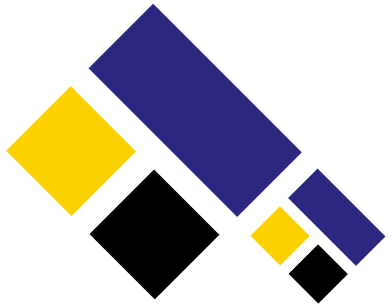


Preliminary data-driven controller for Affetto

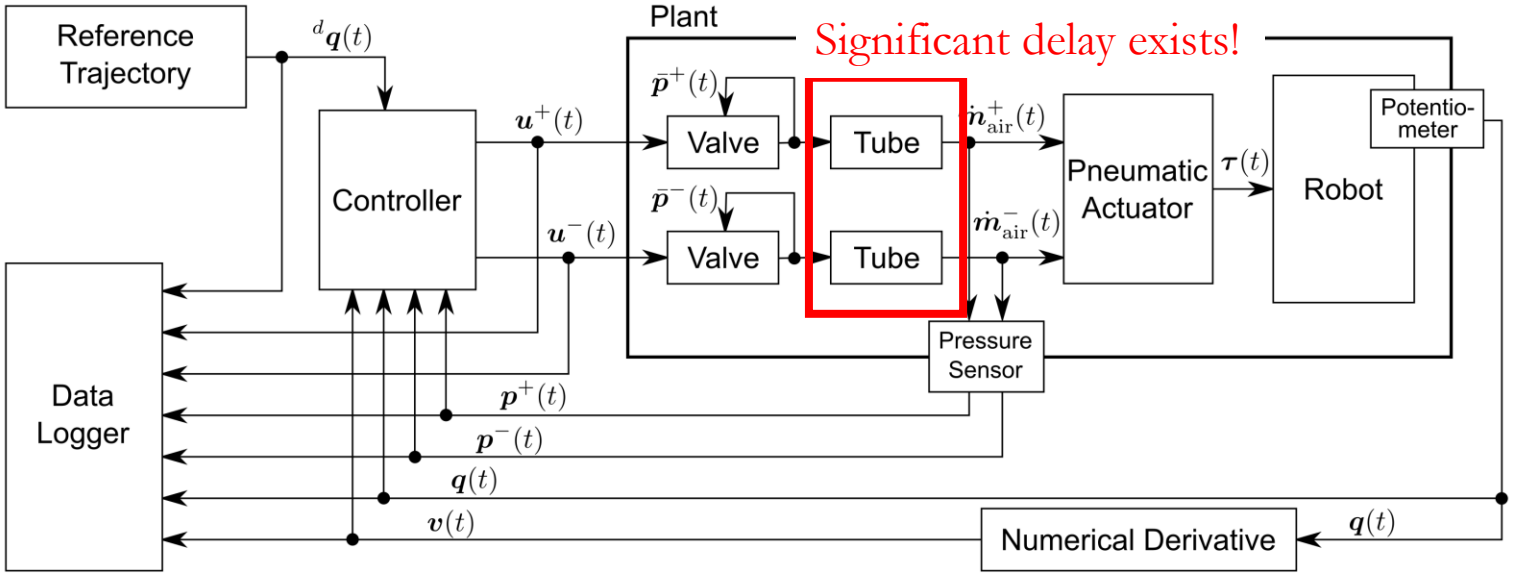
Hiroshi Atsuta

Research goal

NEDO Affetto

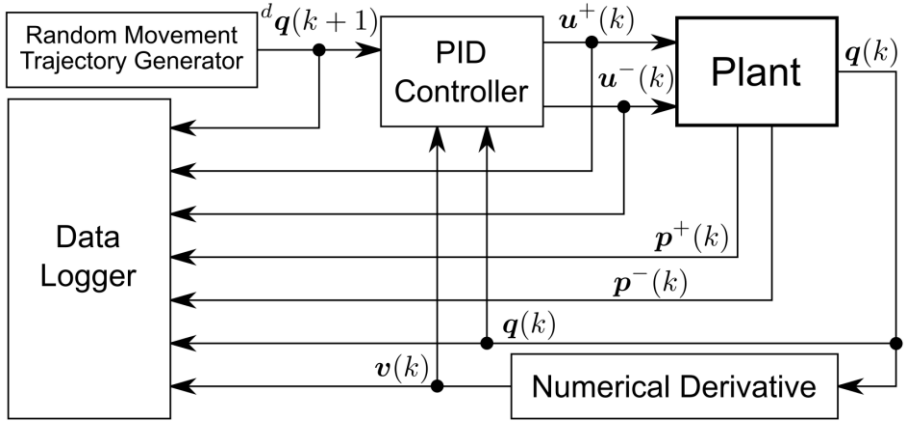
- Development of controller for NEDO Affetto
  - Mechanical configuration of NEDO Affetto
    - 13DOFs
    - Driven pneumatically
    - 13 potentiometers and 26 pressure sensors
  - Difficulties in motion control
    - Redundancy
    - Unknown kinematic model of the robot
    - Unknown dynamic model of the robot and actuators
  - Our approach
    - Learning from demonstration with kinesthetic teaching
    - Data-driven control of multi-DOF pneumatic actuators
    - Efficient learning with reservoir computing (RC)

Control framework of Affetto



- Goal: Development of controller for reference trajectory tracking
  - Consider the robot, pneumatic actuators, tubes, valves and sensors as a plant
  - The plant takes pressure valve commands as input and outputs joint positions and pressures
  - The controller takes reference trajectory and sensory value and generates valve commands
  - Unknown significant delay exists in tubes between valves and pneumatic actuators
  - The goal is to develop a data-driven controller for trajectory tracking

Random motion data collection



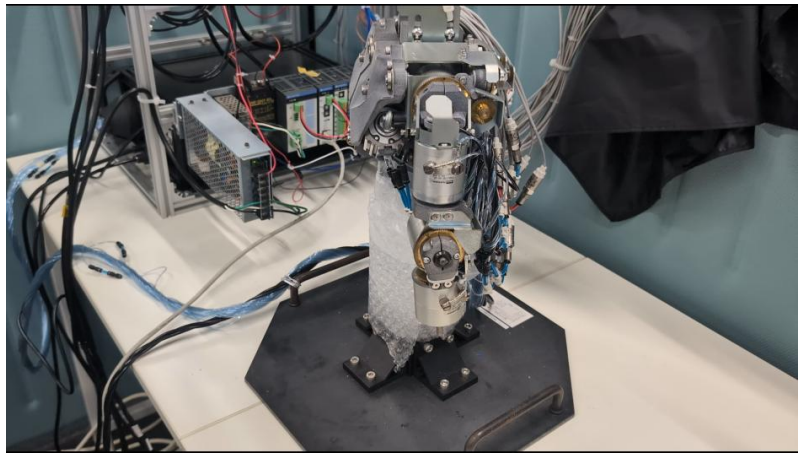
- Collection of dynamic motion data between control input and observable states
  - Movement trajectory based on random walk for each joint is generated
  - Poorly-tuned PID controller is applied to move the robot
  - Store control inputs ($\boldsymbol{u}^+, \boldsymbol{u}^-$) and observable states ($\boldsymbol{p}^+, \boldsymbol{p}^-, \boldsymbol{q}, \boldsymbol{v}$) as time series data
  - Variations of random movement trajectories:
    - Time scale: slow, middle or fast
    - Continuity: discontinuous or continuous
    - Synchronicity: synchronous or asynchronous

Position-based random motion generation

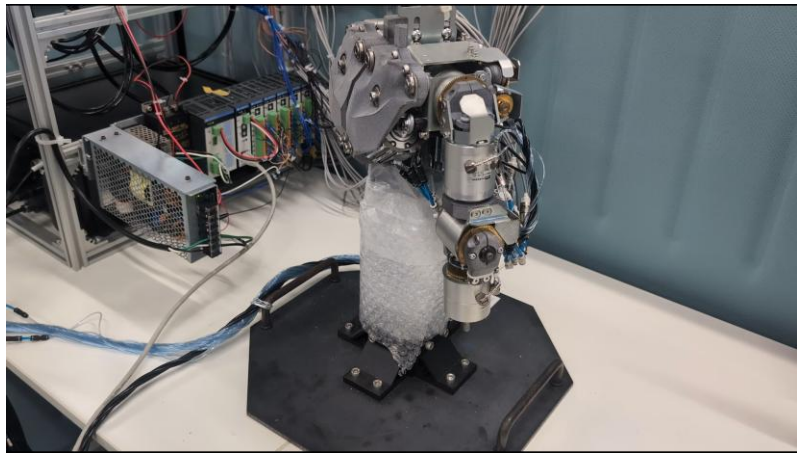
- Random motion generation
  - Including rich dynamics in motion is essential to make learning work well
  - However, directly generating valve commands makes robot easily stuck
    - As motion ranges of joints are very small, each joint easily reach the limit of the movement
    - Making pressure values at random between antagonistic valves will not generate smooth movement
  - Even poor, PID control can make smooth motion with position-based reference
    - PID gains are not necessary to be well tuned because it is not necessary to track the reference correctly
  - To include diverse dynamics in data set, make trajectory with the following properties
    - Time scale: Fast, middle, slow
    - Synchronicity, : Synchronous asynchronous
    - Continuity: Continuous, Discontinuous

Collecting random motion

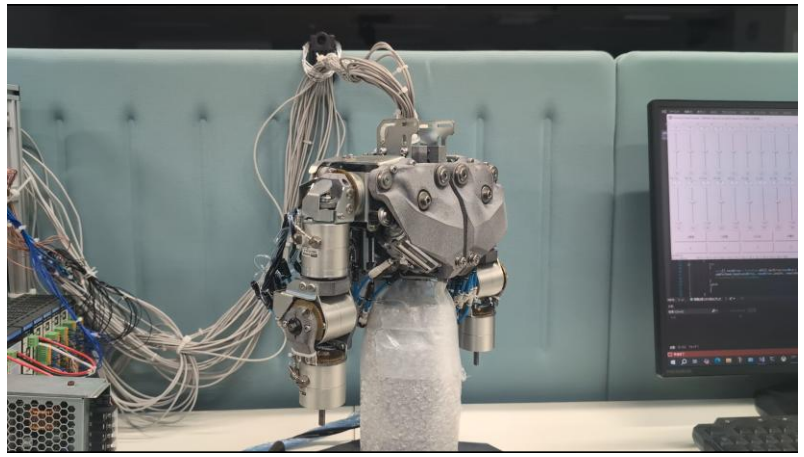
Left elbow



Left arm



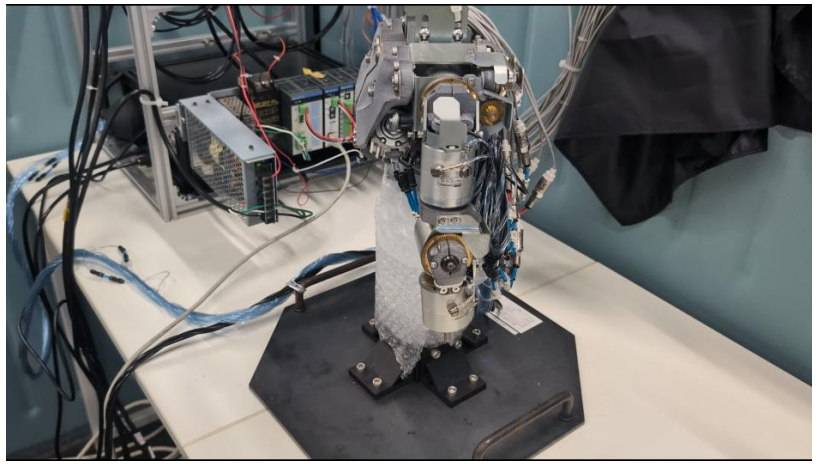
Whole body



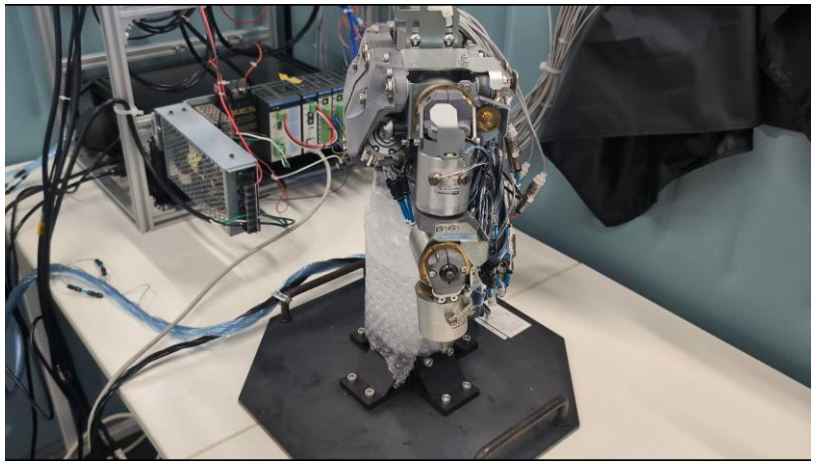
- Gradually increase the degree of freedom (DOF)
  - Need to select many parameters for preprocessing and neural networks
  - 1 DOF: Determine key parameters e.g., time delay
  - 4 DOF: Evaluate effect of asynchronous movement across multi-joints
  - 13 DOF: Confirm performance of selected parameters

Making diverse trajectories (time scale)

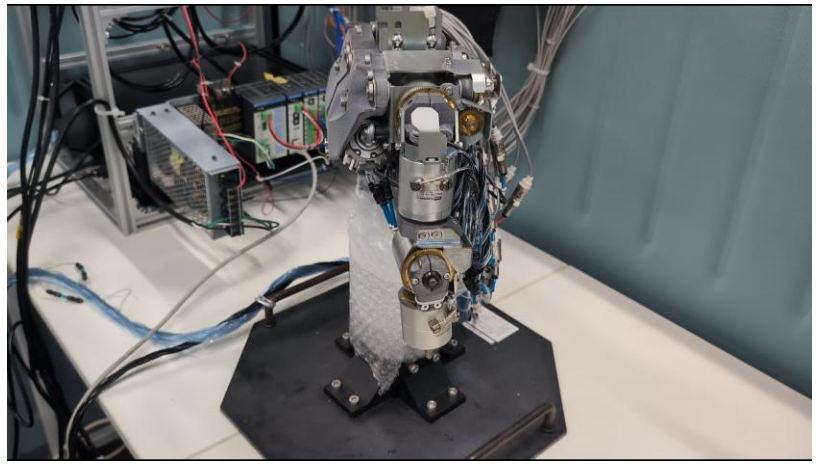
Fast



Middle



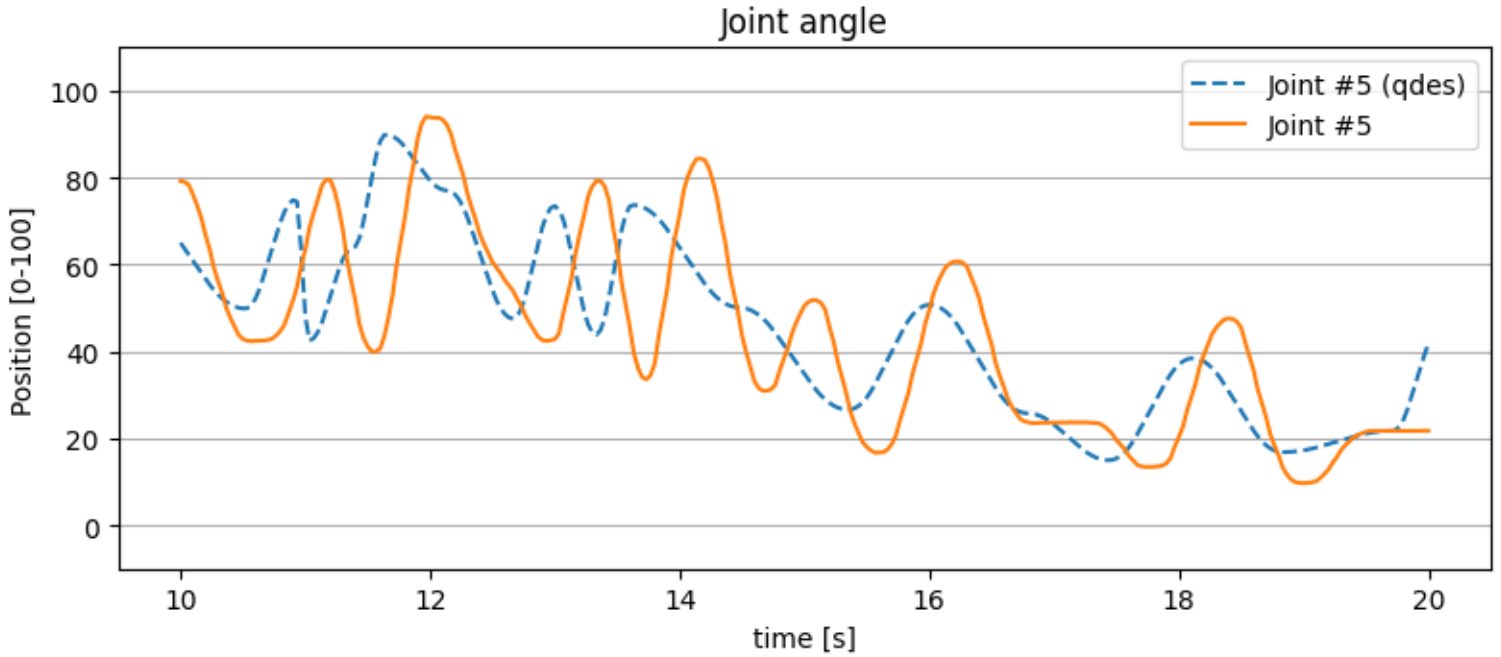
Slow



- Update references in various duration
  - Fast: Update randomly between 0.1s to 1.0s
  - Middle: Update randomly between 1.0s to 2.0s
  - Slow: Update randomly between 2.0s to 4.0s

