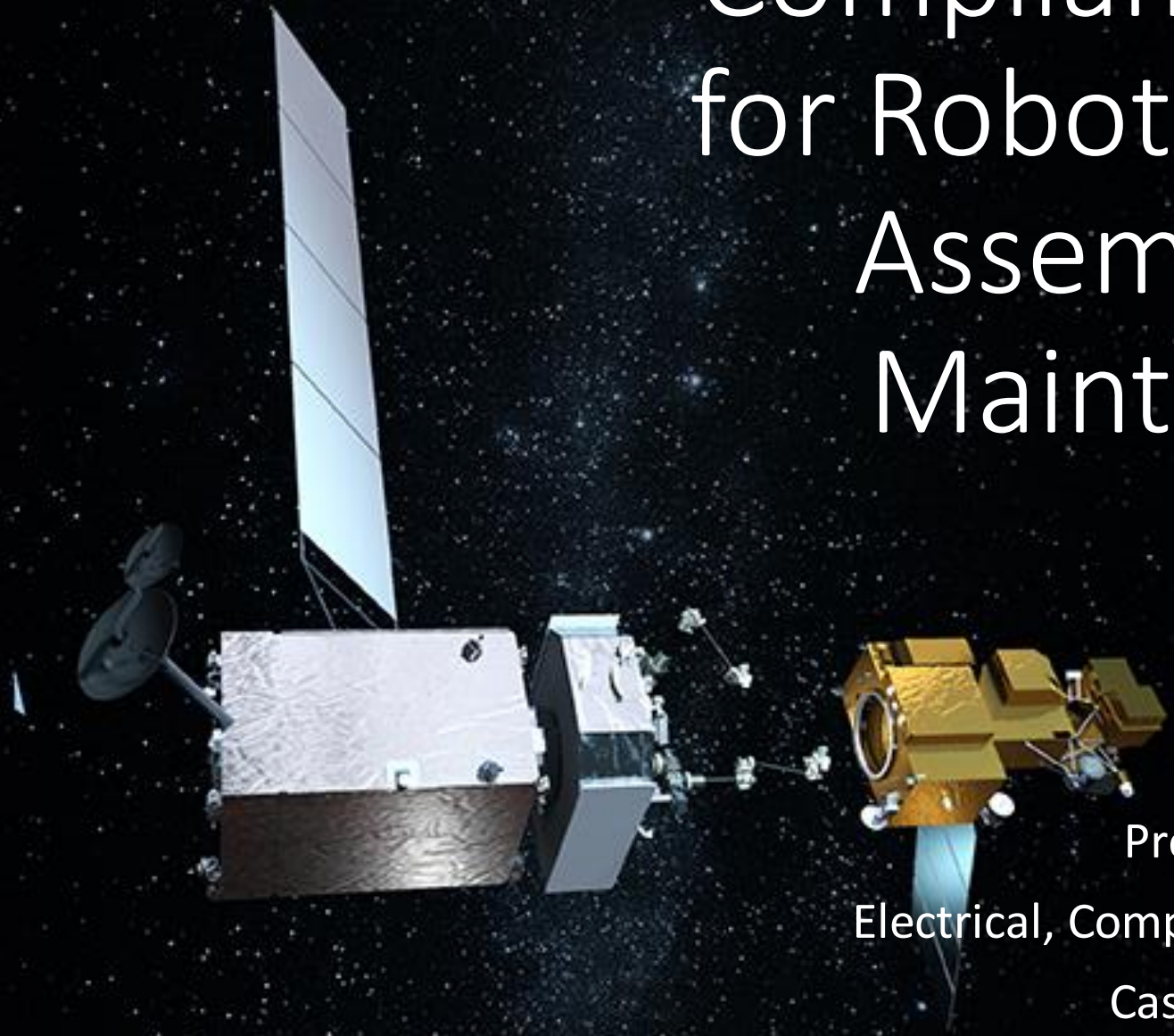
Compliance Control for Robotic In-Space Assembly and Maintenance



Professor Wyatt Newman
Electrical, Computer and Systems Engineering Dept
Case School of Engineering
Case Western Reserve University

Two parts:

- How to analyze force feedback to guarantee contact stability
- How to exploit force feedback for remote, supervisory control of contact operations

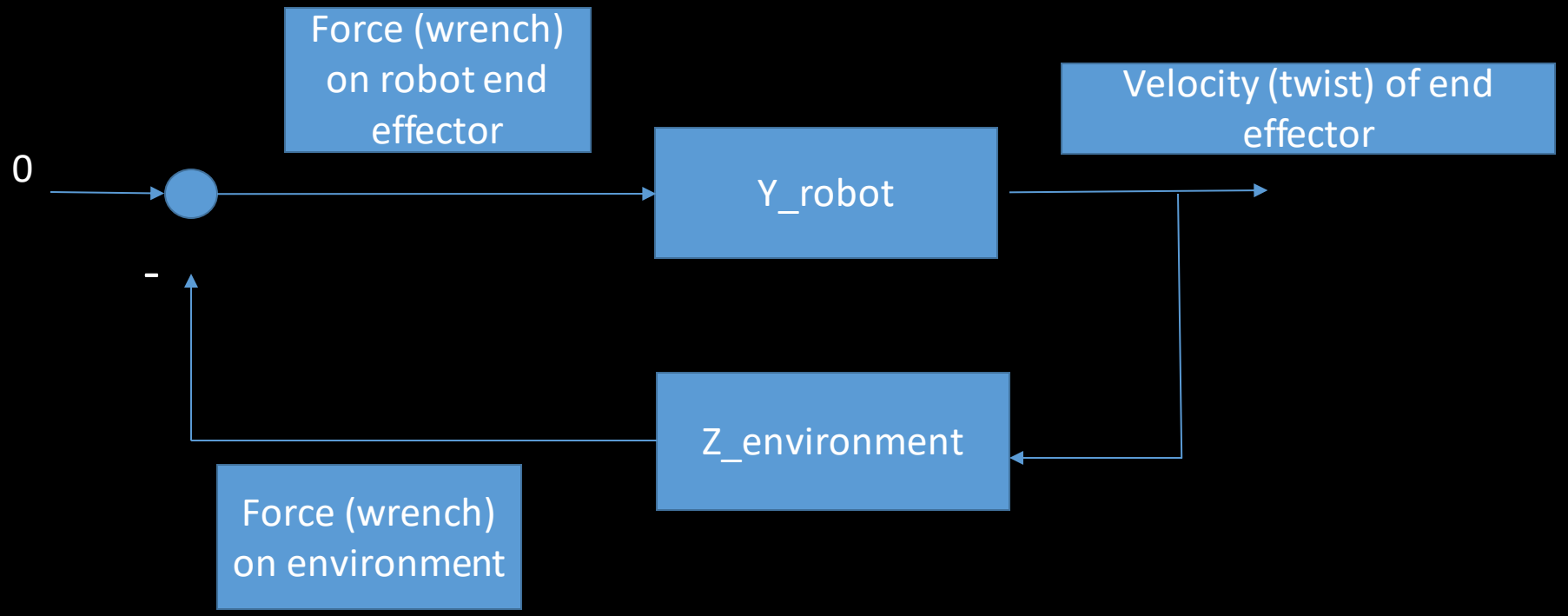
Compliant motion control

- Objective of compliant motion control (e.g. force control, impedance control, admittance control, accommodation control,…) is for the robot to be responsive to forces/moments exerted by the environment on the robot's end effector.
- An essential requirement is that the robot remain stable in contact operations.
- Ideally, contact stability would be guaranteed for all potential environments of interest.
- ***Passivity*** can be used to analyze stability in contact with a very broad class of environments

Some sources of instability in force feedback include:

- Non-collocation of sensing and actuation
- Mechanical resonances
- Controller latency and update rate
- Actuator, sensor and controller bandwidth
- Workpiece/environment dynamics

Conceptual admittance/impedance relationship



Can interpret robot end-effector admittance and environment impedance as a feedback system, subject to analysis techniques of control systems.

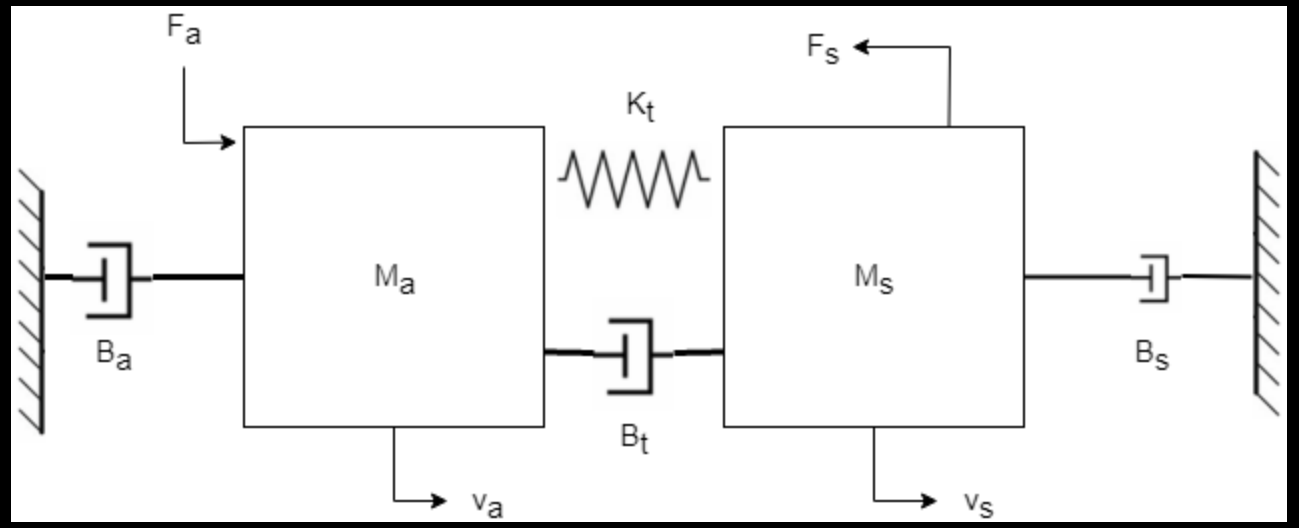
Passivity analysis approach:

- Given a robot under compliant-motion control (e.g. force feedback), express the resulting dynamics as an admittance with respect to the end effector.
- Assume low speed contact (as appropriate for contact operations) and ***linearize*** the dynamics about a proposed operating point (robot pose).
- Express the endpoint dynamics as an admittance matrix, comprised of linear transfer functions in “s”
- The endpoint admittance, Y_robot, is a 6-DOF relationship between endpoint wrench and endpoint twist.
- Evaluate if the resulting admittance matrix is “passive”
- If the endpoint admittance is passive, then the robot will be stable in contact with all passive environments, linear or nonlinear, of arbitrary complexity.