Making diverse trajectories (time scale)

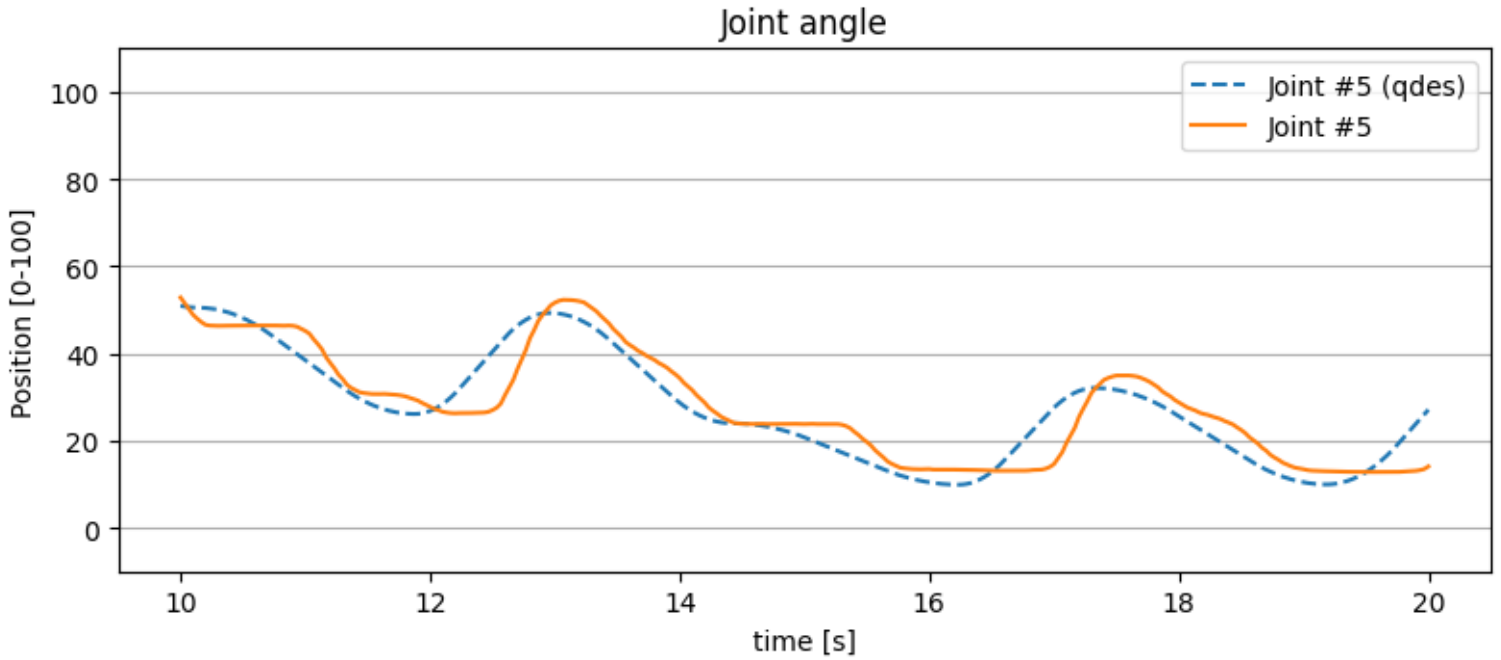
Fast



- Update references in various duration
  - Fast: Update randomly between 0.1s to 1.0s
  - Middle: Update randomly between 1.0s to 2.0s
  - Slow: Update randomly between 2.0s to 4.0s

Making diverse trajectories (time scale)

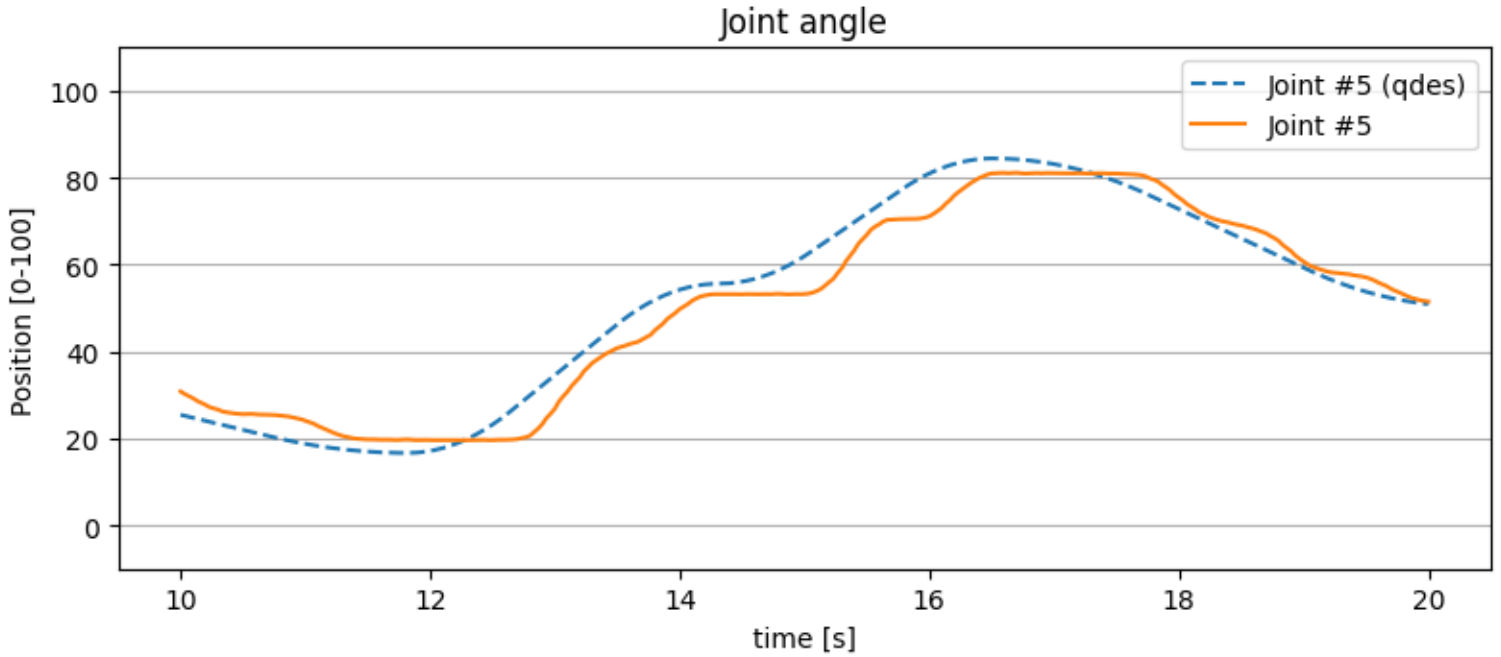
Middle



- Update references in various duration
  - Fast: Update randomly between 0.1s to 1.0s
  - Middle: Update randomly between 1.0s to 2.0s
  - Slow: Update randomly between 2.0s to 4.0s

Making diverse trajectories (time scale)

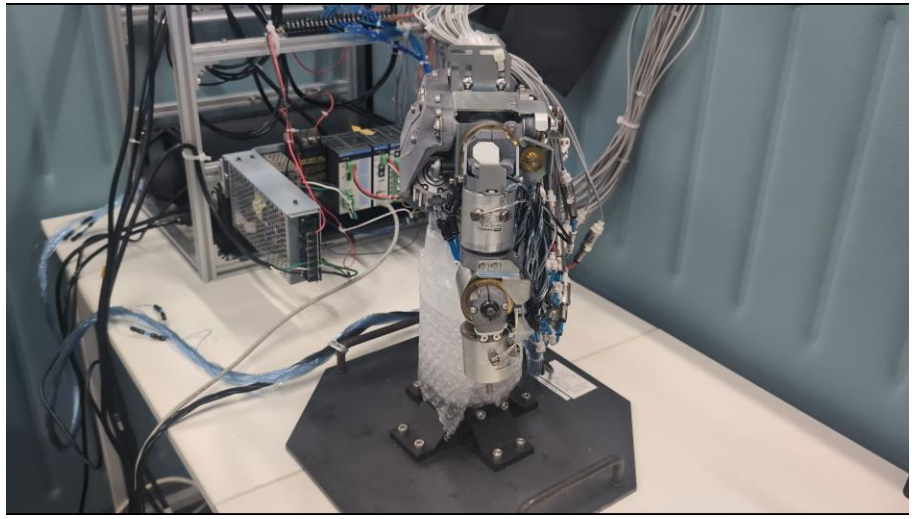
Slow



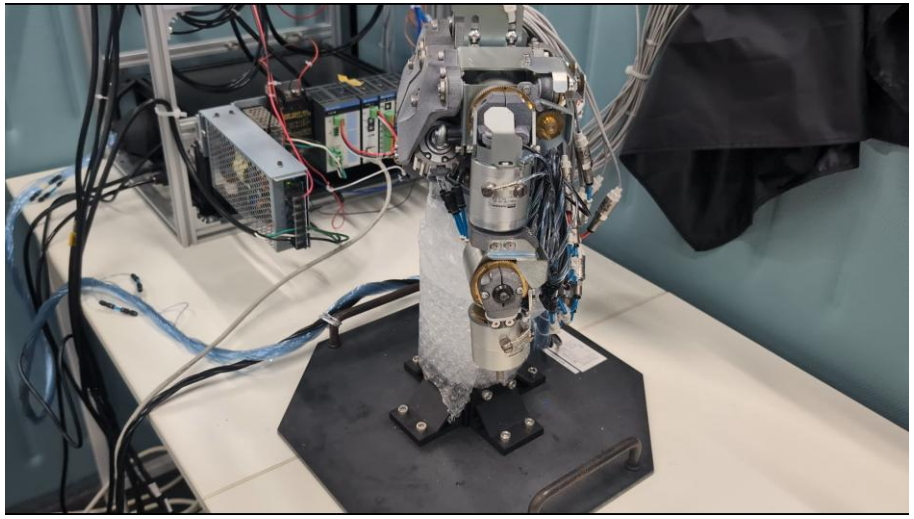
- Update references in various duration
  - Fast: Update randomly between 0.1s to 1.0s
  - Middle: Update randomly between 1.0s to 2.0s
  - Slow: Update randomly between 2.0s to 4.0s

Making diverse trajectories (continuity)

Discontinuous



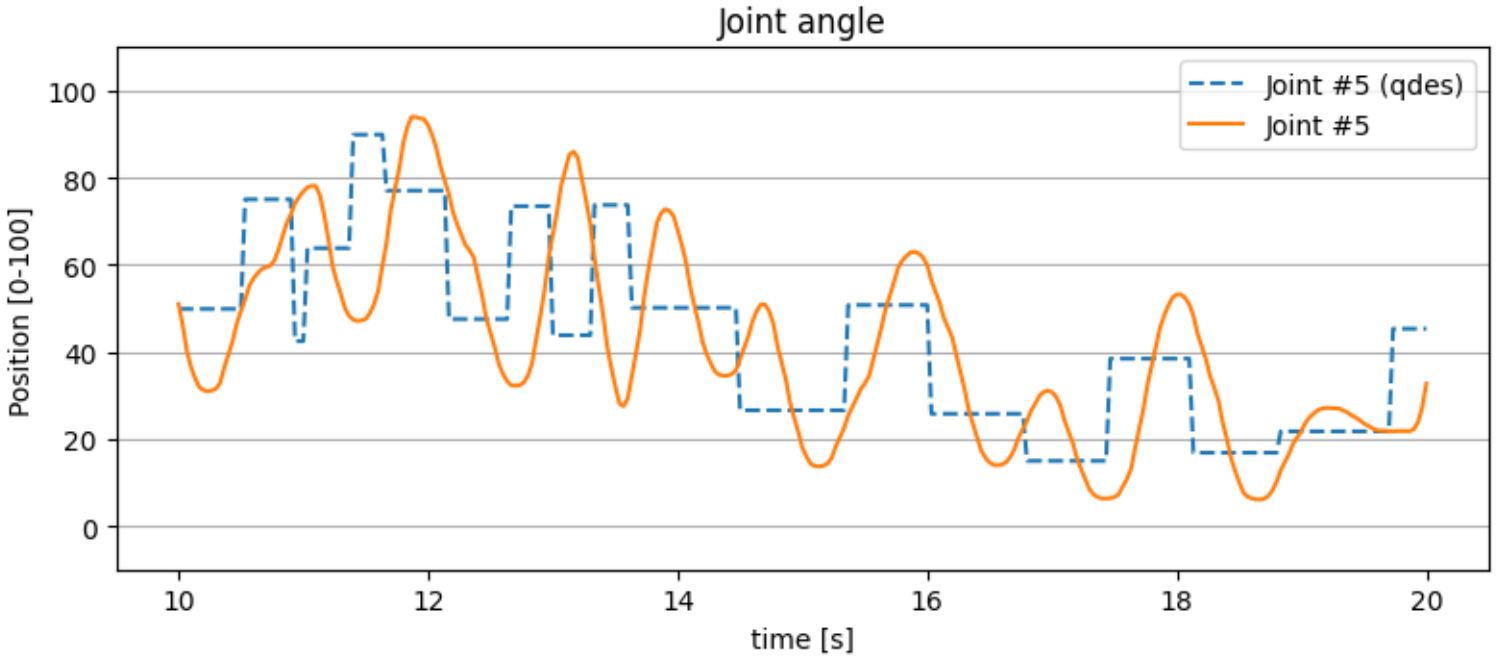
Continuous



- Update references in continuous / discontinuous fashion
  - Discontinuous: Generate reference with step function
  - Continuous: Generate reference in trapezoidal-velocity method

Making diverse trajectories (continuity)

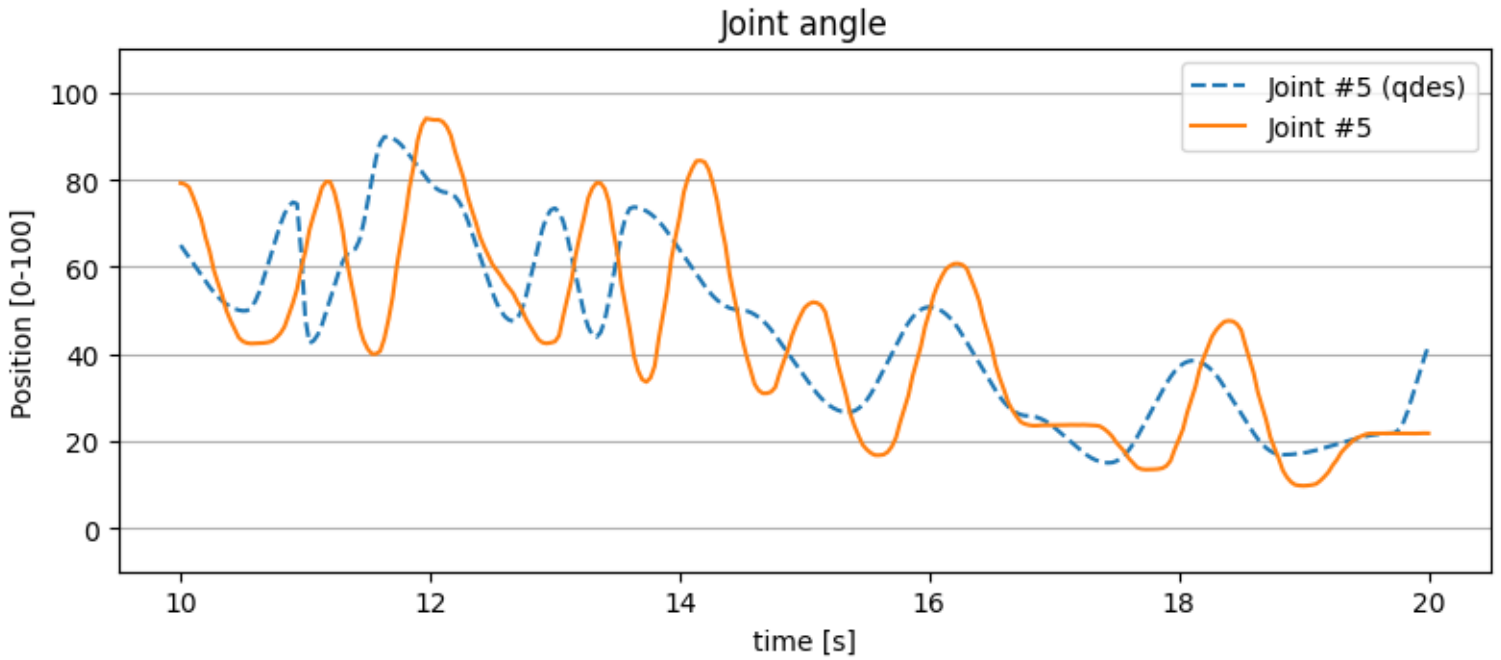
Discontinuous



- Update references in continuous / discontinuous fashion
  - Discontinuous: Generate reference with step function
  - Continuous: Generate reference in trapezoidal-velocity method

Making diverse trajectories (continuity)