Robot modeling:

Since actuator dynamics is known to contribute to instability, include transmission compliance in model of each robot joint.

Rectilinear equivalent model for a single joint is:



Extension to N-DOF linearized robot dynamics (transmission dynamics doubles the order of the differential equations)

$$\mathbf{Y}_{sys}(s) = \mathbf{Z}_{sys}^{-1}(s) = \left(\mathbf{H}_{sys}s + \mathbf{B}_{sys} + \frac{1}{s}\mathbf{K}_{sys}\right)^{-1}$$

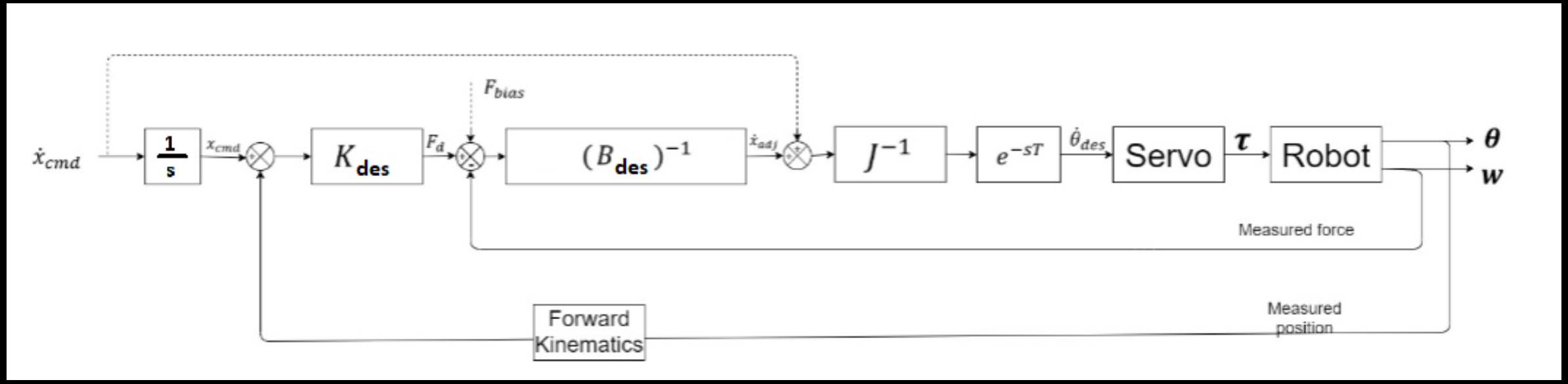
$$\mathbf{H}_{sys} = \begin{bmatrix} \mathbf{H}_{links}(\mathbf{q}_{links}) & 0 \\ 0 & \mathbf{H}_{act} \end{bmatrix}$$

$$\mathbf{B}_{sys} = \begin{bmatrix} \mathbf{B}_{links} & 0 \\ 0 & \mathbf{B}_{act} \end{bmatrix}$$

$$\mathbf{K}_{sys} = \begin{bmatrix} \mathbf{K}_{trans} & -\mathbf{K}_{trans} \\ -\mathbf{K}_{trans} & \mathbf{K}_{trans} \end{bmatrix}$$

Example controller design

- Robot joints controlled individually by linear, PID controllers
- User can control the robot with relatively high-speed incremental position commands
- User retrofits a force/torque sensor and derives incremental position commands to achieve some target endpoint admittance (e.g. desired stiffness and damping)



Closed-loop endpoint admittance:

- The space of linear force-feedback options can be described in terms of two transfer functions:
- Gv(s) represents the contribution to joint velocity commands in response to sensed joint positions and velocities
- Gf(s) represents the contribution to joint velocity commands in response to force/torque sensing at the endpoint.
- Regardless of the controller design choices, these contribute to the closed-loop endpoint admittance as:

$$\mathbf{Y}_{s,cl} = \begin{bmatrix} \mathbf{J} & \mathbf{0} \end{bmatrix} \left(\mathbf{Z}_{sys} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_v(s) \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{J}^T \\ \mathbf{G}_f(s) J^T \end{bmatrix}$$

Example:

- Inner loop is comprised of PD joint controllers
- Outer loop commands joint positions to emulate endpoint behavior of a spring/damper system
- Corresponding Gf and Gv transfer functions (matrices) can be expressed (after some algebra) as:

$$\mathbf{G}_f(s) = \frac{e^{-sT}}{s}\mathbf{K}_p\mathbf{J}^{-1}\mathbf{B}_{des}^{-1}\mathbf{J}^{-T}$$

$$\mathbf{G}_v(s) = \frac{e^{-sT}}{s^2}\mathbf{K}_p\mathbf{J}^{-1}\mathbf{B}_{des}^{-1}\mathbf{K}_{des}\mathbf{J} + \frac{1}{s}\mathbf{K}_p + \mathbf{K}_v$$

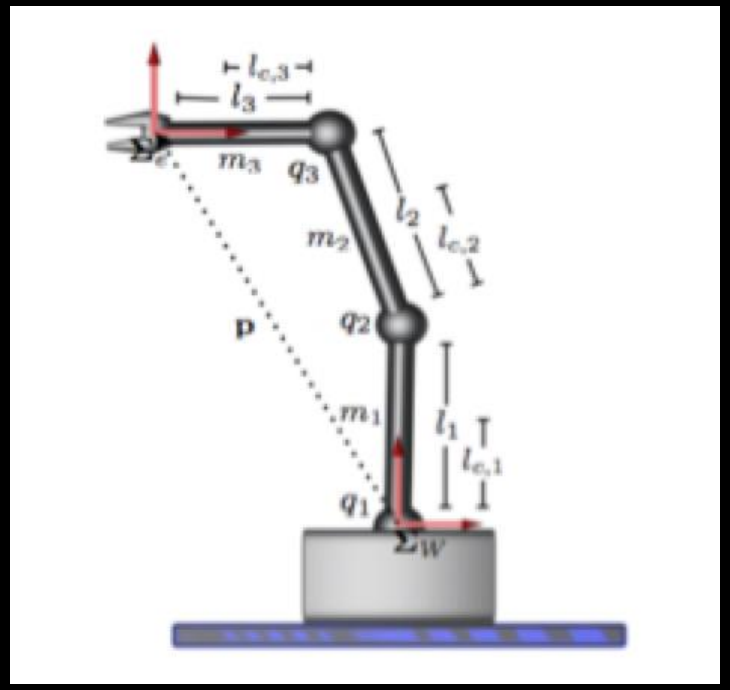
Example: 3DOF planar robot

Jacobian:

$$J = \begin{bmatrix} 0 & 0.71 & 0 \\ 1.71 & 1.01 & 0.30 \\ 1.0 & 1.0 & 1.0 \end{bmatrix} m$$

Matrix Inertia Tensor

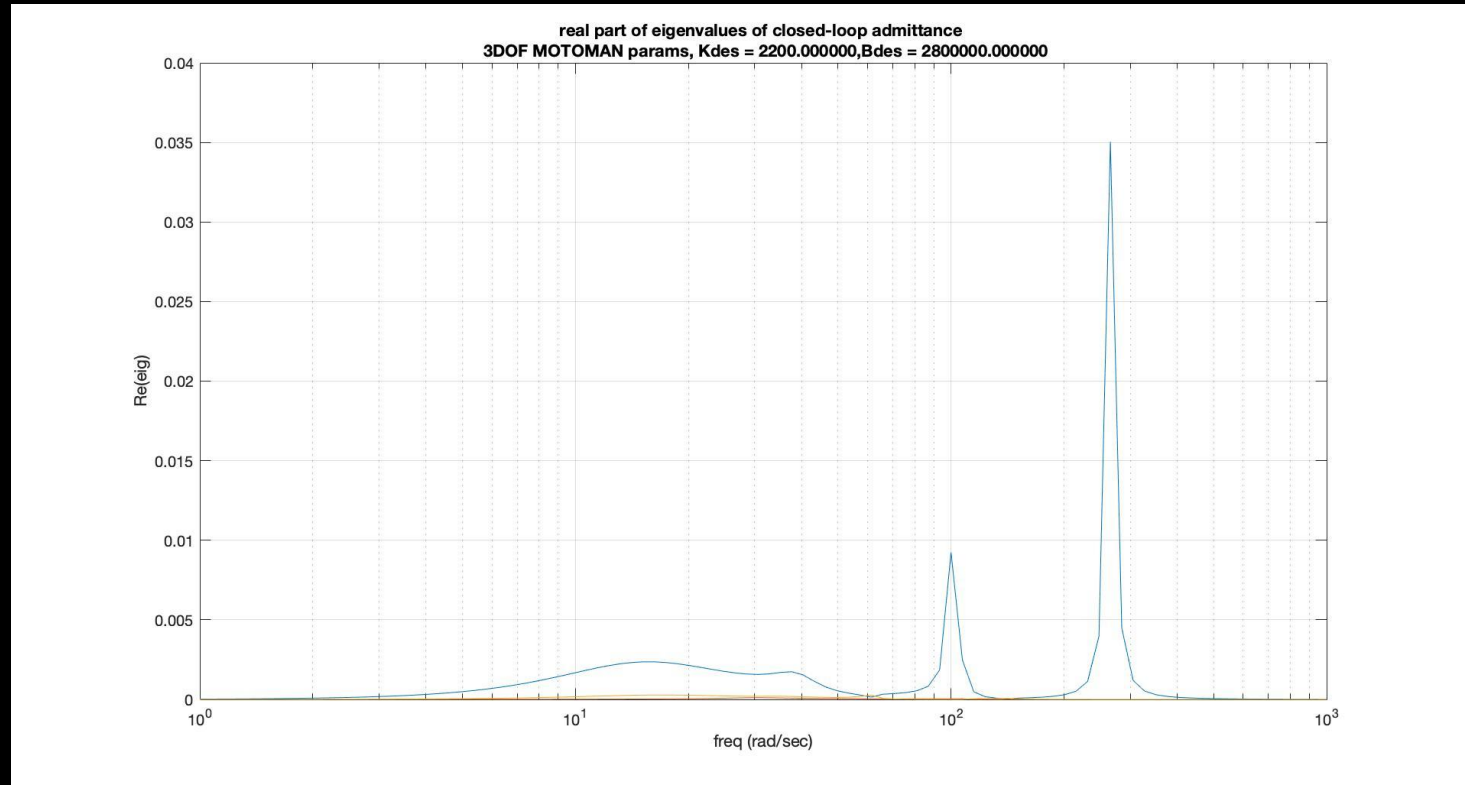
$$H = \begin{bmatrix} 239.9 & 30.5 & 6 & 0 & 0 & 0 \\ 30.5 & 28.1 & 3.6 & 0 & 0 & 0 \\ 6 & 3.6 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 197.4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 96.6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 33.8 \end{bmatrix} kg \cdot m^2$$