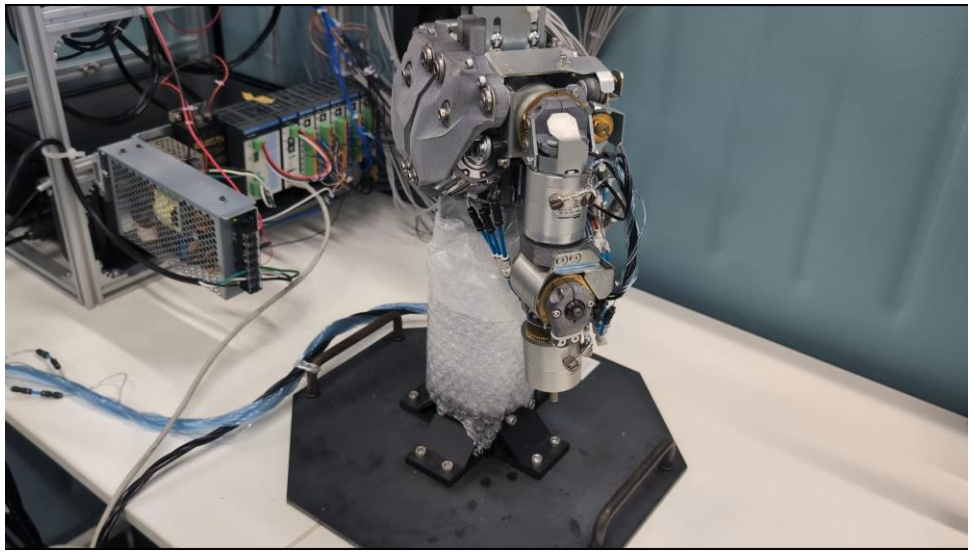
Continuous



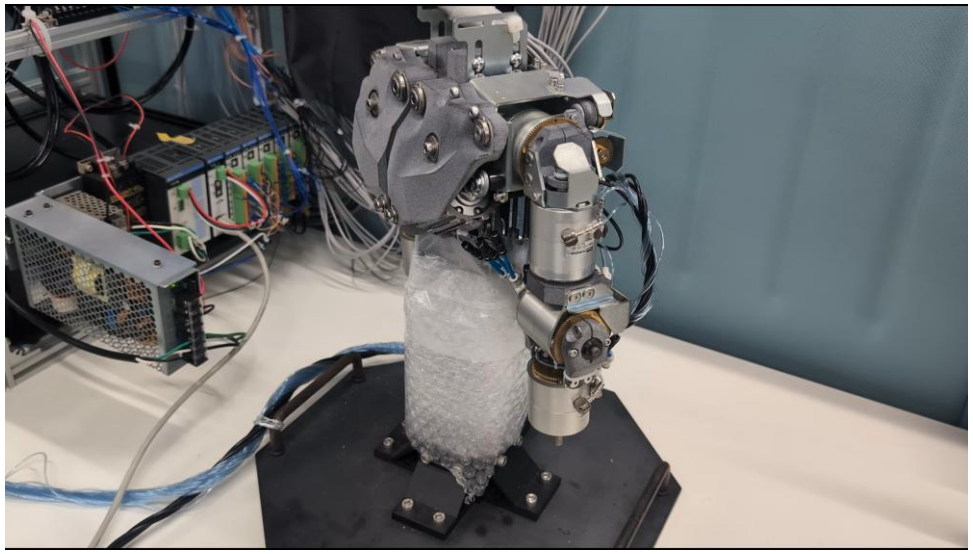
- Update references in continuous / discontinuous fashion
  - Discontinuous: Generate reference with step function
  - Continuous: Generate reference in trapezoidal-velocity method

Making diverse trajectories (synchronicity)

Synchronous



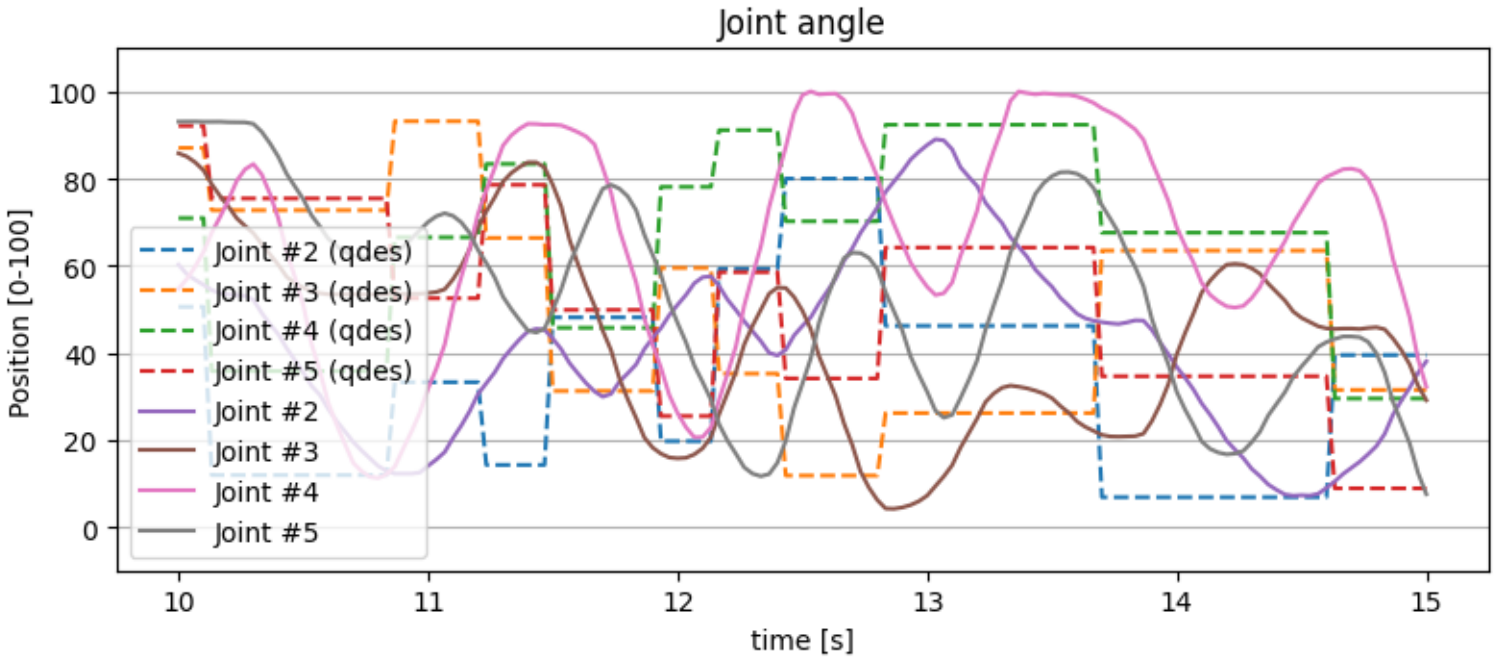
Asynchronous



- Update references in synchronous / asynchronous way across multi-joints
  - Synchronous: Update all joint references at the same timing
  - Asynchronous: Update each joint reference independently

Making diverse trajectories (synchronicity)

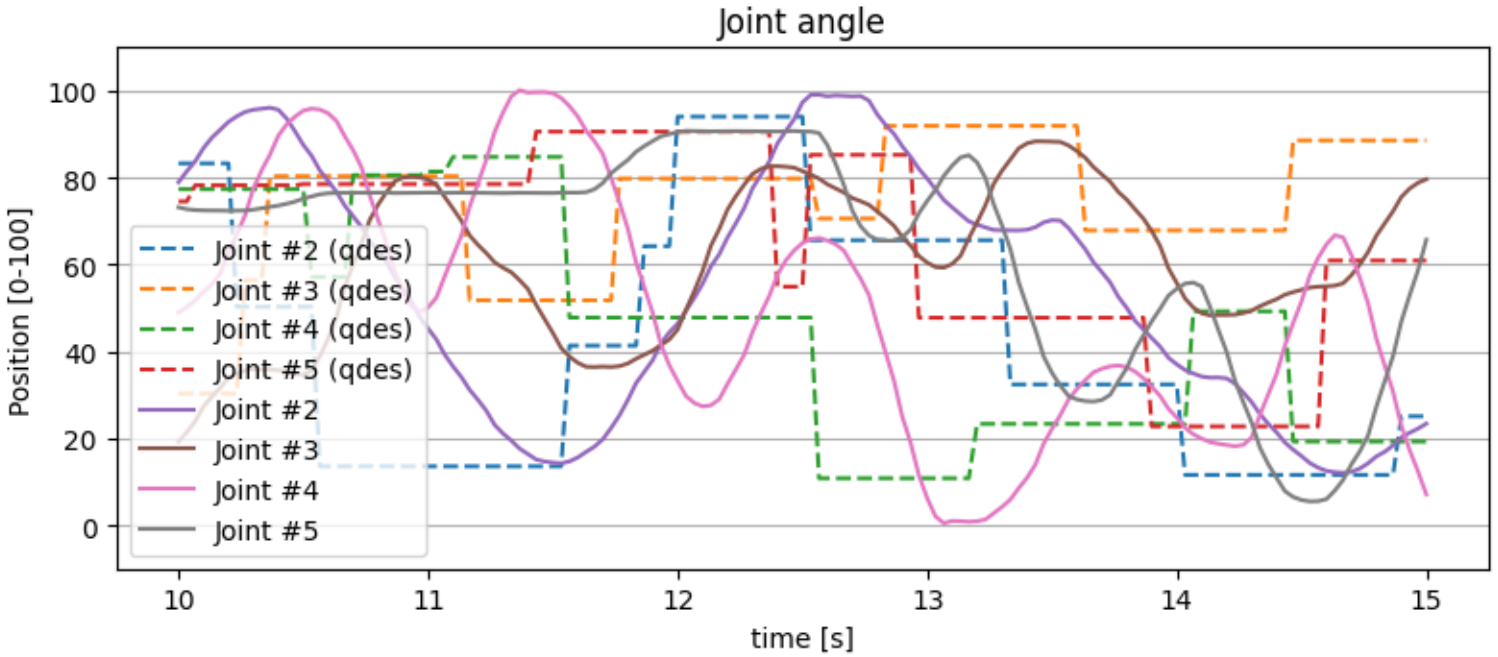
Synchronous



- Update references in synchronous / asynchronous across multi-joints
  - Synchronous: Update all joint references at the same timing
  - Asynchronous: Update each joint reference independently

Making diverse trajectories (synchronicity)

Asynchronous

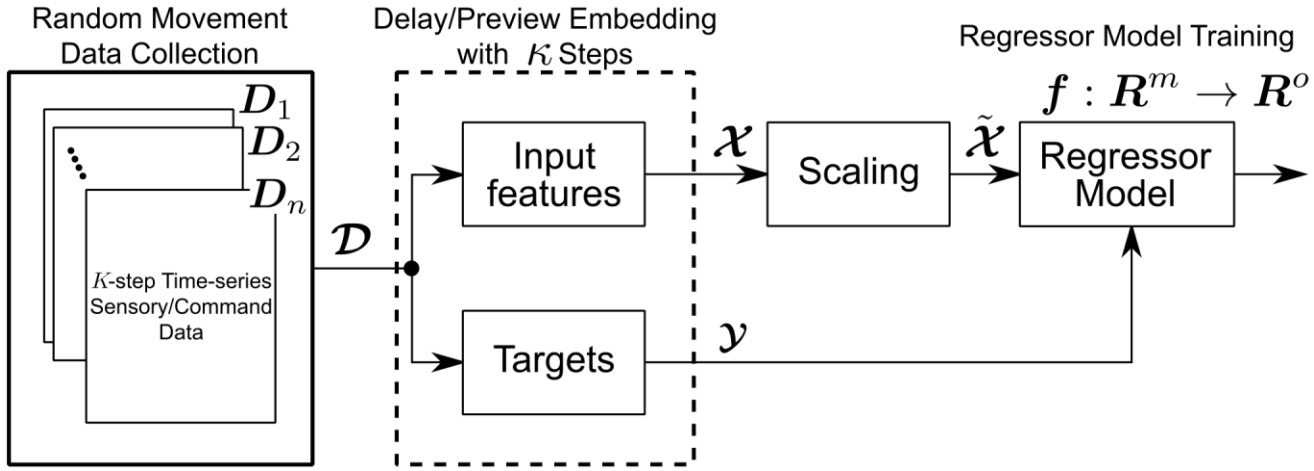


- Update references in synchronous / asynchronous across multi-joints
  - Synchronous: Update all joint references at the same timing
  - Asynchronous: Update each joint reference independently

Amount of collected motion data

- Collect 60-second motion 100 times for each data set
  - Left elbow (1 DOF)
    - 60s × {Fast, middle, slow} × {Discontinuous, Continuous} × 100 = 10 h
    - Sampling time: 0.033s (30Hz) → Total data size = 1,080,000 points, Dimension = 6
  - Left arm (4 DOF)
    - 60s × {Fast, middle, slow} × {Discont., Cont.} × {Synchronous, Asynchronous} × 100 = 20 h
    - Sampling time: 0.033s (30Hz) → Total data size = 2,160,000 points, Dimension = 24
  - Whole body (13 DOF)
    - 60s × {Fast, middle, slow} × {Cont.} × {Asynchronous} × 120 = 6 h
    - Sampling time: 0.033s (30Hz) → Total data size = 648,000 points, Dimension = 6×13 = 78
  - However, not all the data sets are used for training to ensure fair comparisons

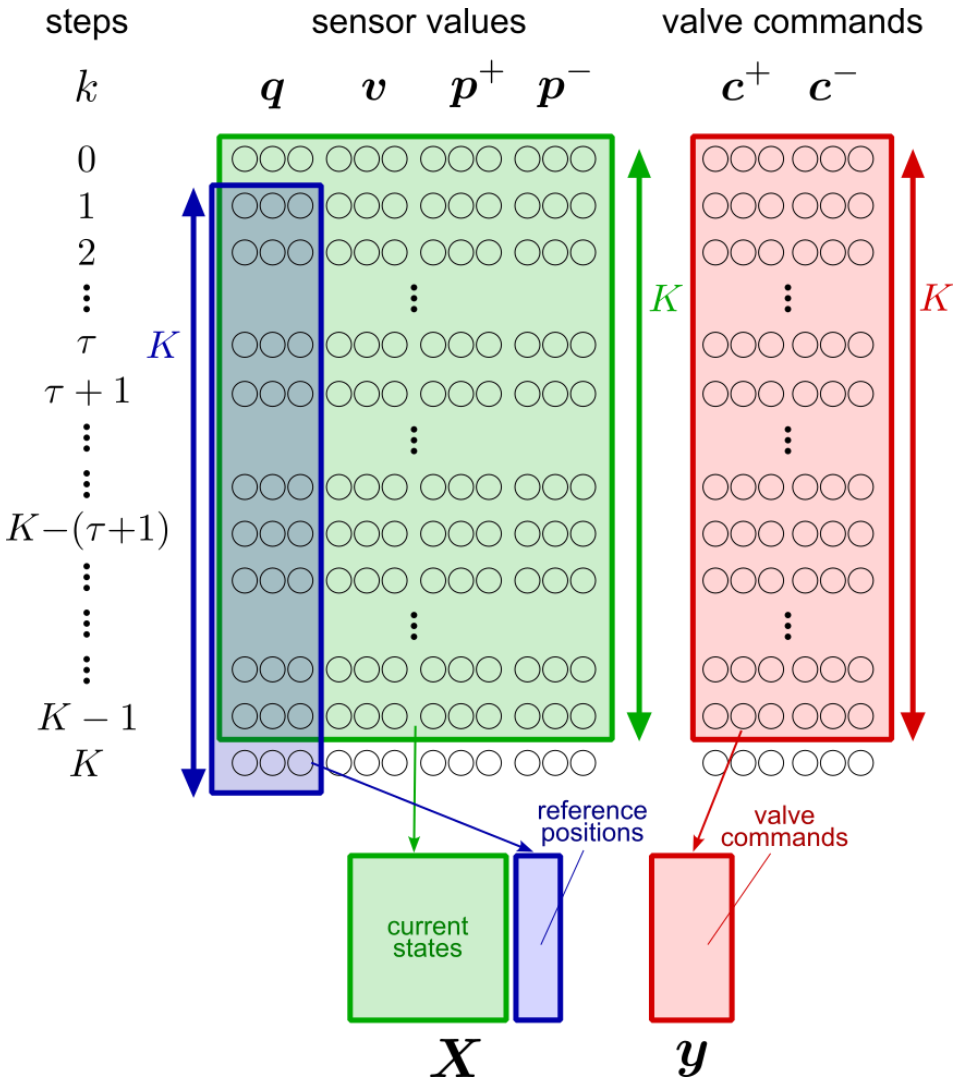
Training framework



- Training of regressor models using collected movement data sets
  - Preprocessing: Delay/Preview embedding with τ steps and scaling
  - Regressor models: find $\boldsymbol{f}$ so that $\boldsymbol{\mathcal{Y}} = \boldsymbol{f}(\tilde{\boldsymbol{\mathcal{X}}})$
    - Linear regressor
    - Multi-layered perceptron regressor
      - Neuron #: 100, 200, 400 or 100-100, 200-200
      - Activation function: ReLU, Tanh, Logistic

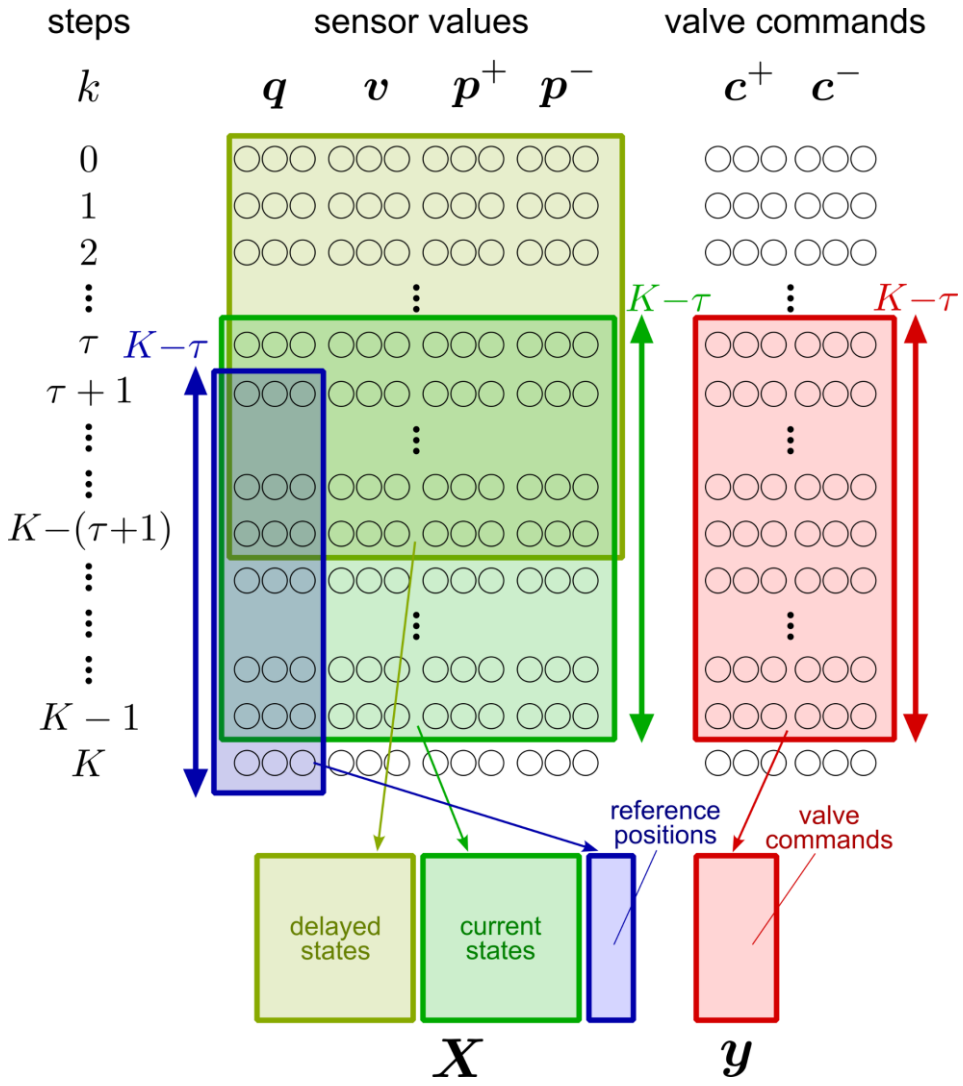
Delay embedding (Pattern 0)

- Preprocess data to make matrix for regression
  - Make a time-series data $\boldsymbol{X}$ consisting of:
    - Current states
    - Next-step positions
  - Make a corresponding output $\boldsymbol{y}$ consisting of:
    - Valve commands
  - **No delay is embedded in this pattern**



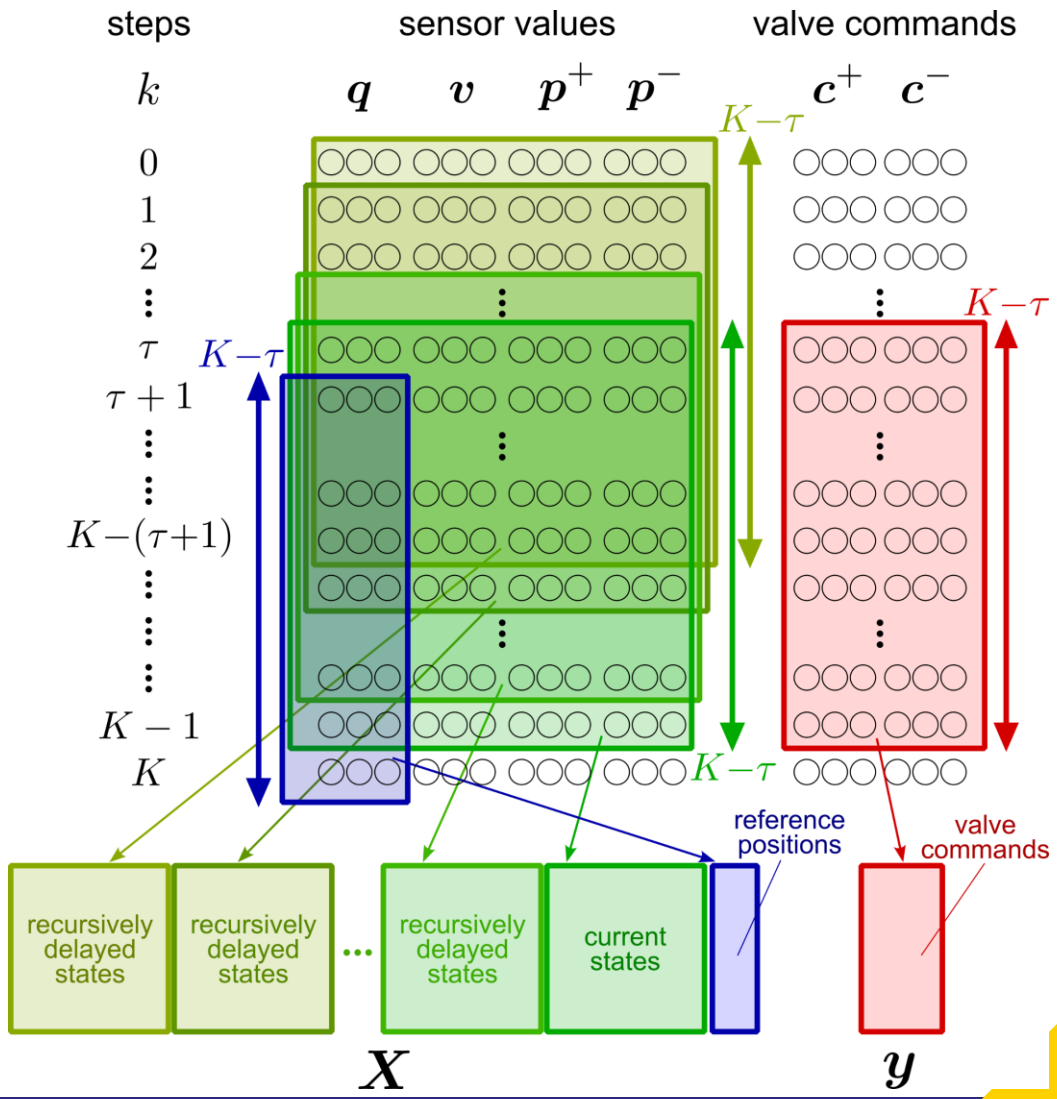
Delay embedding (Pattern 1)

- Preprocess data to compensate delay
  - Make a time-series data $\boldsymbol{X}$ consisting of:
    - τ-step delayed states
    - Current states
    - Next-step positions
  - Make a corresponding output $\boldsymbol{y}$ consisting of:
    - Valve commands



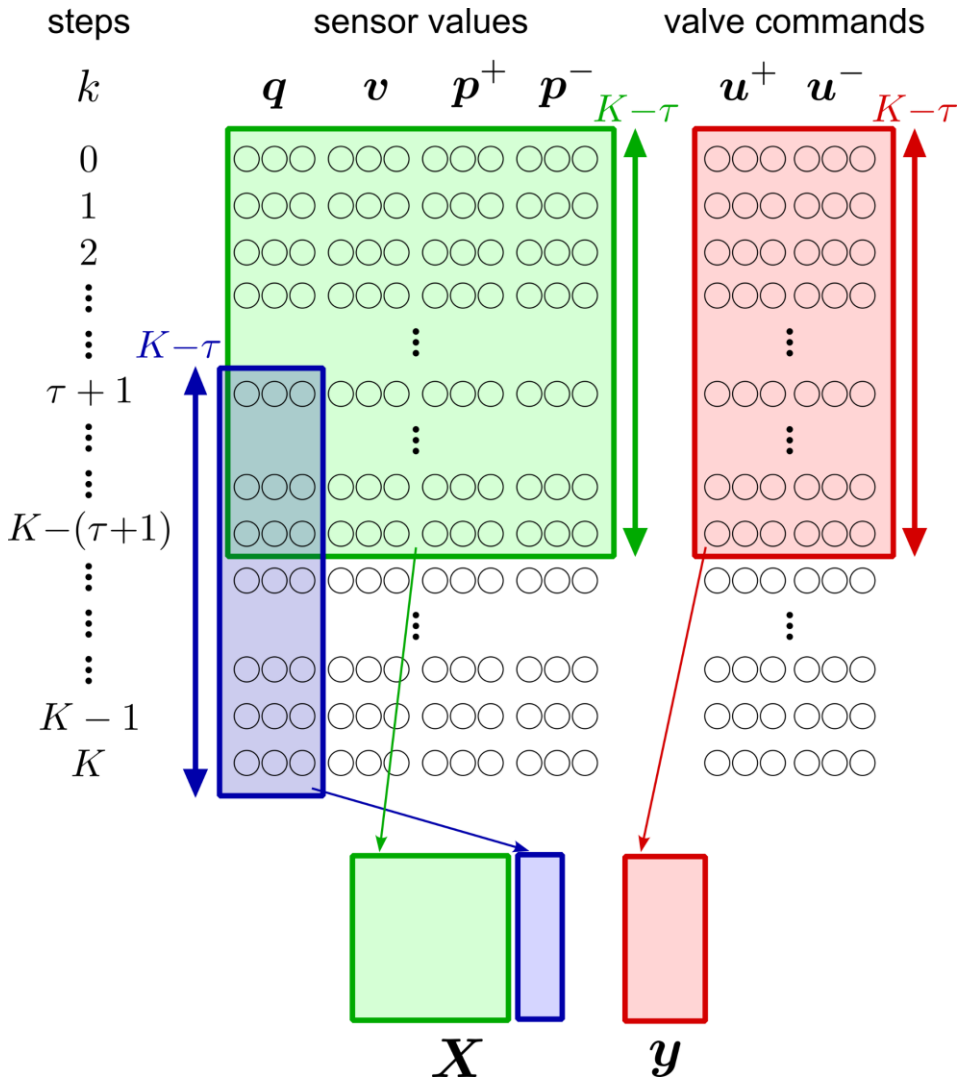
Recursive delay embedding (Pattern 2)

- Preprocess data to compensate delay
  - Make a time-series data $\boldsymbol{X}$ consisting of:
    - τ-step delayed states
    - …
    - 1-step delayed states
    - Current states
    - Next-step positions
  - Make a corresponding output $\boldsymbol{y}$ consisting of:
    - Valve commands



Reference preview embedding (Pattern 3)

- Preprocess data to compensate delay
  - Make a time-series data $\boldsymbol{X}$ consisting of:
    - Current states
    - τ-step ahead positions
  - Make a corresponding output $\boldsymbol{y}$ consisting of:
    - Valve commands



Preprocessing: Scaler

- Scaling preprocessed data before training
  - No scaler
  - MinMax scaler: Transform features by scaling each feature to [0, 1]
  - Standard scaler: Standardize feature by removing the mean and scaling to unit variance
    - $z = (x - u)/s$
  - Robust scaler: Scale features using statistics that are robust to outliers
- Compare performance among these scalers
- In total, 4 delay embedding patterns × 4 scalers = 16 preprocessing patterns

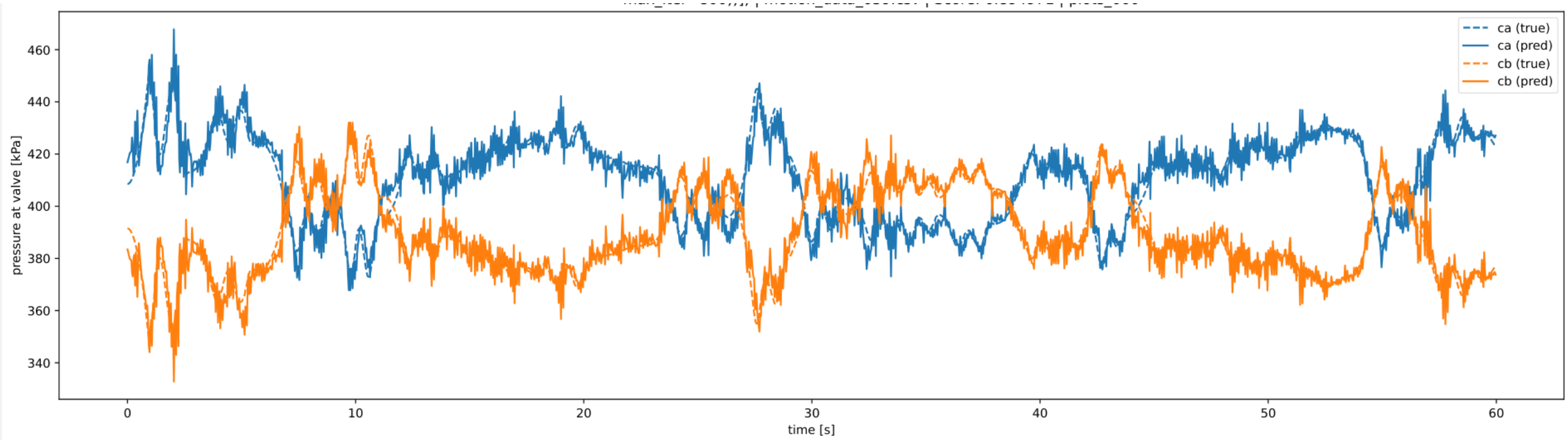
Parameter and model selection

- Selection of preview/delay steps
  - Using whole body (13DOF) for evaluation of parameters and models is time-consuming
  - Assuming delay steps of joints are same, select best delay steps using 1 DOF

- Evaluation
  - Data set: Continuous-Synchronous-All time scales
  - # of data sets for training: 60s × 72
  - # of data sets for testing: 60s × 30
  - Calculate R^2 scores for each preview/delay step

Results of valve command prediction (pattern 0)

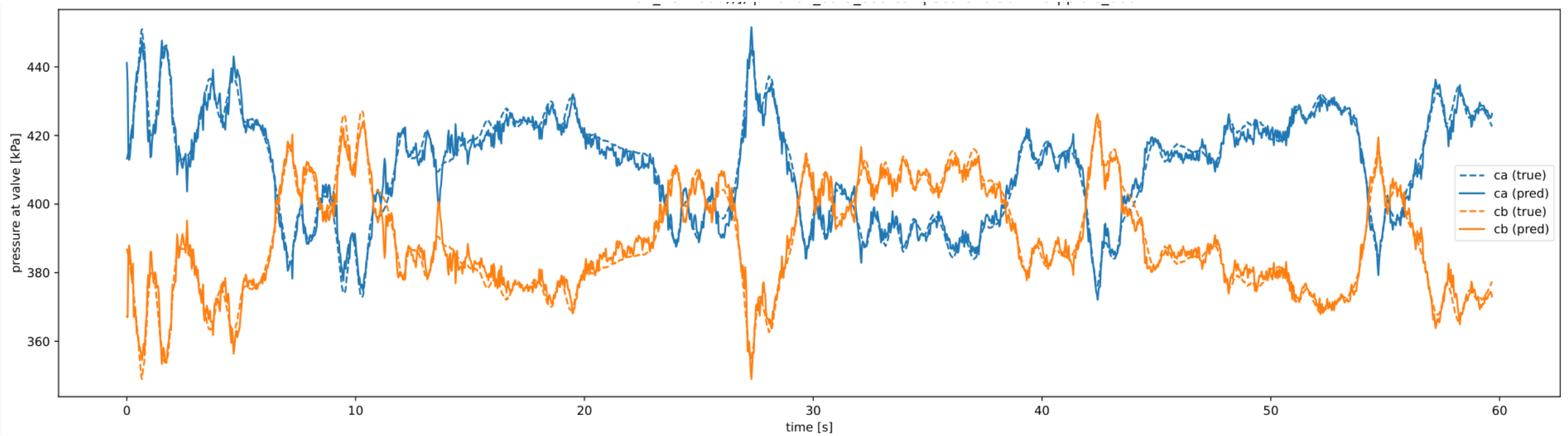
$R^2 = 0.895$



Results of valve command prediction (pattern 1)

$R^2 = 0.962$

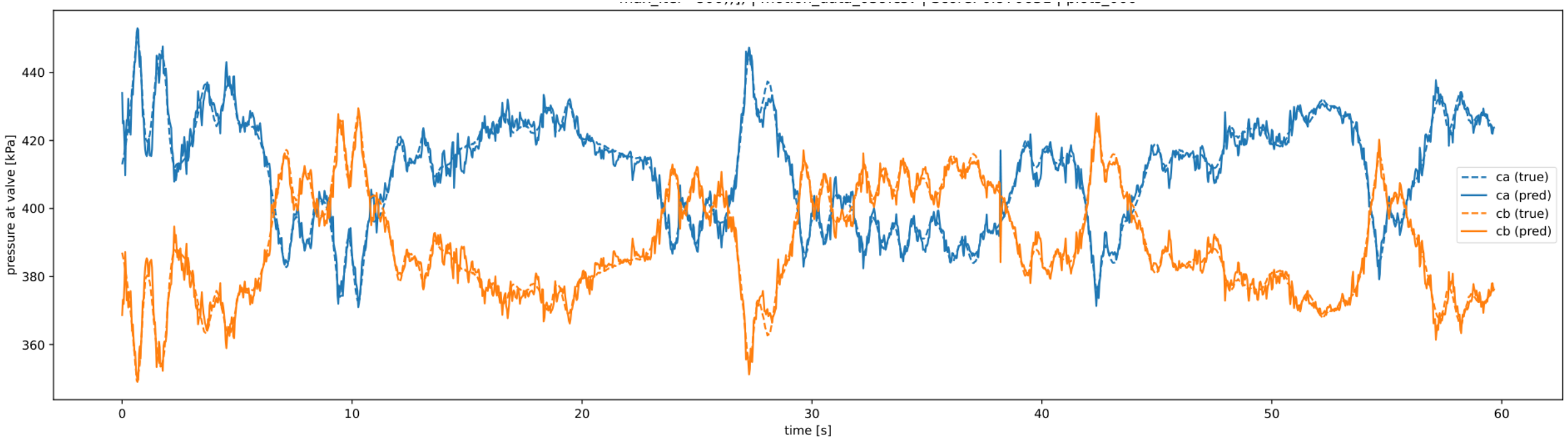
Delay step = 9



Results of valve command prediction (pattern 2)