Example 3-DOF model analysis

- Compute Y(s) as a 3x3 matrix of transfer functions
- Choose frequencies at which to evaluate
- Substitute s=jω at chosen frequencies to convert Y(s) into an array of complex numbers
- Compute the eigenvalues of the resulting matrix
- Evaluate if any eigenvalue has a negative real part→ non-passive

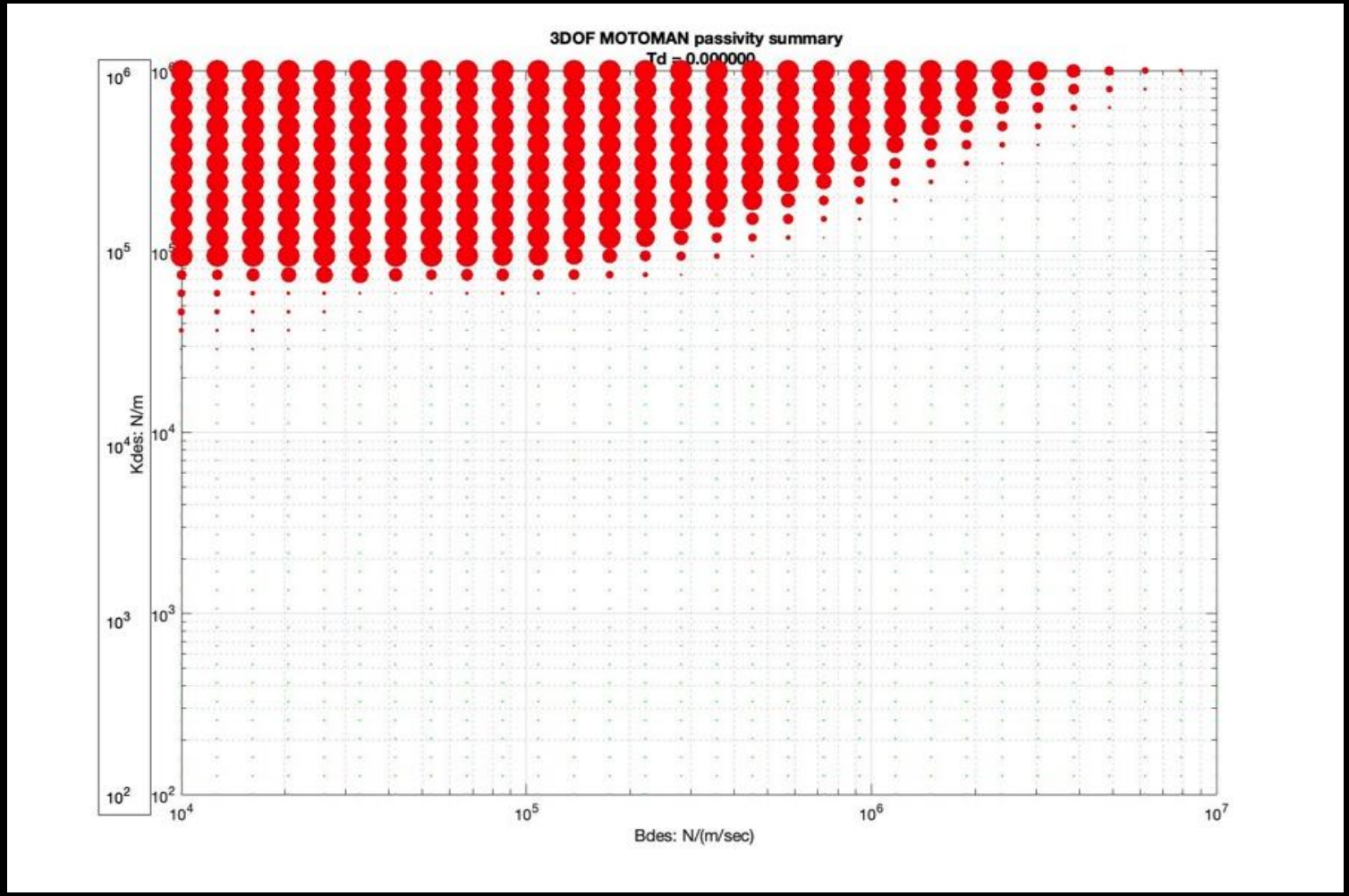
	Shoulder Link	Elbow Link	Wrist Link
Transmission stiffness $(N-m-rad^{-1})$	400000	200000	100000
Transmission damping $(N-m-s-rad^{-1})$	200	200	200
Actuator damping $(N-m-s-rad^{-1})$	1500	600	300
Link damping $(N-m-s-rad^{-1})$	20	20	20
Bandwidth (Hz)	5.5	5.5	5.5
Damping ratio: ζ	1.0	1.0	1.0
K_p	520000	36000	7200
K_v	30000	2100	400
K_I	7500	500	100
Joint Angle (rad)	$\frac{\pi}{4}$	$-\frac{\pi}{2}$	$\frac{\pi}{4}$



Parameter search:

Given a robot model:

- Choose pose
- Set inner and outer-loop (e.g. Kdes, Bdes) gains
- Sweep across frequencies to compute eigenvalues
- Map out passive/non-passive results as a function of control parameters



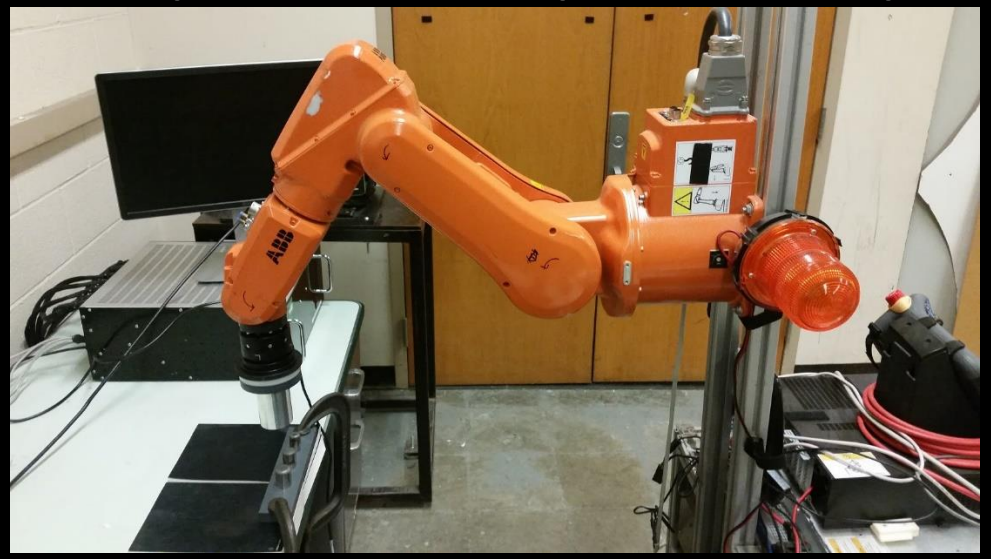
Analysis technique is:

- Extensible to higher dimensions, including redundant kinematics
- Applicable to arbitrary feedback designs
- Capable of yielding a universal guarantee of stability with respect to passive environments

Part 2: Supervisory Control of Compliantly-Controlled Robots

- Programmable compliance is desired to perform contact operations, such as mechanical assembly
- Communication latency for remote operations makes haptic feedback impractical
- Responsiveness to contact efforts should be delegated to the robot's local controller
- Higher-level, supervisory commands may be sent by a remote operator
- One can interface with a compliant-motion controller using trajectories of virtual (soft) attractors

Abstracted tasks: sleeve on peg and childproof cap manipulation







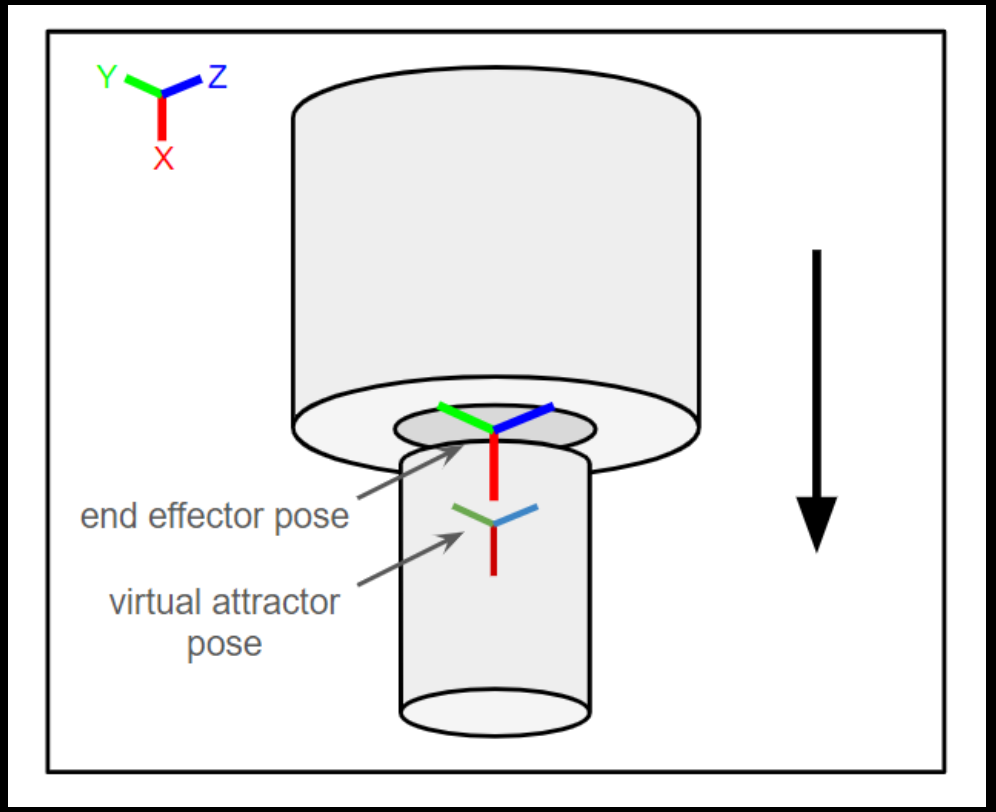


The Accommodation Controller

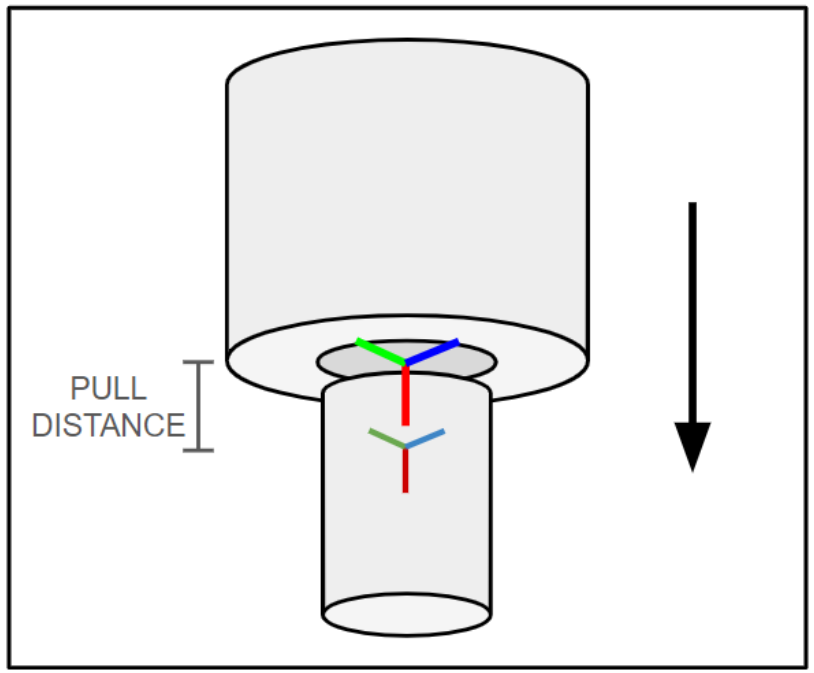
$$F_{virtual} = K_{translational} * (P_{virtual\ attractor} - P_{end\ effector}) \quad (1)$$

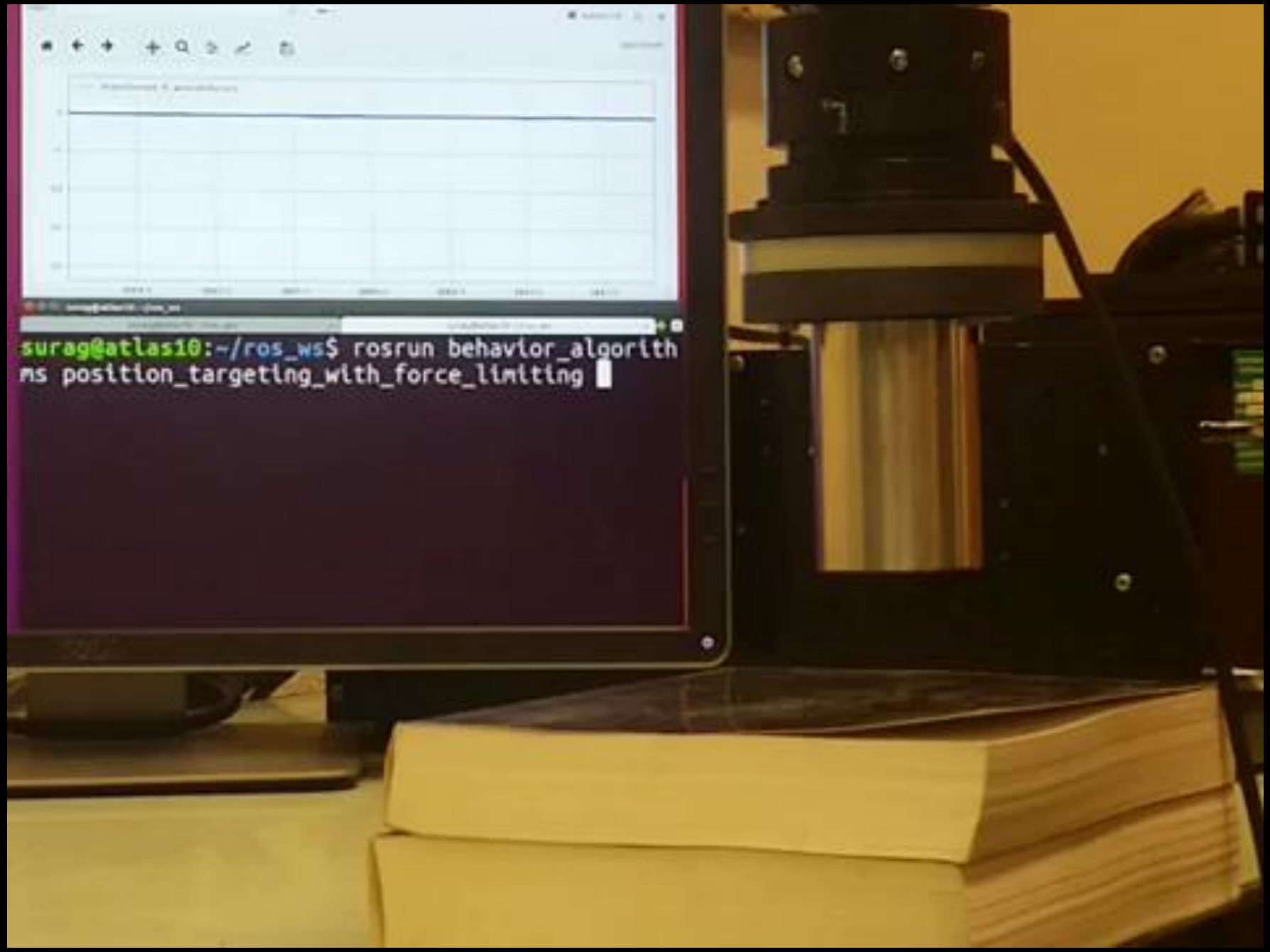
$$To_{virtual} = K_{angular} * (R_{virtual\ attractor} - R_{end\ effector}) \quad (2)$$

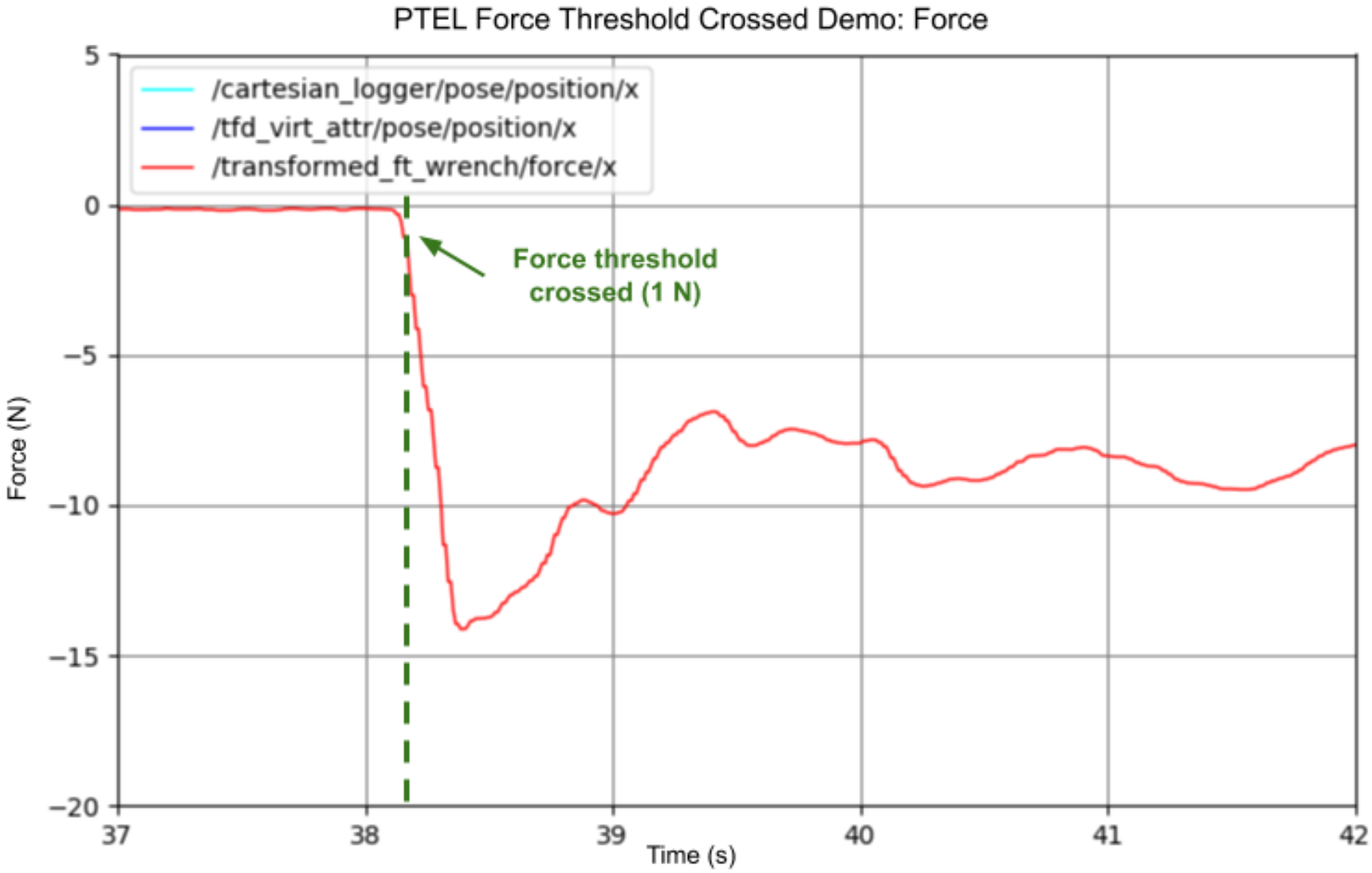
$$T_{Command} = \frac{T_{previous} * B + W_{physical} + W_{virtual}}{I} * \Delta t \quad (3)$$