$R^2 = 0.971$

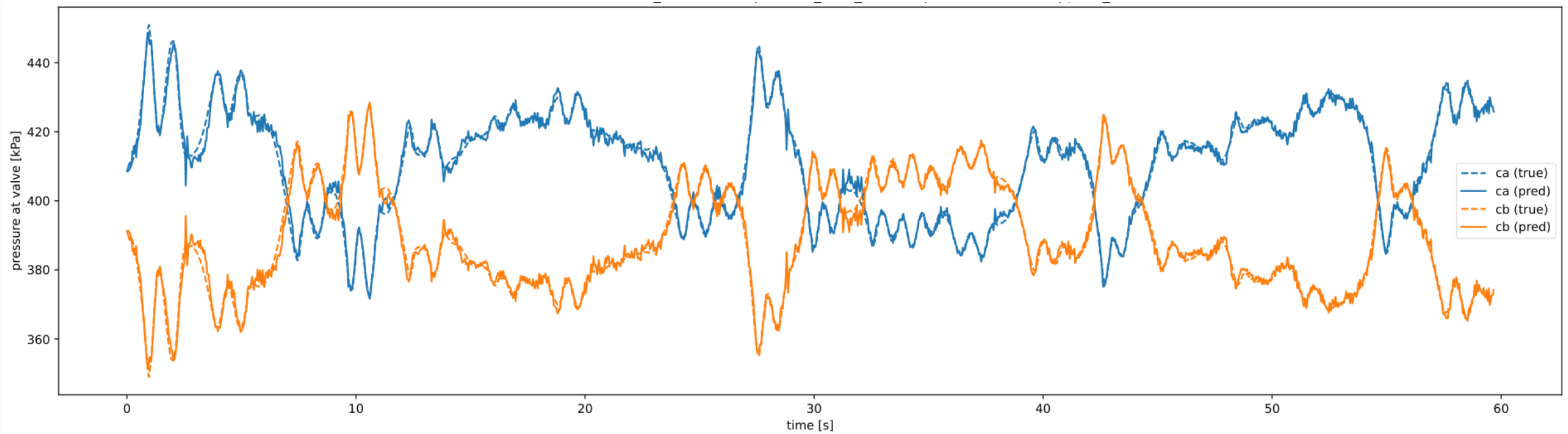
Delay step = 9



Results of valve command prediction (pattern 3)

$R^2 = 0.988$

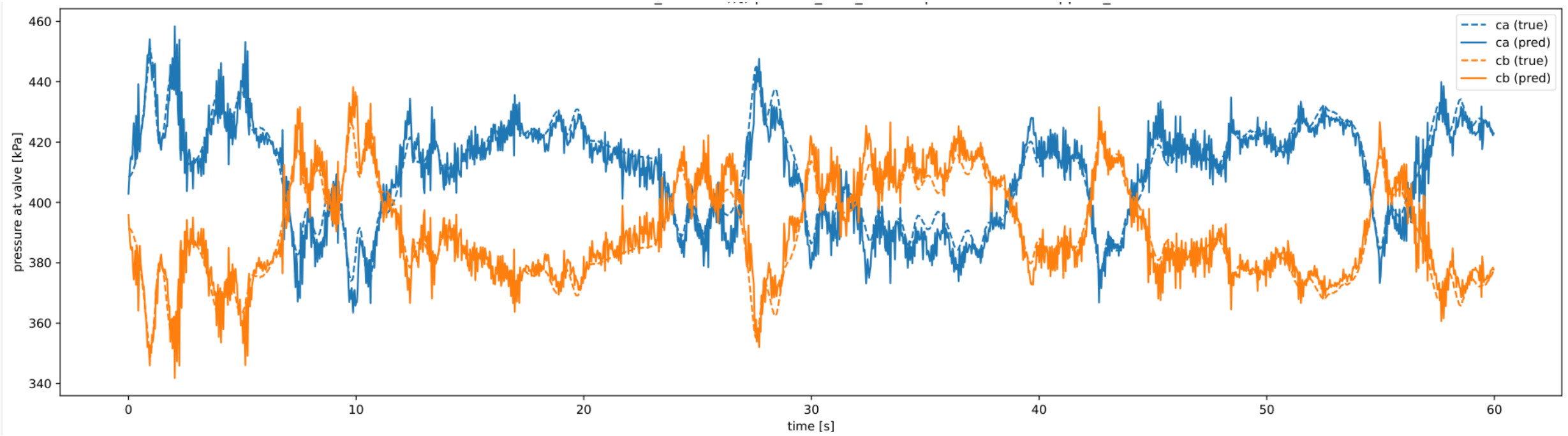
Preview step = 9



Results of valve command prediction (pattern 3)

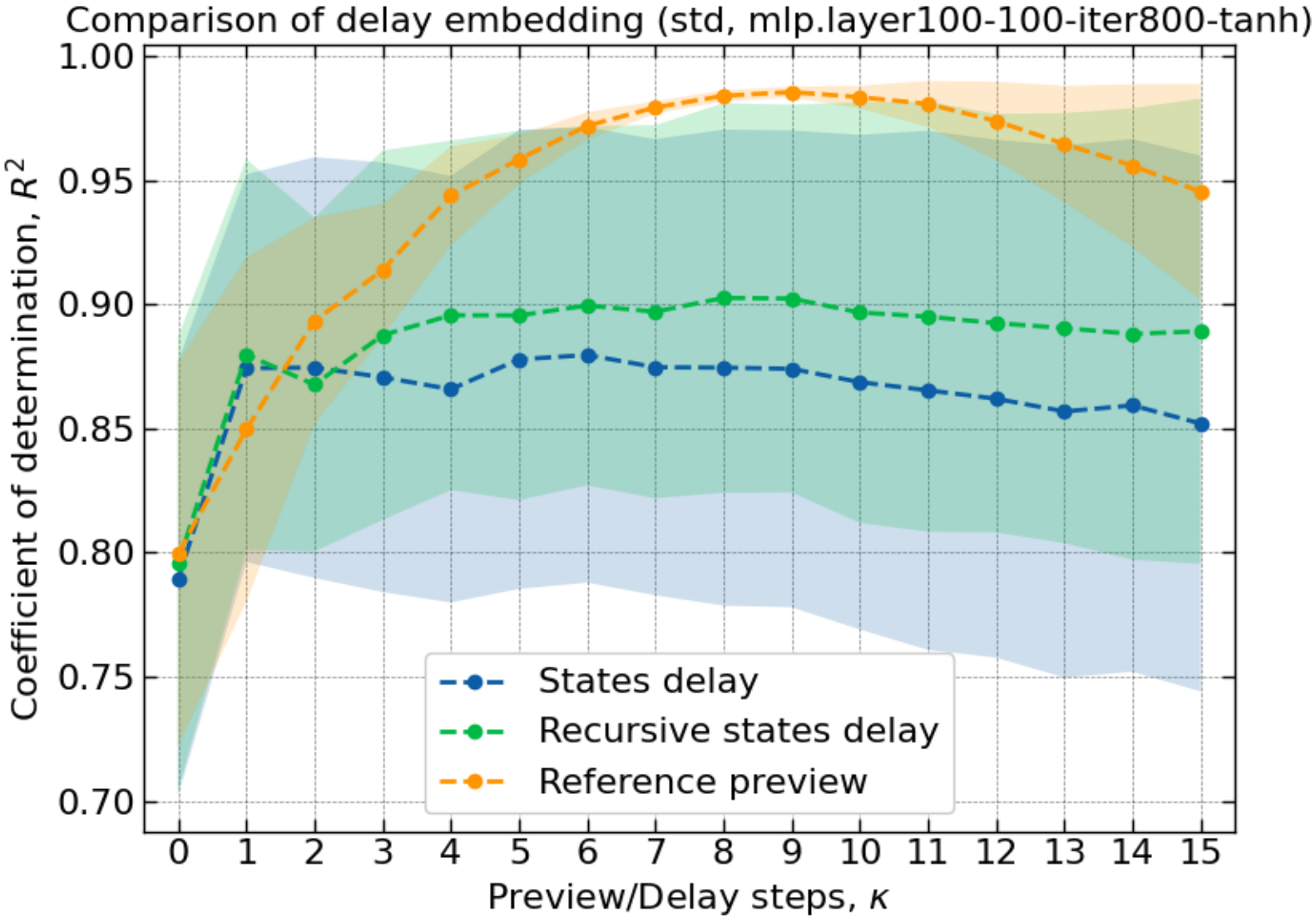
$R^2 = 0.879$

Preview step = 0



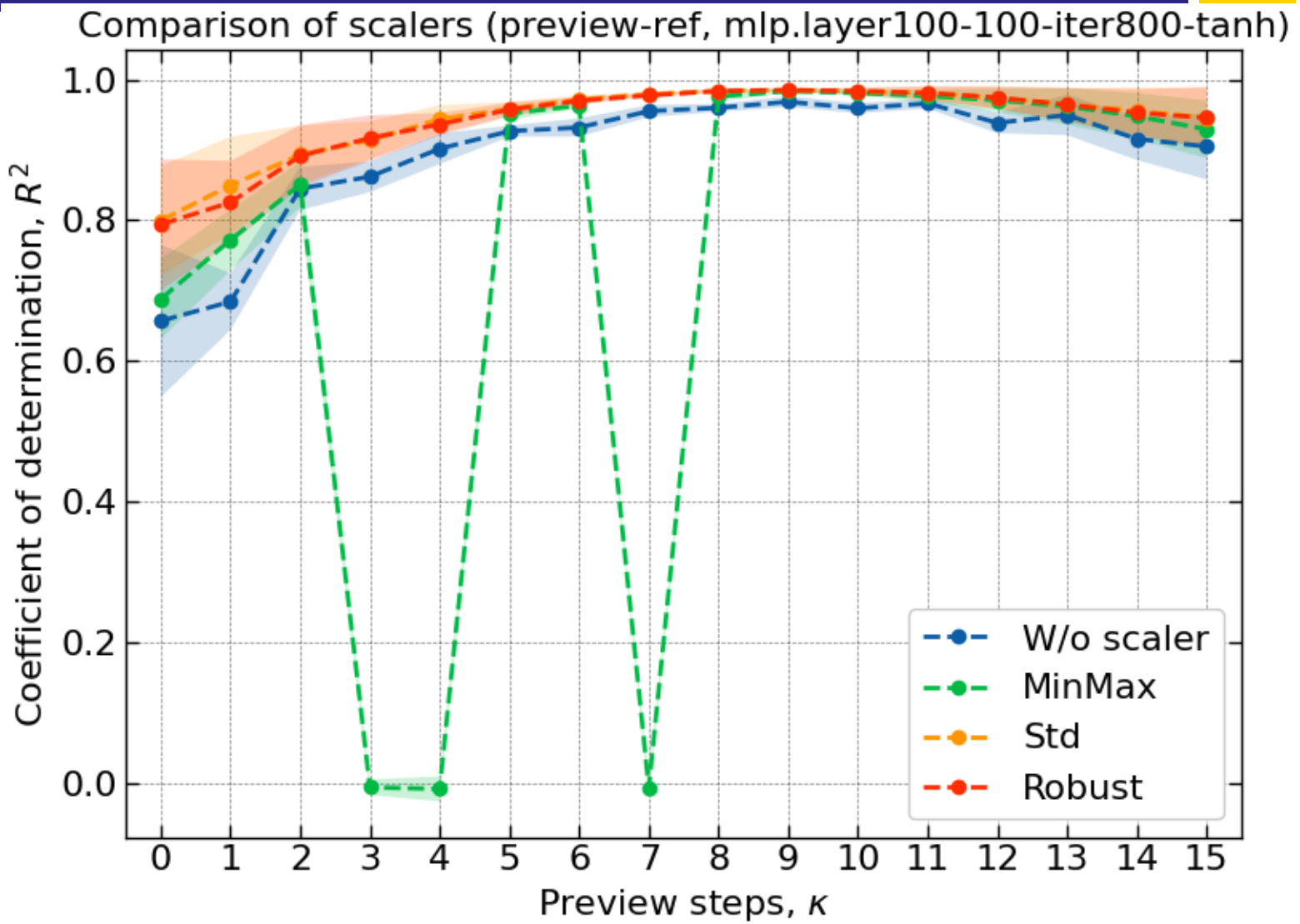
Comparison of preview/delay embedding

- Scaler: Standard
- NN model: MLP (#100-#100, tanh)
- Best delay step = 9



Comparison of scalers

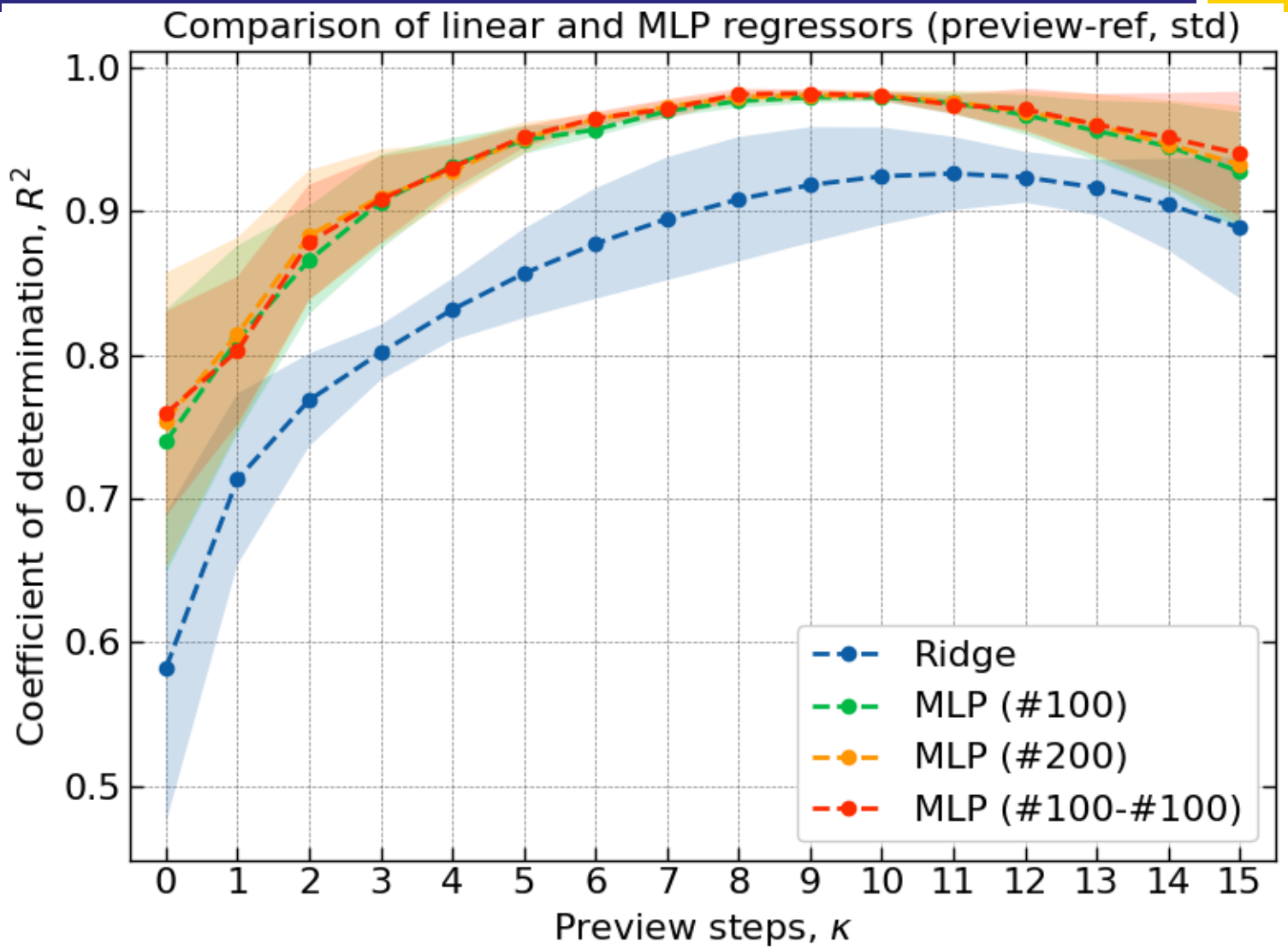
- Preview reference
- NN model: MLP (#100-#100, tanh)
- Best scaler = Std