Skills – Position Target, Effort Limiting



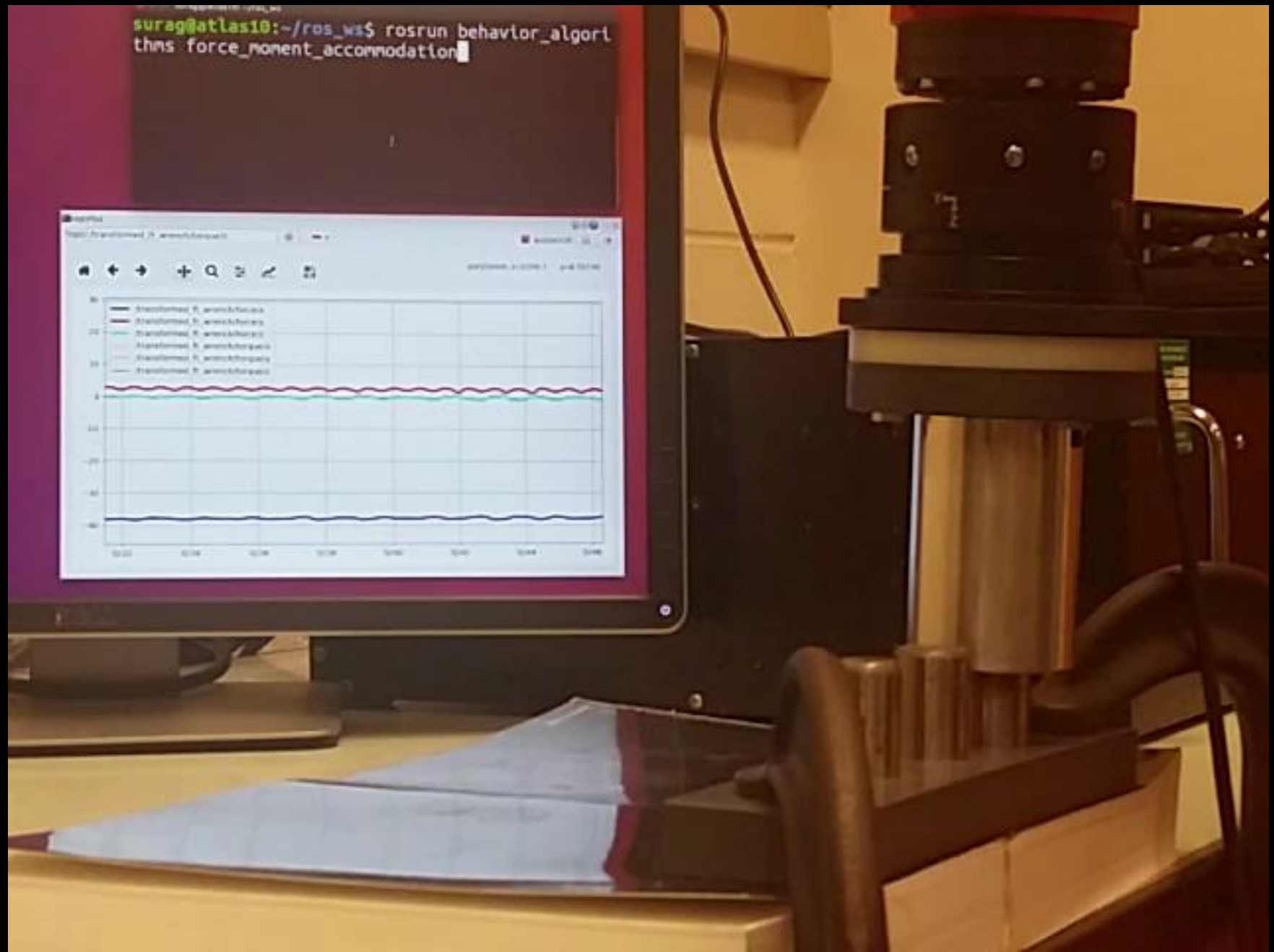

surag@atlas10:~/ros_ws$ rosrun behavior_algorith
ms position_targeting_with_force_limiting

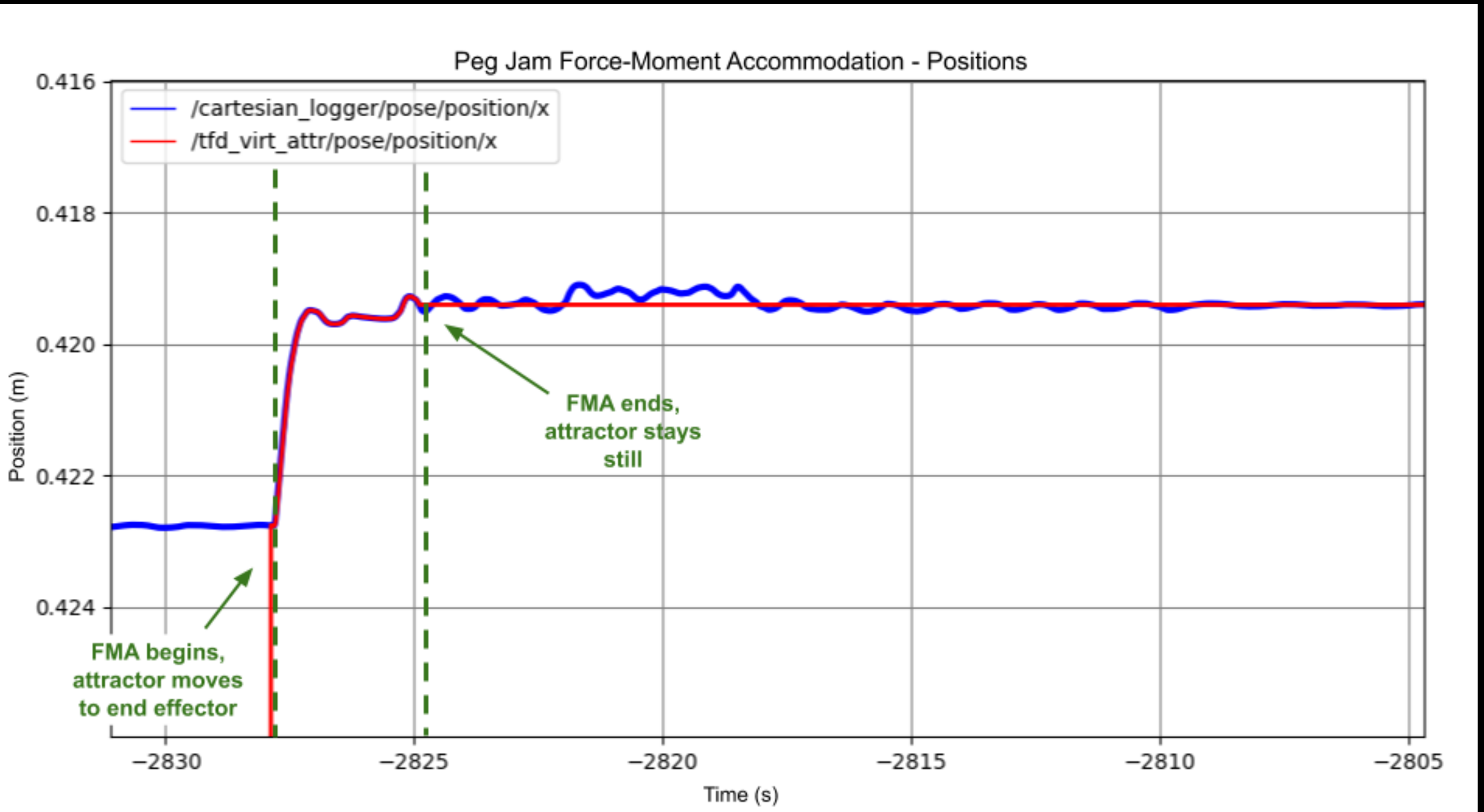

PTEL Force Threshold Crossed Demo: Force
/cartesian_logger/pose/position/x
/tfd_virt_attr/pose/position/x
/transformed_ft_wrench/force/x
Force threshold
crossed (1 N)
Force (N)
Time (s)
5
0
−5
−10
−15
−20
37
38
39
40
41
42