Comparison between linear regression and MLP

- Preview reference
- MLP activation: ReLU

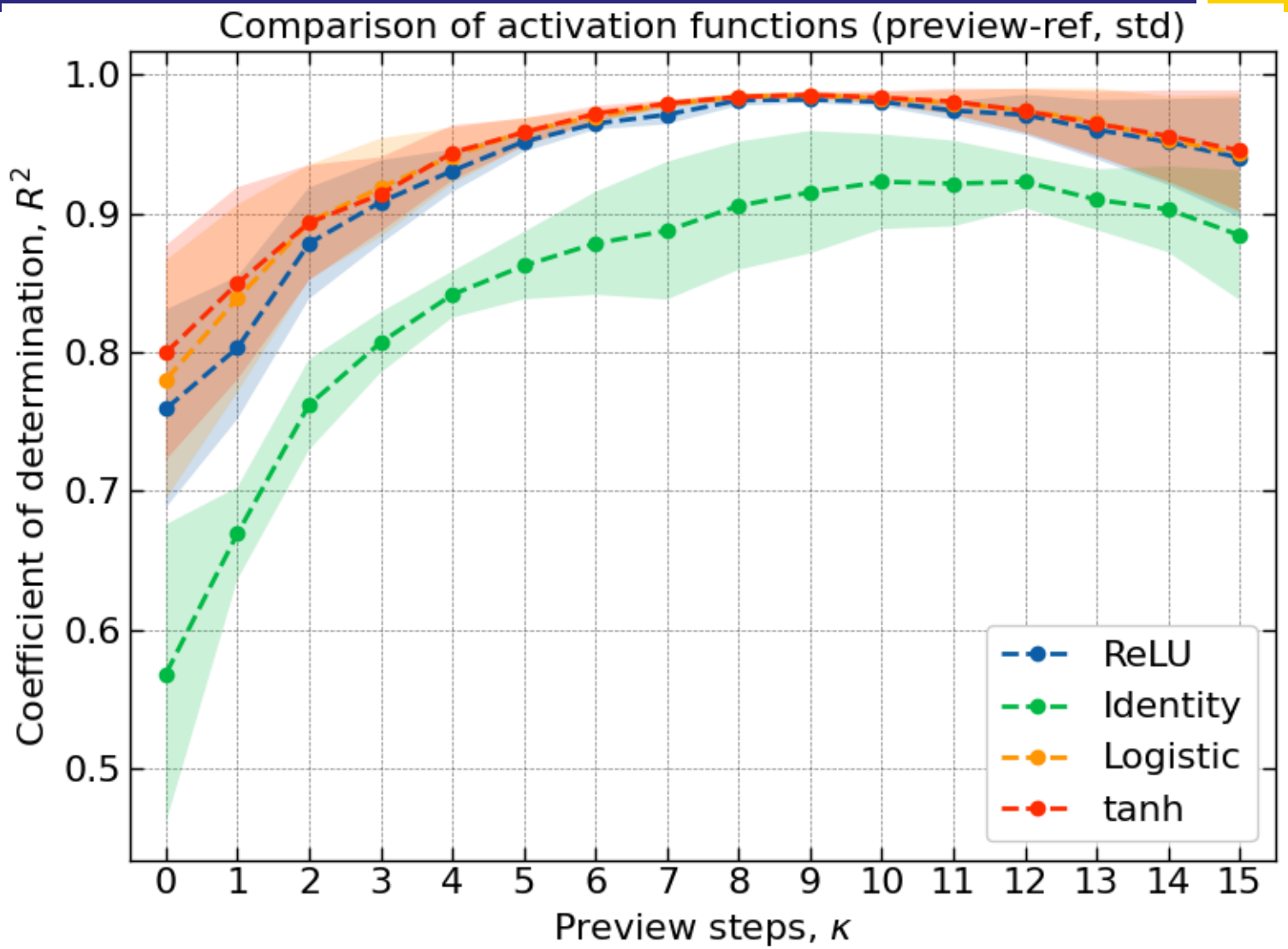
- Best regressor =
  MLP (#100-#100)



Comparison of activation functions

- Preview reference
- Scaler: Standard
- MLP (#100-#100)

- Best activation = tanh



Comparison of all regressors

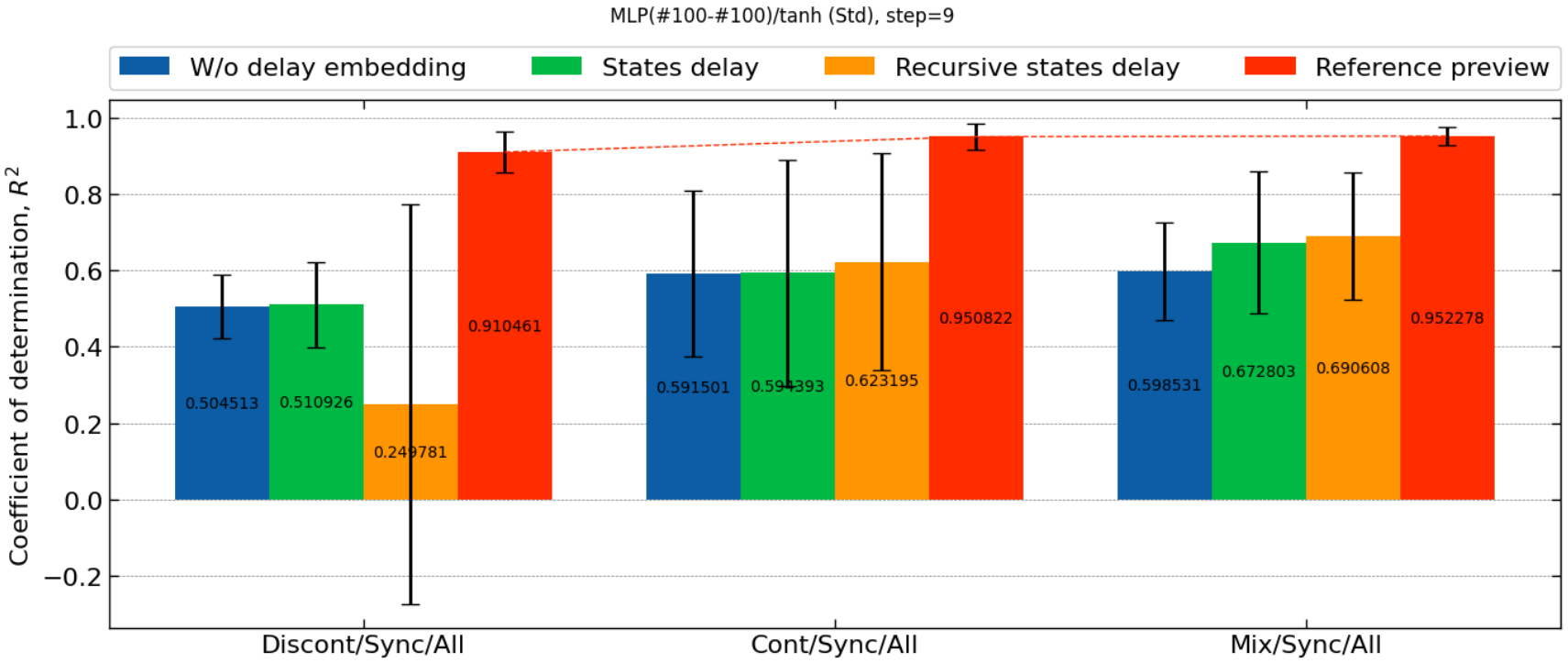
Table 1: Comparison of R^2 scores across dataset tags. Delay/Preview step: 9. Dataset tag: Cont/Sync/All.

	W/o delay embedding			States delay			Recursive states delay			Reference preview		
Regressor Models	MinMax	Std	Robust	MinMax	Std	Robust	MinMax	Std	Robust	MinMax	Std	Robust
Linear	0.58347 (0.10743)	0.58347 (0.10743)	0.58347 (0.10743)	0.78281 (0.11340)	0.78281 (0.11340)	0.78281 (0.11340)	0.83550 (0.08521)	0.83550 (0.08521)	0.83550 (0.08521)	0.91828 (0.04012)	**0.91828** (0.04012)	0.91828 (0.04012)
Ridge	0.53820 (0.11998)	0.58196 (0.10775)	0.57553 (0.10964)	0.77949 (0.11641)	0.78289 (0.11371)	0.78251 (0.11440)	0.83287 (0.08859)	0.83580 (0.08572)	0.83546 (0.08642)	0.91729 (0.04156)	0.91827 (0.04013)	**0.91828** (0.04012)
MLP(#100)/ReLU	0.66484 (0.12470)	0.74333 (0.09439)	0.73443 (0.09345)	0.80789 (0.11017)	0.86187 (0.10407)	0.85990 (0.10695)	0.84876 (0.07793)	0.89027 (0.06968)	0.88796 (0.06881)	0.94796 (0.01477)	0.97922 (0.00392)	**0.97948** (0.00346)
MLP(#200)/ReLU	0.65901 (0.11847)	0.74504 (0.09024)	0.76301 (0.10509)	0.83582 (0.10070)	0.86431 (0.10069)	0.85817 (0.10390)	0.86836 (0.07673)	0.87795 (0.06611)	0.88427 (0.07039)	0.96000 (0.00887)	0.98062 (0.00382)	**0.98063** (0.00317)
MLP(#100–#100)/ReLU	0.64084 (0.07509)	0.74530 (0.06433)	0.76519 (0.09649)	0.83669 (0.09246)	0.85964 (0.09454)	0.84424 (0.08554)	0.86725 (0.07238)	0.90291 (0.07062)	0.90273 (0.07427)	0.93778 (0.02213)	0.98202 (0.00270)	**0.98456** (0.00240)
MLP(#100–#100)/Identity	0.49596 (0.10552)	0.59199 (0.12310)	0.60244 (0.12533)	0.77750 (0.11200)	0.77348 (0.10598)	0.75937 (0.10572)	0.81451 (0.08033)	0.81143 (0.07814)	0.82851 (0.08658)	**0.91558** (0.04220)	0.91521 (0.04411)	0.91414 (0.04368)
MLP(#100–#100)/Logistic	0.71998 (0.08570)	0.78232 (0.08665)	0.78943 (0.09411)	0.86135 (0.10189)	0.87748 (0.09646)	0.86869 (0.09426)	0.88508 (0.06735)	0.90400 (0.07131)	0.90635 (0.07468)	0.98131 (0.00298)	**0.98501** (0.00182)	0.98499 (0.00221)
MLP(#100–#100)/tanh	0.67904 (0.05181)	0.79573 (0.09113)	0.78581 (0.09192)	0.85106 (0.09545)	0.87394 (0.09600)	0.87365 (0.09856)	0.80807 (0.04552)	0.90225 (0.07805)	0.89983 (0.08152)	0.98333 (0.00217)	**0.98543** (0.00218)	0.98486 (0.00220)

Comparison of data set continuity

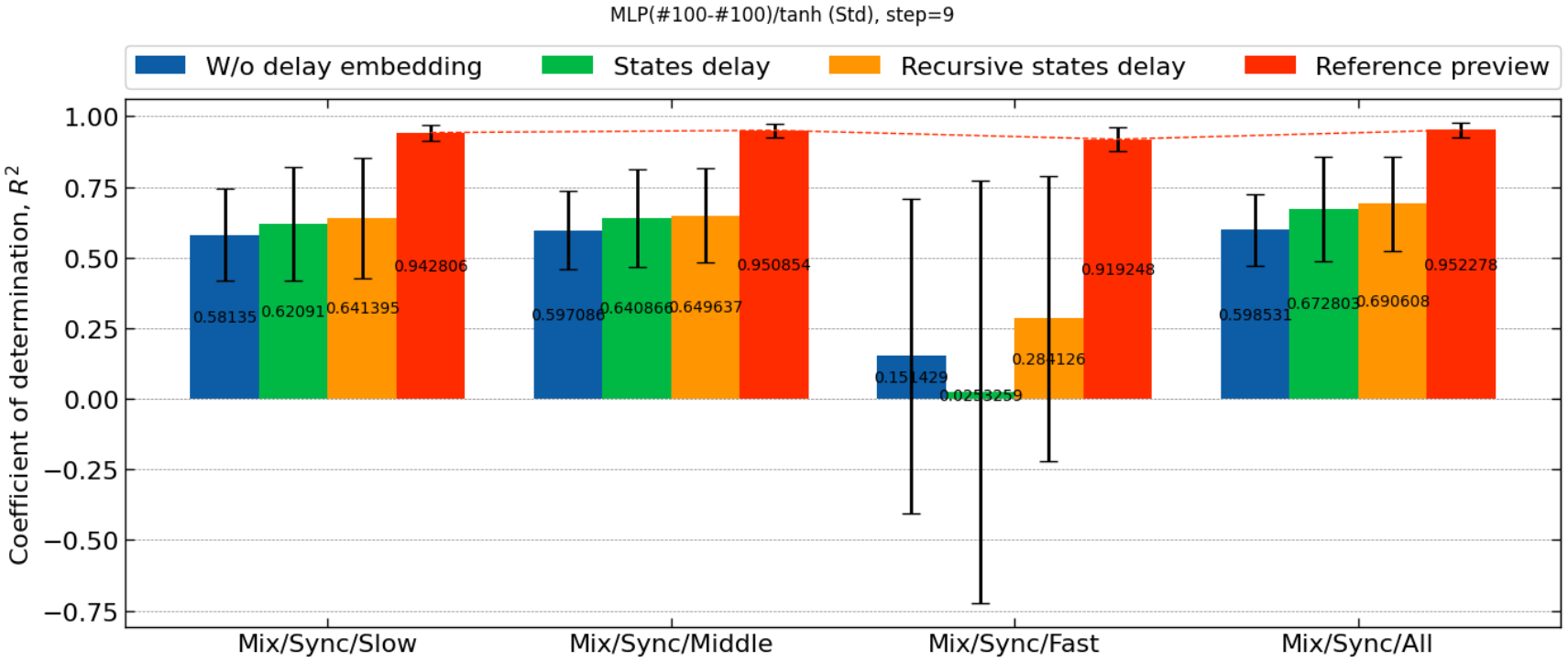
- Preview reference
- Scaler: Standard
- MLP (#100-#100,tanh)

- Best continuity = mix



Comparison of data set time scale

- Preview reference
- Scaler: Standard
- MLP (#100-#100,tanh)
- Best time scale = all



Comparison of data set properties

Table 1: Comparison of R^2 scores across dataset tags. Regressor model: MLP(#100–#100)/tanh. Delay/Preview step: 9.

	W/o delay embedding				States delay				Recursive states delay				Reference preview			
Dataset tags	W/o scaler	MinMax	Std	Robust	W/o scaler	MinMax	Std	Robust	W/o scaler	MinMax	Std	Robust	W/o scaler	MinMax	Std	Robust
Discont/Sync/Slow	0.29339 (0.20143)	0.43561 (0.09012)	0.49949 (0.08497)	0.42061 (0.15409)	0.38960 (0.07625)	-0.00892 (0.02445)	0.19660 (0.47636)	0.22154 (0.43659)	0.29864 (0.10873)	0.57830 (0.08352)	0.48150 (0.08578)	0.26954 (0.31478)	0.86932 (0.07858)	-0.00743 (0.02126)	**0.90465** (0.04737)	0.88305 (0.08155)
Discont/Sync/Middle	0.36600 (0.17684)	0.41371 (0.14045)	0.47042 (0.12872)	0.34446 (0.29162)	0.25454 (0.31356)	0.38515 (0.22834)	0.28782 (0.41435)	0.02909 (0.82486)	0.18601 (0.16612)	0.54155 (0.07833)	-0.09627 (1.00509)	-0.09287 (1.06496)	0.86321 (0.09778)	**0.89728** (0.06570)	0.89357 (0.07618)	0.88765 (0.08564)
Discont/Sync/Fast	-0.54343 (1.36764)	-0.53779 (1.31610)	-0.16800 (0.87667)	0.04447 (0.63335)	-0.08255 (0.80006)	-0.12869 (0.90266)	0.04752 (0.68776)	-0.01054 (0.70246)	-0.34556 (1.00326)	0.12417 (0.58410)	-0.22153 (1.11486)	-0.26401 (1.07691)	0.79866 (0.20406)	0.85544 (0.12848)	**0.88283** (0.09284)	0.85657 (0.12831)
Discont/Sync/All	0.15287 (0.47394)	0.42966 (0.14186)	0.50451 (0.08360)	0.38279 (0.26055)	0.41564 (0.13767)	0.50987 (0.13190)	0.51093 (0.11107)	0.27563 (0.45582)	0.21616 (0.19214)	0.56462 (0.07755)	0.24978 (0.52406)	0.45429 (0.21628)	0.87692 (0.08390)	0.90669 (0.05661)	**0.91046** (0.05474)	0.90898 (0.05832)
Cont/Sync/Slow	0.33853 (0.34293)	0.38290 (0.34339)	0.39001 (0.35297)	0.39738 (0.34728)	0.43102 (0.32929)	-0.00335 (0.01046)	0.38521 (0.38841)	0.40218 (0.37643)	-0.00379 (0.01306)	-0.00362 (0.01226)	0.46244 (0.34264)	0.42014 (0.37398)	-0.00333 (0.01156)	-0.00314 (0.00778)	**0.75816** (0.15815)	0.68924 (0.20467)
Cont/Sync/Middle	0.40739 (0.29147)	-0.00378 (0.01295)	0.50007 (0.28713)	0.49188 (0.29236)	0.49533 (0.30185)	0.58218 (0.25466)	0.59441 (0.25842)	0.57744 (0.27191)	0.47902 (0.24552)	-0.00333 (0.01026)	0.62603 (0.24570)	0.62650 (0.24752)	0.78418 (0.09735)	**0.87360** (0.08124)	0.84052 (0.10660)	0.85250 (0.09711)
Cont/Sync/Fast	-0.00278 (0.59993)	0.24560 (0.37719)	0.31285 (0.32803)	0.30282 (0.34102)	-0.02143 (0.68898)	0.31395 (0.37025)	0.35454 (0.30229)	0.29815 (0.38437)	0.03961 (0.47026)	0.53294 (0.16831)	0.51923 (0.19167)	0.53093 (0.19004)	0.89863 (0.05403)	0.92454 (0.03477)	**0.92861** (0.03347)	0.92596 (0.03415)
Cont/Sync/All	0.44653 (0.20988)	0.55009 (0.19032)	0.59150 (0.21640)	0.58260 (0.20784)	0.57723 (0.24247)	0.61366 (0.25219)	0.59439 (0.29599)	0.60995 (0.28145)	0.46094 (0.19103)	0.65393 (0.23869)	0.62319 (0.28564)	0.65191 (0.25533)	0.93205 (0.03147)	0.94606 (0.03465)	**0.95082** (0.03367)	0.94883 (0.03518)
Mix/Sync/Slow	0.37759 (0.11097)	-0.00316 (0.01016)	0.58135 (0.16410)	0.59367 (0.19281)	0.56056 (0.18281)	-0.00432 (0.01488)	0.62091 (0.20141)	0.61220 (0.20268)	0.37426 (0.14507)	-0.00408 (0.01410)	0.64140 (0.21304)	0.64298 (0.20902)	0.91884 (0.02810)	-0.00309 (0.00830)	**0.94281** (0.02894)	0.94189 (0.02766)
Mix/Sync/Middle	0.44322 (0.11972)	0.56405 (0.13448)	0.59709 (0.13862)	0.60006 (0.14264)	0.51444 (0.12605)	0.63277 (0.17569)	0.64087 (0.17383)	0.61156 (0.14479)	0.33287 (0.12006)	0.64477 (0.13695)	0.64964 (0.16739)	0.62357 (0.16013)	0.93015 (0.02172)	-0.00308 (0.00979)	**0.95085** (0.02349)	0.94780 (0.02130)
Mix/Sync/Fast	-0.04415 (0.66700)	-0.26848 (1.05077)	0.15143 (0.55727)	0.21243 (0.46634)	-0.08023 (0.74425)	0.24863 (0.43174)	0.02533 (0.74869)	0.05886 (0.70296)	-0.23603 (0.89236)	0.28112 (0.44008)	0.28413 (0.50394)	0.43929 (0.28648)	0.84360 (0.13341)	**0.92241** (0.03924)	0.91925 (0.04314)	0.92035 (0.04192)
Mix/Sync/All	0.36995 (0.15667)	0.55958 (0.11355)	0.59853 (0.12745)	0.54391 (0.10338)	0.52796 (0.13719)	0.62894 (0.16934)	0.67280 (0.18622)	0.66380 (0.17833)	0.19364 (0.19517)	0.69105 (0.15835)	0.69061 (0.16577)	0.68830 (0.16783)	0.91700 (0.02607)	0.94360 (0.02232)	**0.95228** (0.02494)	0.95158 (0.02115)

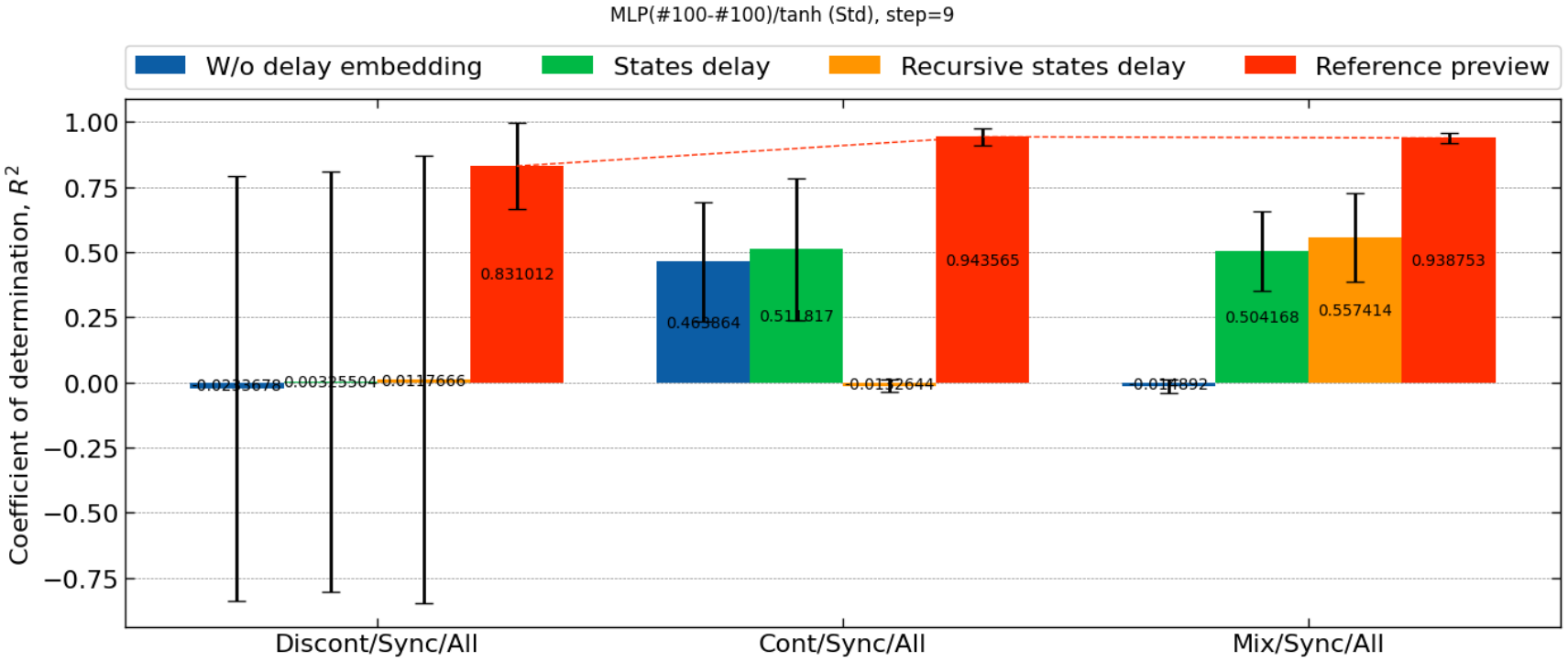
Parameter and model selection

- Selection of data sets
  - Motion data sets are generated with various properties
    - Time scale, synchronicity and continuity
  - Evaluate performance using 4 DOF

- Evaluation
  - Data set: Continuous-Synchronous-All time scales
  - # of data sets for training: 60s × 72
  - # of data sets for testing: 60s × 60
  - Calculate R^2 scores for each preview/delay step
  - Calculate average of R^2 scores among joints

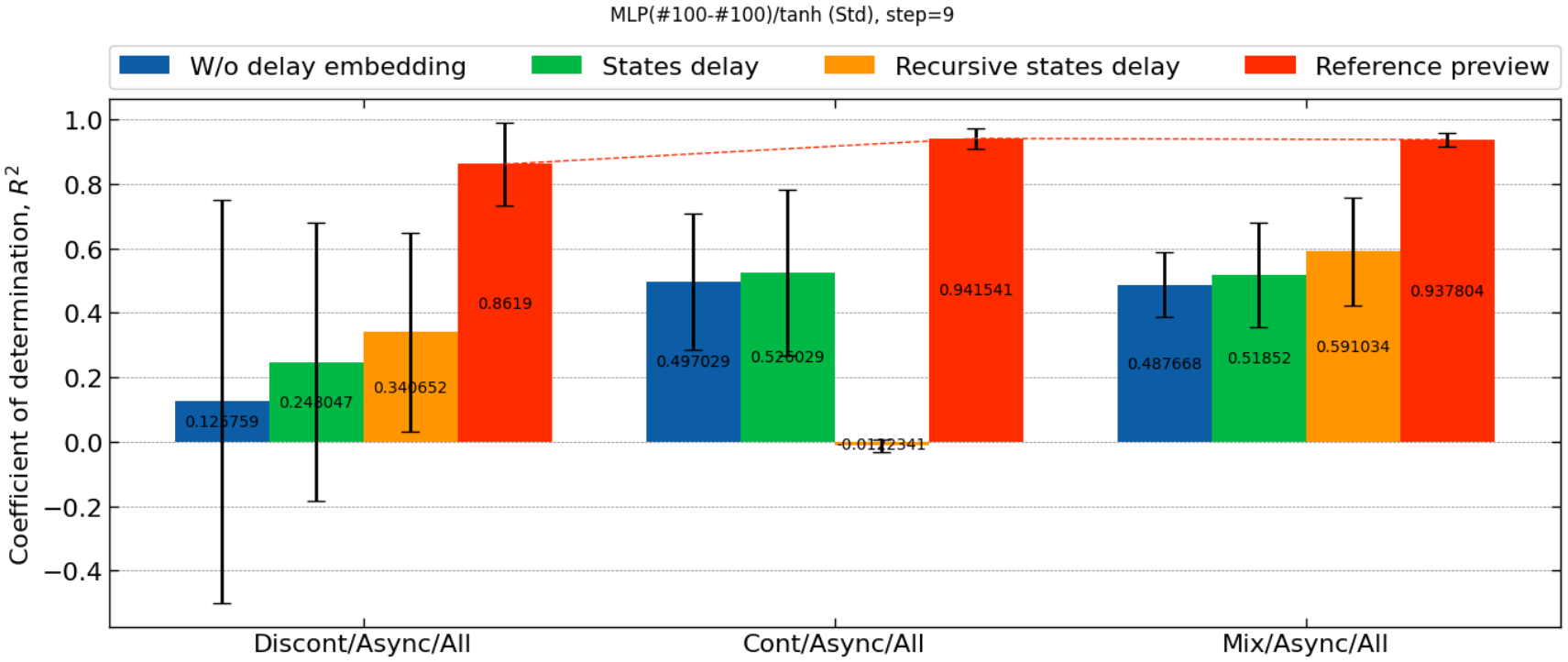
Comparison of data set continuity (sync)

- Preview reference
- Scaler: Standard
- MLP (#100-#100,tanh)



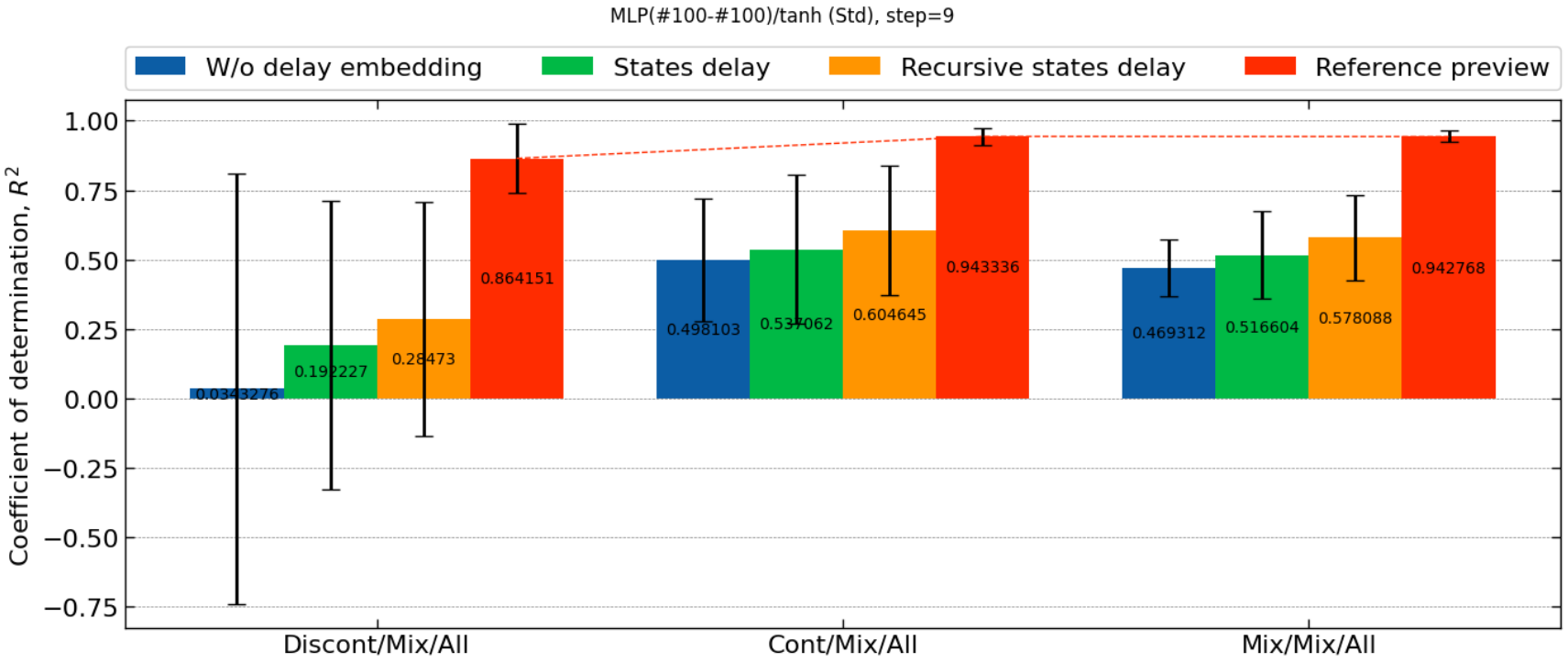
Comparison of data set continuity (async)

- Preview reference
- Scaler: Standard
- MLP (#100-#100,tanh)



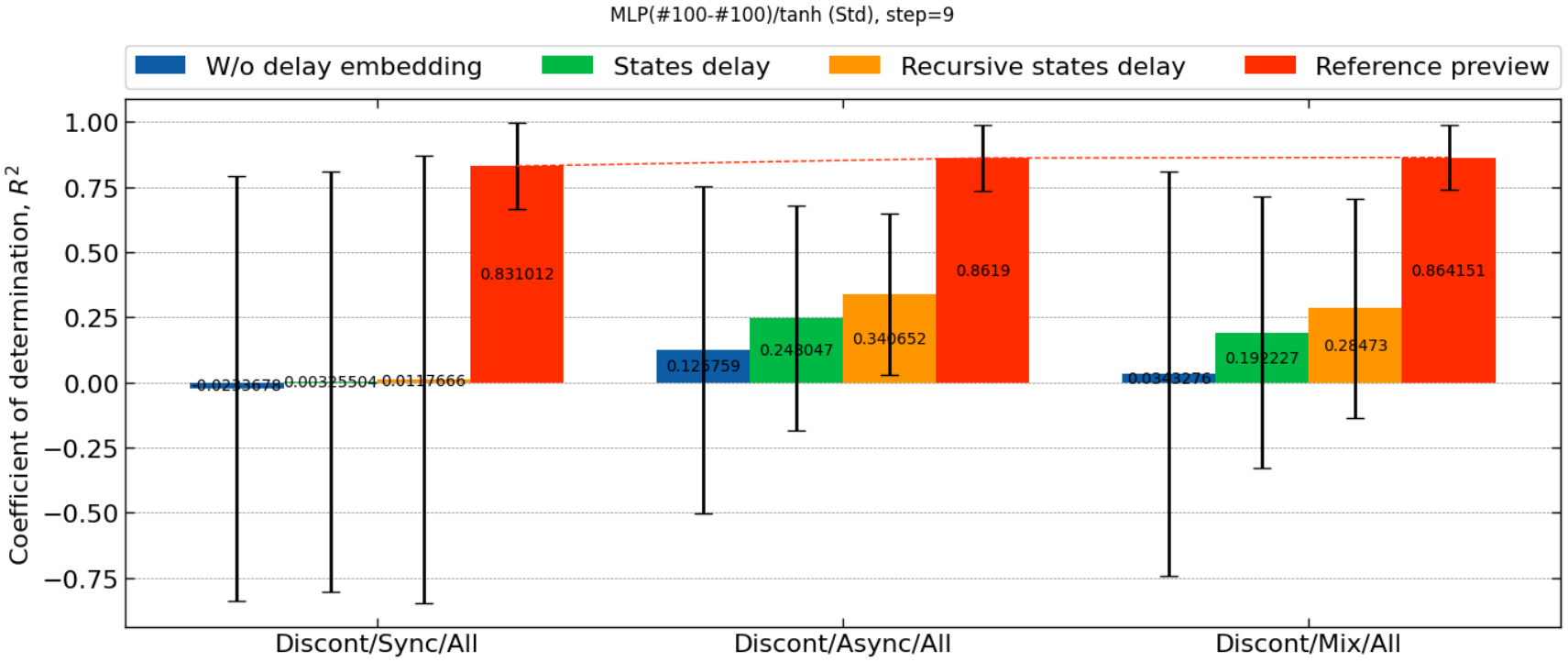
Comparison of data set continuity (mix)

- Preview reference
- Scaler: Standard
- MLP (#100-#100,tanh)



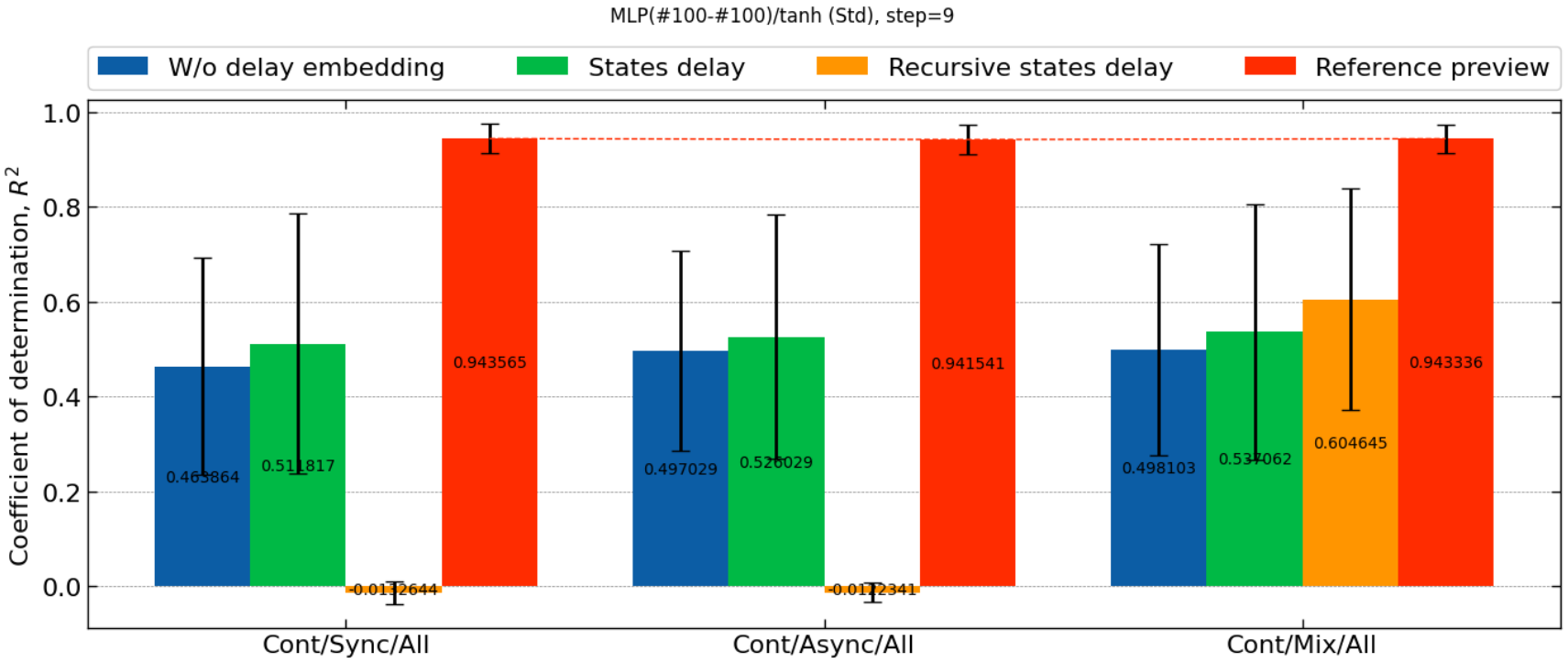
Comparison of data set synchronicity (discont.)