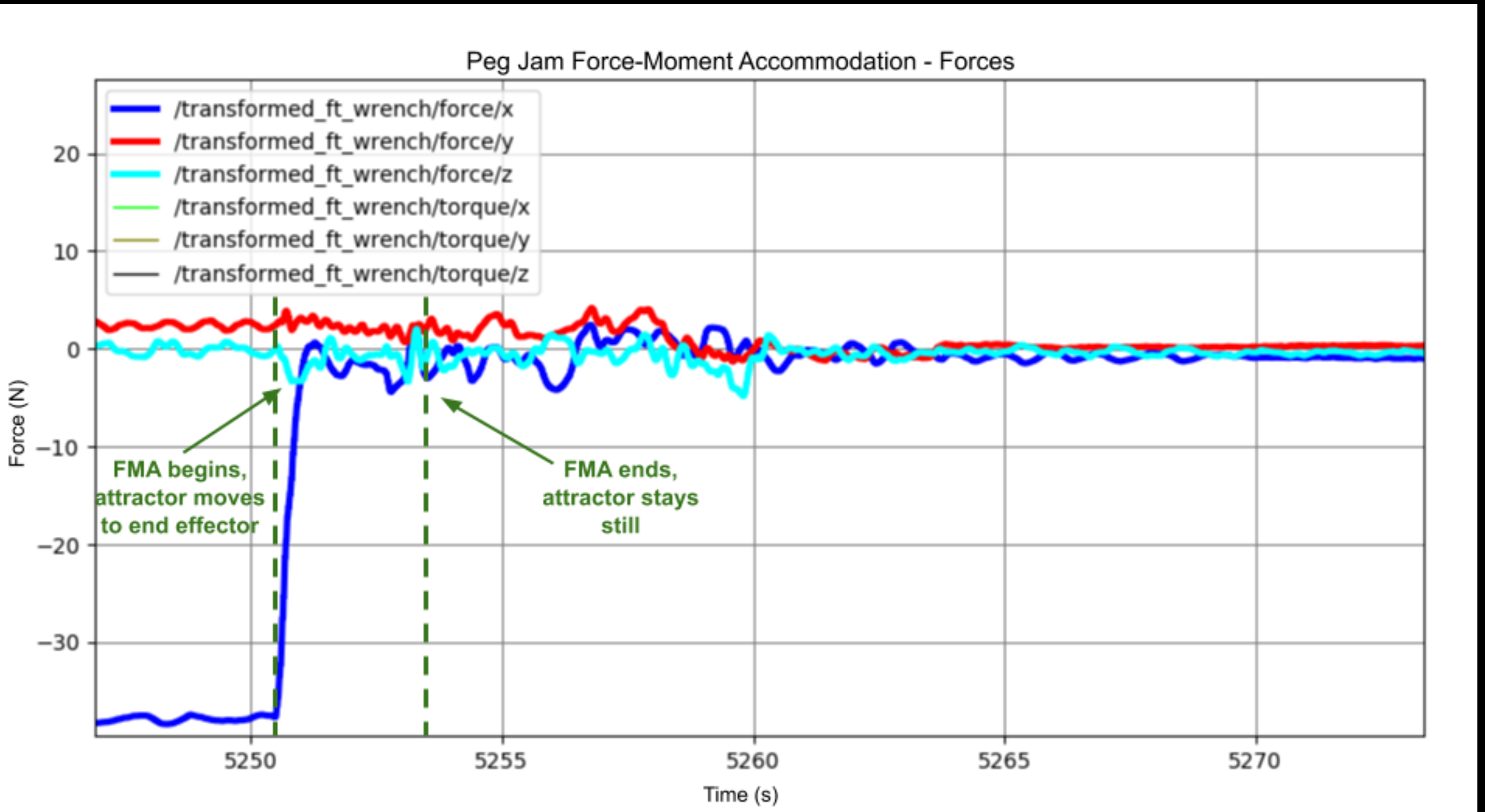
Skills – Reset Effort Equilibrium (REE)

```
while time_elapsed < end_time
{
      virtual_attractor_pose = current_pose;
}
```

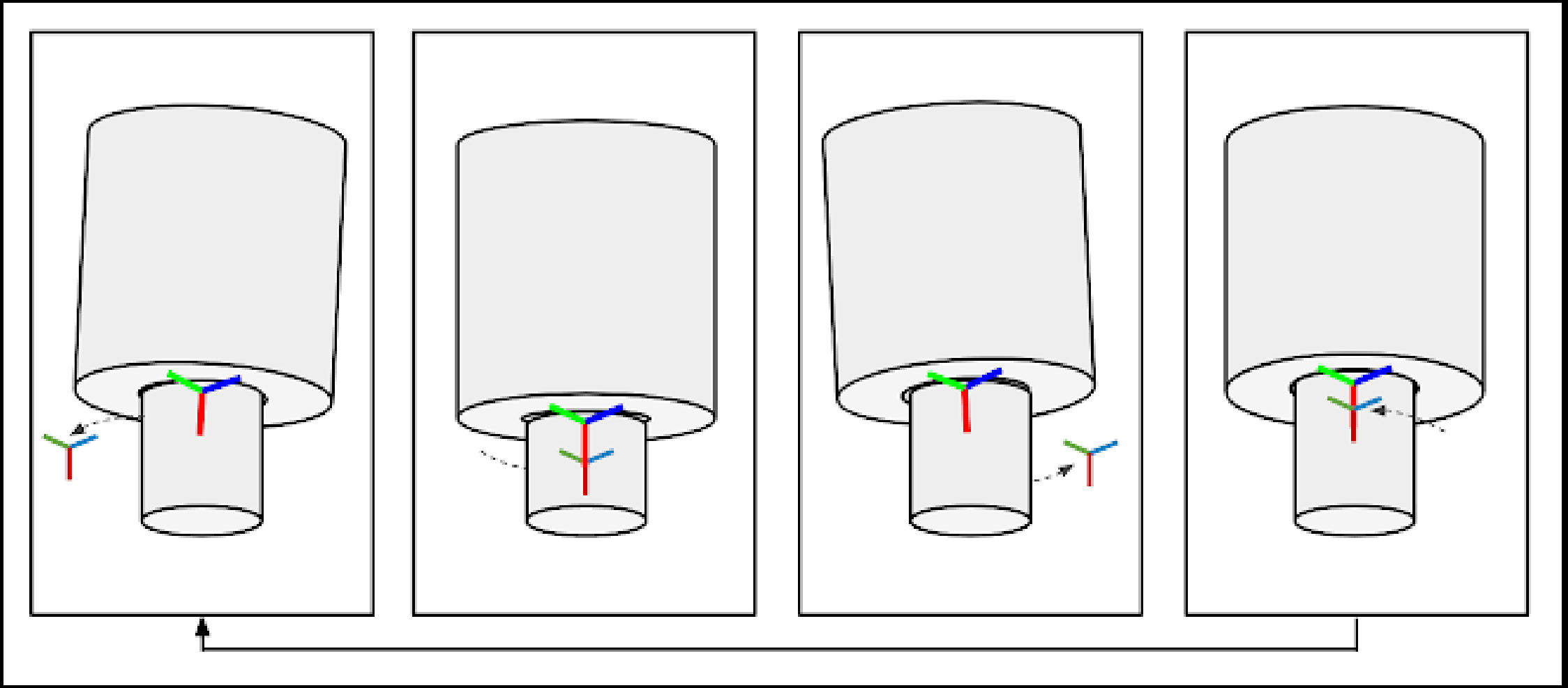
Sleeve on peg assembly: 90micron clearance; requires multiple skills to succeed



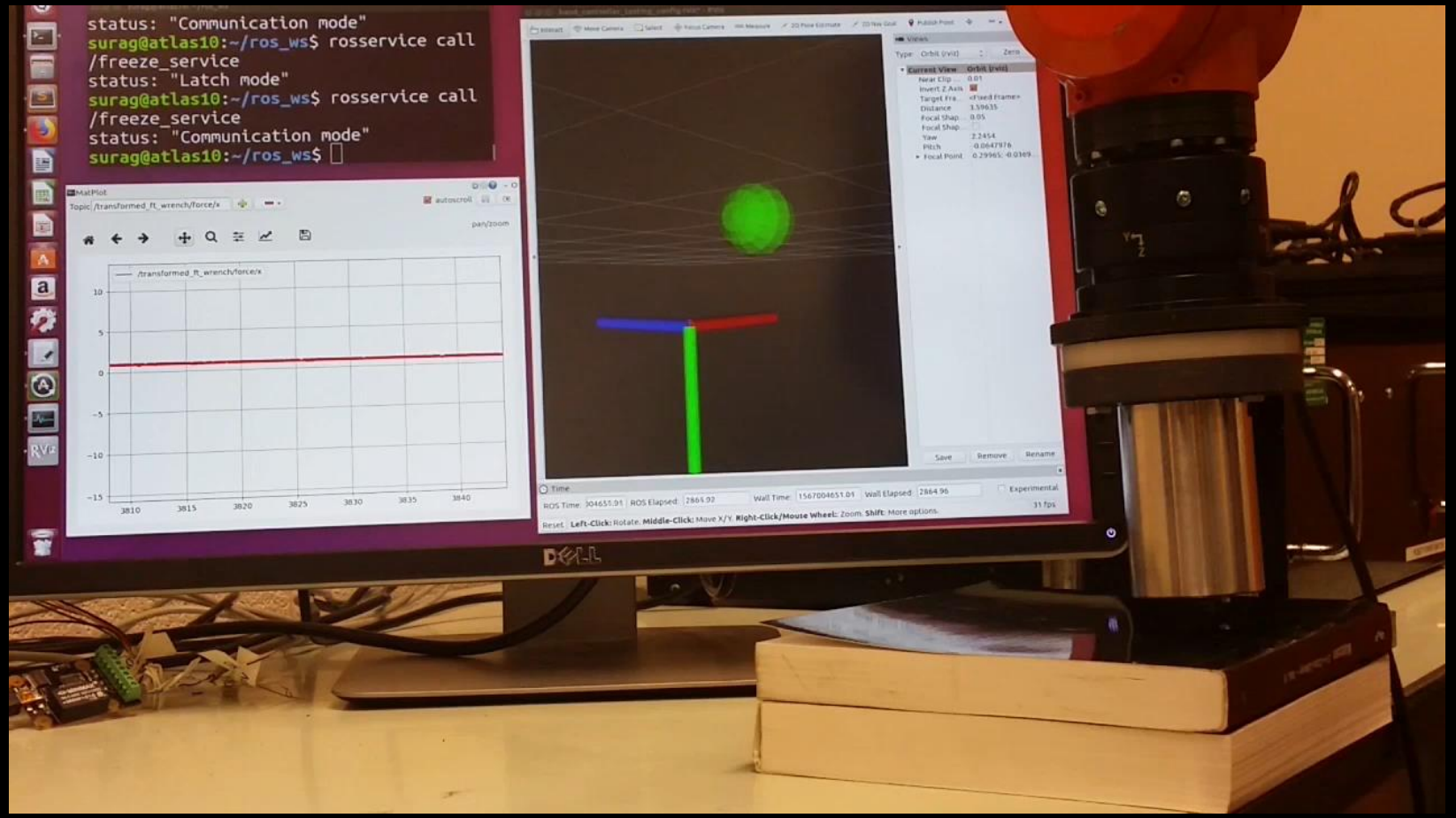

Peg Jam Force-Moment Accommodation - Positions
/cartesian_logger/pose/position/x
/tfd_virt_attr/pose/position/x
0.416
0.418
0.420
0.422
0.424
Position (m)
FMA ends,
attractor stays
still
FMA begins,
attractor moves
to end effector
−2830
−2825
−2820
−2815
−2810
−2805
Time (s)


Peg Jam Force-Moment Accommodation - Forces
/transformed_ft_wrench/force/x
/transformed_ft_wrench/force/y
/transformed_ft_wrench/force/z
/transformed_ft_wrench/torque/x
/transformed_ft_wrench/torque/y
/transformed_ft_wrench/torque/z
20
10
0
−10
−20
−30
Force (N)
FMA begins, attractor moves to end effector
FMA ends, attractor stays still
5250
5255
5260
5265
5270
Time (s)

Skills –Dither Soft Attractor



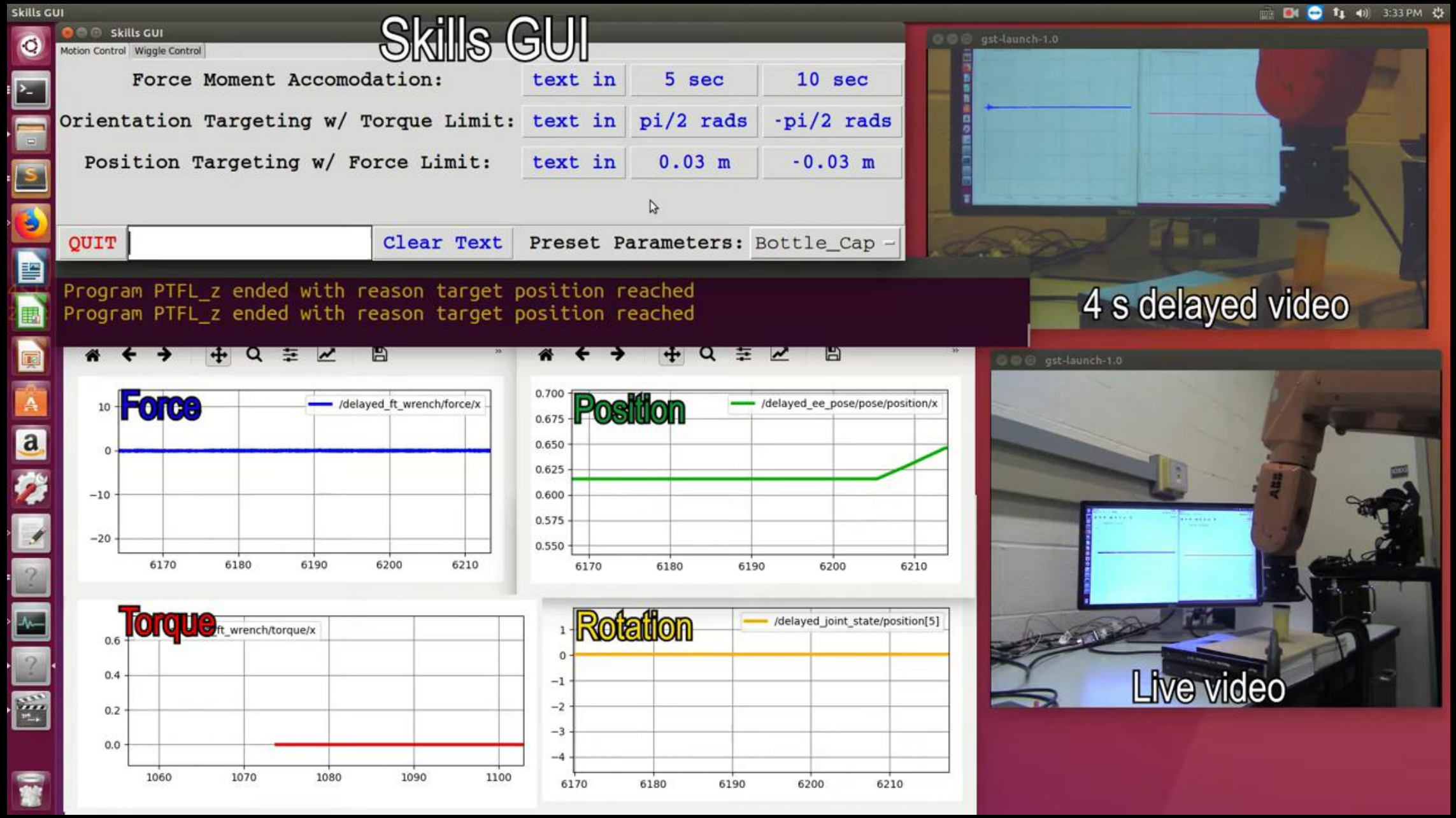
Skills – Bumpless Transition

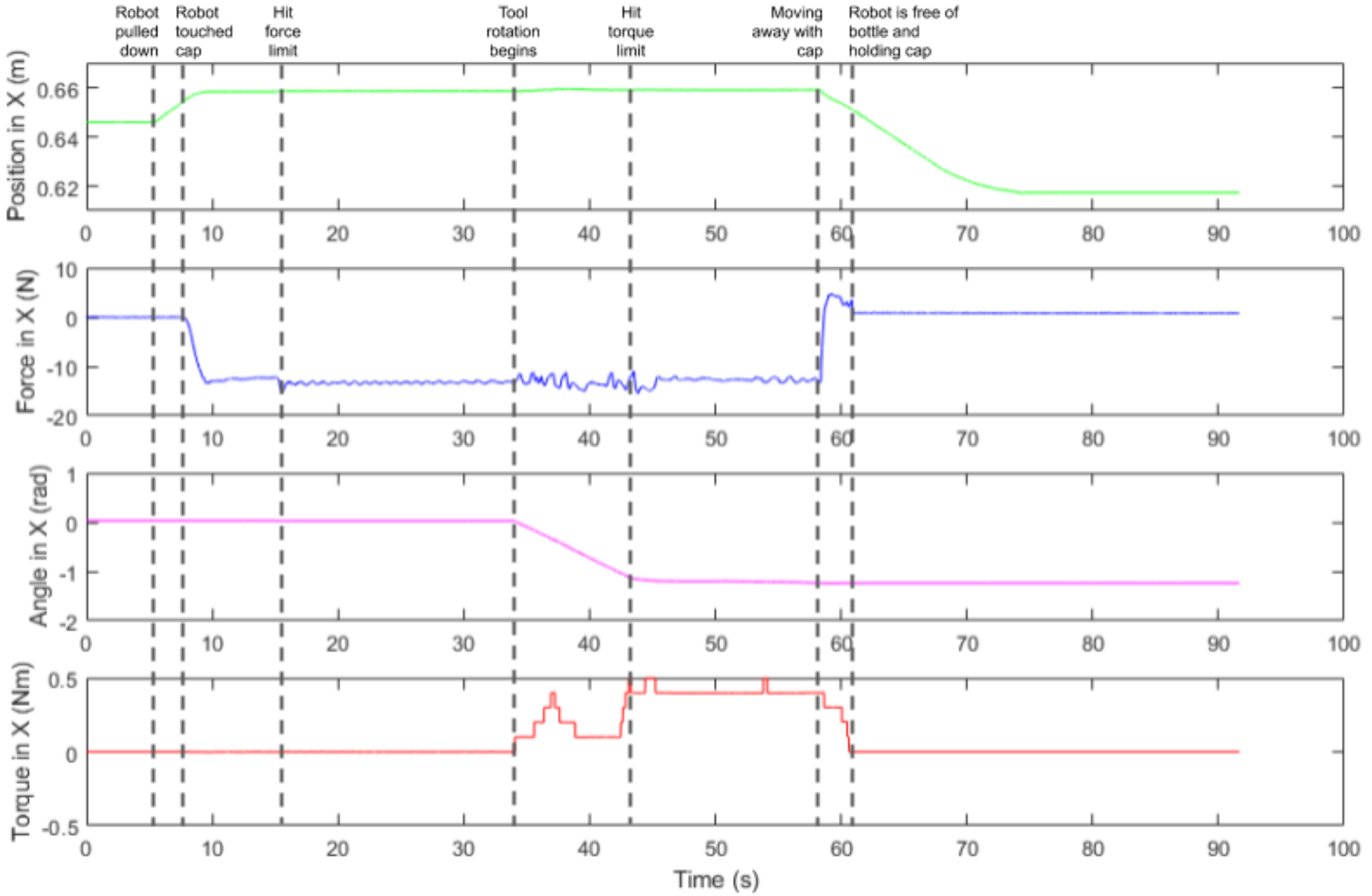


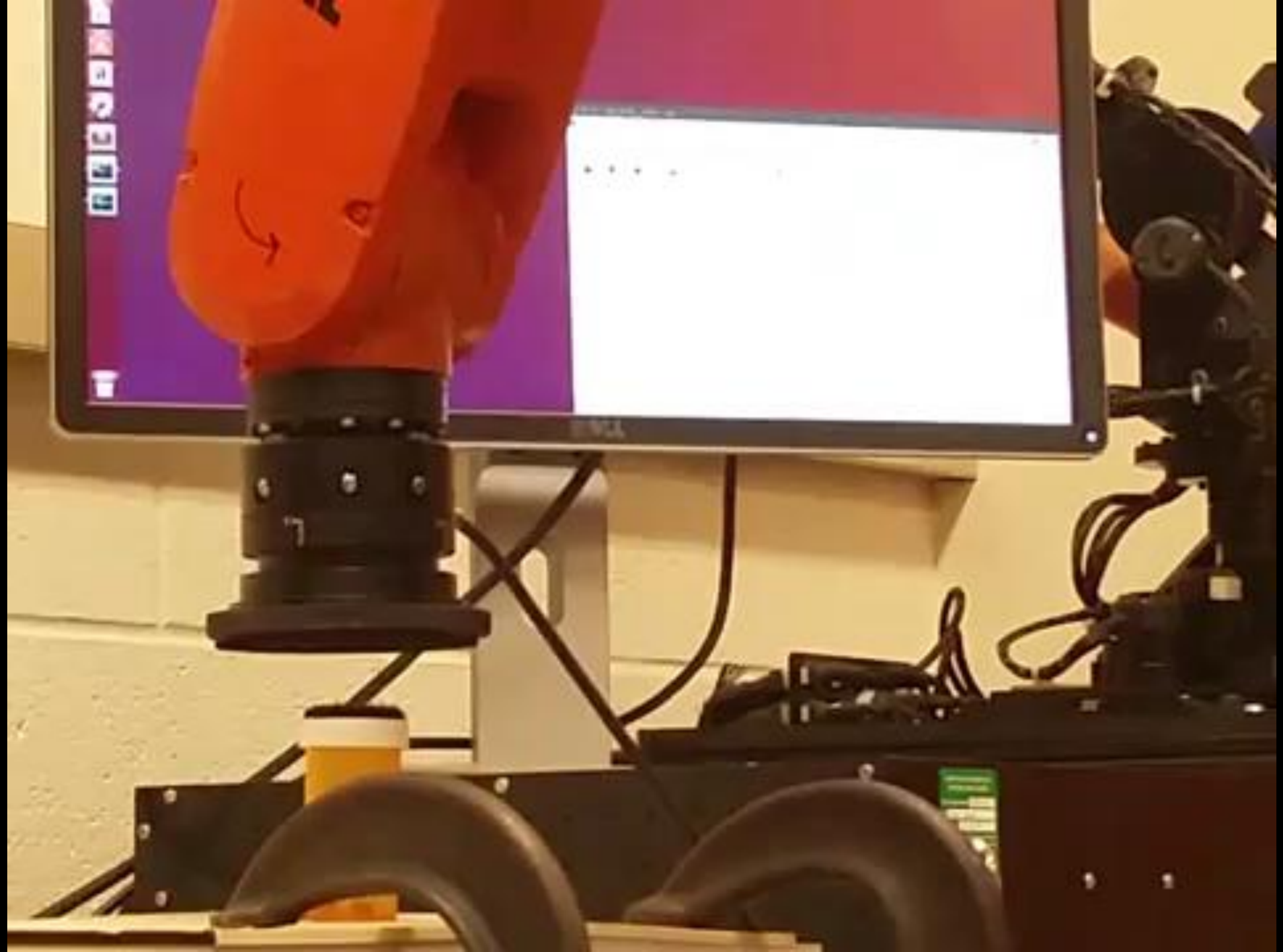
Skills – Bumpless Transition

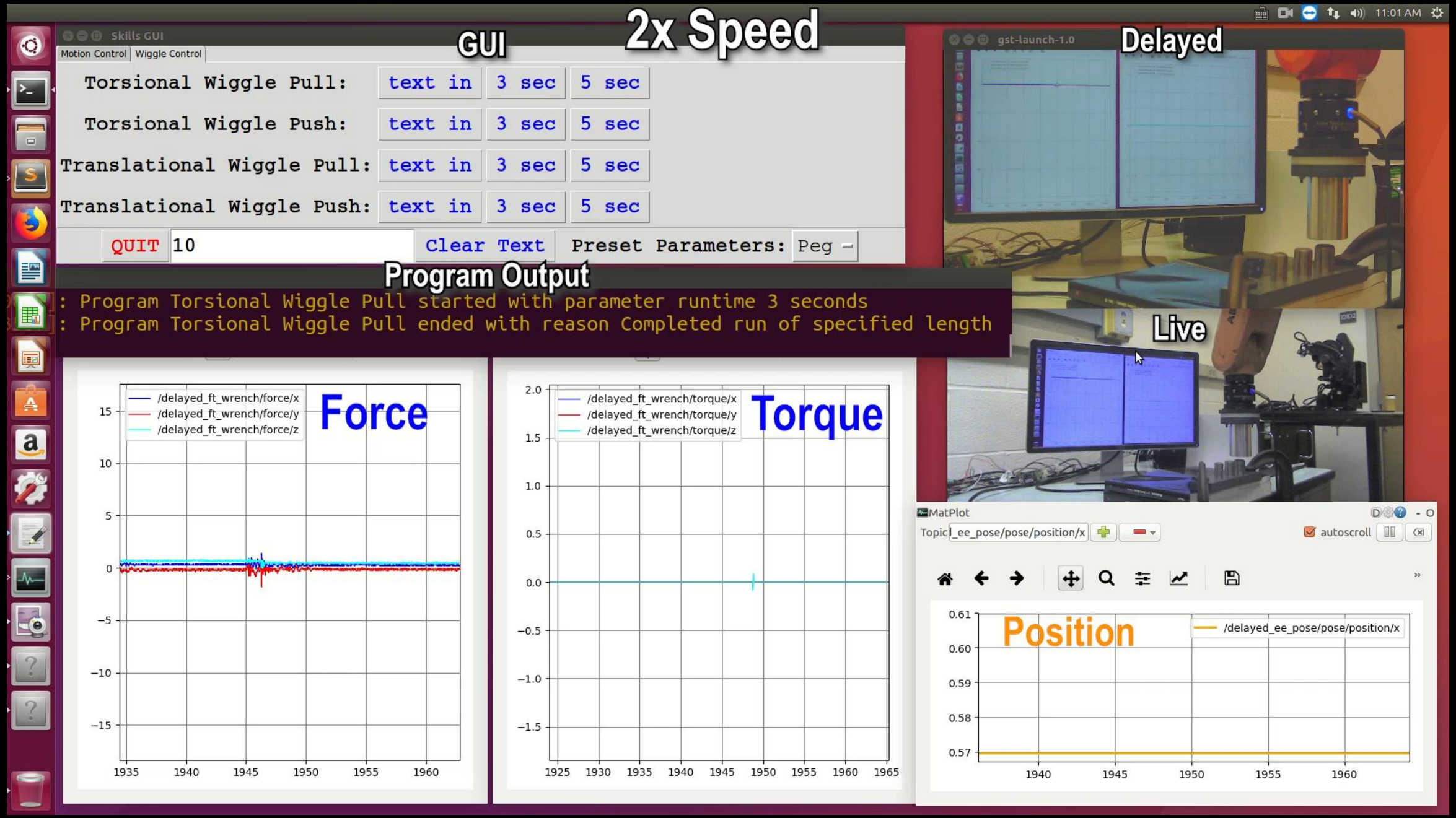
$$P_{virtual\ attractor} = \frac{F_{physical}}{K_{translational}} + P_{end\ effector} \qquad (4)$$

$$R_{virtual\ attractor} = \frac{T_{physical}}{K_{angular}} + R_{end\ effector} \qquad (5)$$


Skills GUI
Skills GUI
Motion Control
Wiggle Control
Force Moment Accomodation:
text in
5 sec
10 sec
Orientation Targeting w/ Torque Limit:
text in
pi/2 rads
-pi/2 rads
Position Targeting w/ Force Limit:
text in
0.03 m
-0.03 m
QUIT
Clear Text
Preset Parameters:
Bottle_Cap
Program PTFL_z ended with reason target position reached
Program PTFL_z ended with reason target position reached
gst-launch-1.0
4 s delayed video
Force
/delayed_ft_wrench/force/x
Position
/delayed_ee_pose/pose/position/x
Torque
Rotation
/delayed_joint_state/position[5]
gst-launch-1.0
Live video


Robot pulled down
Robot touched cap
Hit force limit
Tool rotation begins
Hit torque limit
Moving away with cap
Robot is free of bottle and holding cap
Position in X (m)
Force in X (N)
Angle in X (rad)
Torque in X (Nm)
Time (s)




GUI
2x Speed
Skills GUI
Motion Control
Wiggle Control
Torsional Wiggle Pull: text in 3 sec 5 sec
Torsional Wiggle Push: text in 3 sec 5 sec
Translational Wiggle Pull: text in 3 sec 5 sec
Translational Wiggle Push: text in 3 sec 5 sec
QUIT
10
Clear Text
Preset Parameters: Peg
Program Output
: Program Torsional Wiggle Pull started with parameter runtime 3 seconds
: Program Torsional Wiggle Pull ended with reason Completed run of specified length
gst-launch-1.0
Delayed
Live
Force
/delayed_ft_wrench/force/x
/delayed_ft_wrench/force/y
/delayed_ft_wrench/force/z
Torque
/delayed_ft_wrench/torque/x
/delayed_ft_wrench/torque/y
/delayed_ft_wrench/torque/z
MatPlot
Topic
_ee_pose/pose/position/x
autoscroll
Position
/delayed_ee_pose/pose/position/x

Conclusions:

- Compliant-motion control is essential for many necessary operations
  - E.g. mechanical assembly; using buttons, levers, handles, doors; scraping, grinding, cutting;
- Contact stability is a concern with force/torque feedback
  - Stability can be predicted with passivity analysis
  - Alternative control systems and parameters can be analyzed in a common framework
- Exploiting compliant-motion control can be done via skills and soft attractors
  - Force/torque limited attractor goals
  - Bumpless transitions to/from position control
  - Relief of equilibrium forces/torques
  - Soft dither behaviors for jam clearing

Thanks to:

- Matt Haberbusch
- Surag Balajepalli
- Roshan Kasal
- Rahul Pokharna
- Support from NASA Goddard Space Flight Center