- Preview reference
- Scaler: Standard
- MLP (#100-#100,tanh)



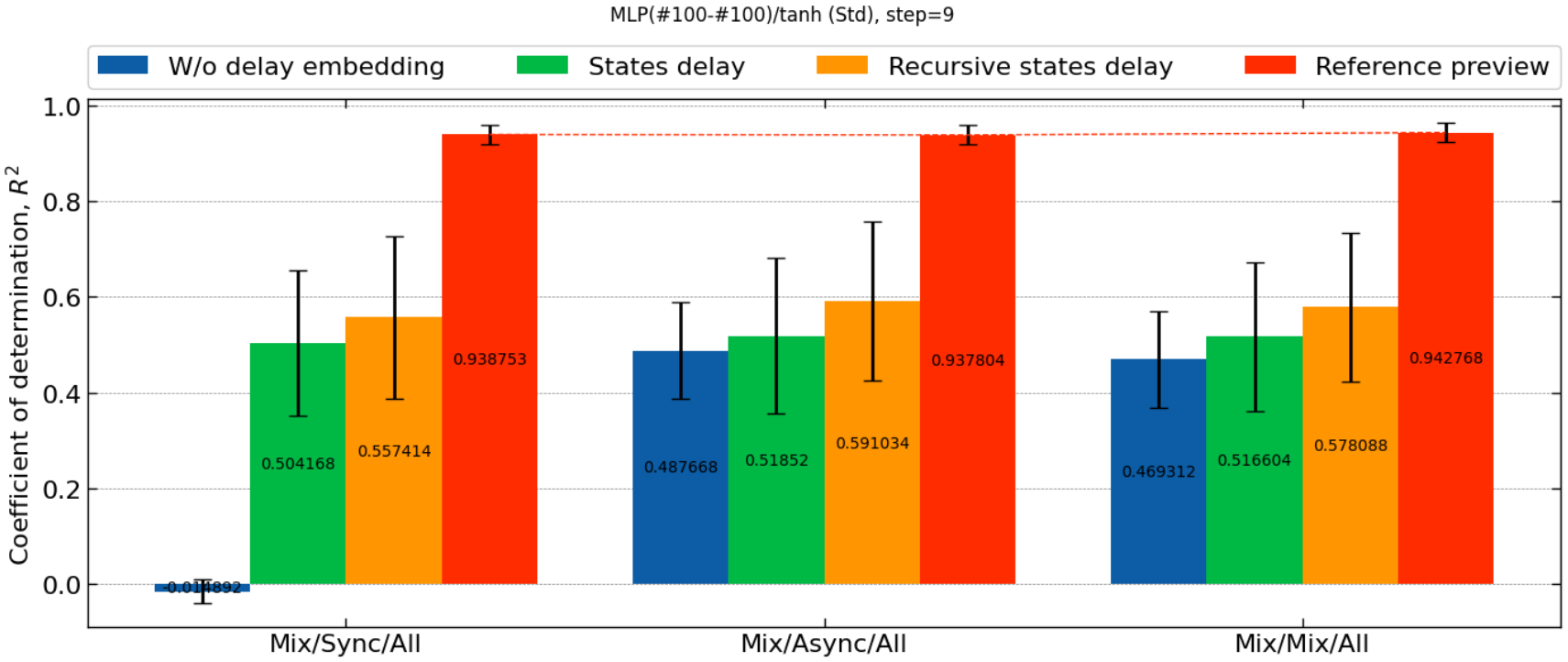
Comparison of data set synchronicity (cont.)

- Preview reference
- Scaler: Standard
- MLP (#100-#100,tanh)



Comparison of data set synchronicity (mix)

- Preview reference
- Scaler: Standard
- MLP (#100-#100,tanh)



Comparison of data set properties

Table 1: Comparison of R^2 scores across dataset tags. Regressor model: MLP(#100–#100)/tanh. Delay/Preview step: 9.

	W/o delay embedding				States delay				Recursive states delay				Reference preview			
Dataset tags	W/o scaler	MinMax	Std	Robust	W/o scaler	MinMax	Std	Robust	W/o scaler	MinMax	Std	Robust	W/o scaler	MinMax	Std	Robust
Discont/Sync/All	-0.39149 (0.97010)	0.05526 (0.74963)	-0.02337 (0.81469)	-0.04258 (0.90119)	0.01838 (0.06587)	-0.01332 (0.02127)	0.00326 (0.80874)	0.11145 (0.57742)	-0.01273 (0.02008)	-0.01228 (0.02070)	0.01177 (0.85927)	0.02923 (0.84920)	0.71512 (0.28711)	0.82566 (0.18295)	0.83101 (0.16590)	**0.84075** (0.14564)
Discont/Async/All	-0.21377 (0.60986)	0.18659 (0.51580)	0.12576 (0.62667)	0.12155 (0.57850)	-0.08597 (0.42458)	-0.01229 (0.01994)	0.24805 (0.43080)	-0.01254 (0.02125)	-0.01331 (0.02212)	-0.01421 (0.02419)	0.34065 (0.30937)	0.28384 (0.46237)	0.72455 (0.29119)	0.85107 (0.14390)	0.86190 (0.12809)	**0.86224** (0.12830)
Discont/Mix/All	-0.33467 (1.02246)	-0.08874 (0.95039)	0.03433 (0.77633)	0.01303 (0.79657)	-0.17971 (0.51315)	-0.01255 (0.02140)	0.19223 (0.52062)	0.16833 (0.55943)	-0.01245 (0.01937)	-0.01314 (0.02145)	0.28473 (0.42155)	0.09620 (0.82320)	0.50621 (0.40652)	0.83845 (0.16805)	**0.86415** (0.12466)	0.84678 (0.15579)
Cont/Sync/All	0.17824 (0.10598)	-0.01248 (0.01932)	0.46386 (0.22897)	0.45787 (0.24161)	0.01351 (0.07934)	-0.01302 (0.02103)	0.51182 (0.27363)	-0.01278 (0.02106)	-0.01251 (0.02243)	-0.01287 (0.02428)	-0.01326 (0.02396)	0.59713 (0.23426)	0.72722 (0.13610)	-0.01340 (0.02522)	**0.94356** (0.03125)	0.94131 (0.03286)
Cont/Async/All	0.09212 (0.06484)	-0.01189 (0.02062)	0.49703 (0.21034)	0.49445 (0.21664)	0.02694 (0.05033)	-0.01256 (0.02308)	0.52603 (0.25712)	0.52080 (0.26207)	-0.01248 (0.02211)	-0.01177 (0.02033)	-0.01223 (0.02030)	0.60092 (0.23368)	0.86412 (0.04284)	-0.01144 (0.02106)	**0.94154** (0.03117)	0.93945 (0.03432)
Cont/Mix/All	0.33072 (0.14350)	-0.01158 (0.01907)	0.49810 (0.22257)	0.48475 (0.22672)	0.07009 (0.06017)	-0.01210 (0.02095)	0.53706 (0.26927)	-0.01215 (0.02002)	-0.01180 (0.01938)	-0.01204 (0.02188)	0.60464 (0.23315)	-0.01230 (0.01888)	0.88028 (0.03145)	-0.01238 (0.02221)	**0.94334** (0.02992)	0.94188 (0.03090)
Mix/Sync/All	0.08777 (0.19845)	0.44081 (0.11442)	-0.01489 (0.02483)	0.45952 (0.12505)	-0.00128 (0.12313)	-0.01593 (0.02727)	0.50417 (0.15175)	0.50924 (0.15554)	-0.01670 (0.02562)	-0.01371 (0.02399)	0.55741 (0.16931)	0.55659 (0.16504)	0.84170 (0.07281)	-0.01496 (0.02378)	0.93875 (0.02110)	**0.94026** (0.02007)
Mix/Async/All	0.08603 (0.22118)	0.43068 (0.11837)	0.48767 (0.10130)	0.47113 (0.10080)	-0.01871 (0.33635)	-0.01238 (0.02019)	0.51852 (0.16294)	0.51580 (0.15439)	-0.01328 (0.02358)	-0.01174 (0.02047)	0.59103 (0.16707)	-0.01208 (0.02039)	0.80776 (0.15790)	-0.01300 (0.02016)	0.93780 (0.02033)	**0.94113** (0.02098)
Mix/Mix/All	0.06294 (0.25164)	0.45829 (0.09860)	0.46931 (0.10160)	0.46533 (0.10266)	-0.03774 (0.14276)	0.53856 (0.11809)	0.51660 (0.15611)	0.51207 (0.13998)	-0.01466 (0.02340)	-0.01779 (0.02694)	0.57809 (0.15437)	0.58954 (0.15294)	0.80668 (0.12598)	0.92748 (0.02741)	**0.94277** (0.02028)	0.94230 (0.02041)

Comparison of data set properties

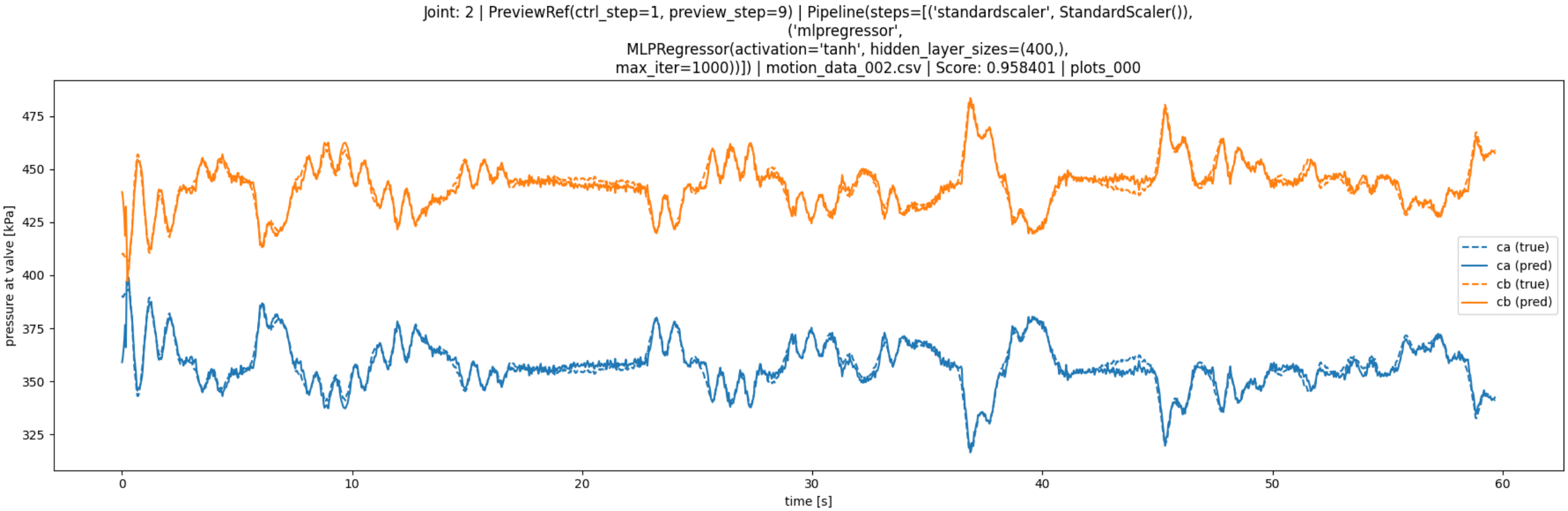
Table 1: Comparison of R^2 scores across dataset tags. Regressor model: MLP(#100–#100)/ReLU. Delay/Preview step: 9.

	W/o delay embedding				States delay				Recursive states delay				Reference preview			
Dataset tags	W/o scaler	MinMax	Std	Robust	W/o scaler	MinMax	Std	Robust	W/o scaler	MinMax	Std	Robust	W/o scaler	MinMax	Std	Robust
Discont/Sync/All	-0.05350	0.07639	0.11568	0.13020	0.41238	0.43483	0.33354	0.09598	0.42890	0.48443	0.25469	0.31944	0.81422	0.80689	**0.86174**	0.82982
	(0.91640)	(0.69057)	(0.71039)	(0.65537)	(0.21565)	(0.22050)	(0.36562)	(0.79001)	(0.15436)	(0.17650)	(0.51890)	(0.40956)	(0.20251)	(0.21834)	(0.12345)	(0.17838)
Discont/Async/All	0.05582	0.12858	0.14589	0.13166	0.48025	0.46836	0.42931	0.37839	0.44110	0.49915	0.34775	0.38179	0.86278	0.79837	0.87131	**0.87510**
	(0.71988)	(0.56158)	(0.63140)	(0.63758)	(0.08286)	(0.14558)	(0.19277)	(0.24439)	(0.13354)	(0.10897)	(0.37361)	(0.29576)	(0.13034)	(0.21222)	(0.11510)	(0.10875)
Discont/Mix/All	0.16015	0.06800	0.12212	0.13302	0.40052	0.48611	0.37857	0.24179	0.34729	0.51106	0.35654	0.30424	0.85180	0.80916	0.86082	**0.86153**
	(0.52738)	(0.70158)	(0.68792)	(0.64853)	(0.21643)	(0.12380)	(0.26961)	(0.49891)	(0.29538)	(0.12207)	(0.35213)	(0.45807)	(0.14430)	(0.21852)	(0.13548)	(0.13261)
Cont/Sync/All	0.46111	0.43930	0.49247	0.47943	0.54902	0.55798	0.53690	0.53599	0.57325	0.60896	0.59807	0.58911	0.94108	0.93816	**0.94206**	0.94152
	(0.19319)	(0.14220)	(0.19885)	(0.19862)	(0.21318)	(0.17395)	(0.23958)	(0.23267)	(0.18281)	(0.18444)	(0.22397)	(0.23249)	(0.02949)	(0.02783)	(0.03119)	(0.03049)
Cont/Async/All	0.49356	0.46022	0.51260	0.52066	0.56135	0.56366	0.55787	0.55051	0.59057	0.59938	0.60715	0.60785	0.93843	0.93818	0.94241	**0.94247**
	(0.21398)	(0.13660)	(0.20404)	(0.20058)	(0.18763)	(0.16370)	(0.23523)	(0.23285)	(0.19524)	(0.15821)	(0.22989)	(0.23027)	(0.02847)	(0.02951)	(0.03056)	(0.03171)
Cont/Mix/All	0.47768	0.43802	0.51493	0.51945	0.55237	0.57568	0.56172	0.54890	0.57237	0.59660	0.59532	0.61070	0.94124	0.93436	**0.94347**	0.94185
	(0.19025)	(0.14190)	(0.17819)	(0.17920)	(0.20267)	(0.17522)	(0.23453)	(0.23996)	(0.18204)	(0.15548)	(0.23007)	(0.23116)	(0.03009)	(0.02719)	(0.03117)	(0.03057)
Mix/Sync/All	0.40703	0.39036	0.44009	0.43812	0.54816	0.55347	0.52939	0.52247	0.59108	0.62401	0.56802	0.56714	0.92314	0.92955	**0.94064**	0.93234
	(0.13617)	(0.18611)	(0.14570)	(0.12578)	(0.13610)	(0.11698)	(0.12491)	(0.12010)	(0.13510)	(0.13632)	(0.12889)	(0.13324)	(0.02638)	(0.02654)	(0.01985)	(0.02320)
Mix/Async/All	0.46942	0.38450	0.46746	0.47105	0.52362	0.53042	0.53049	0.54864	0.42815	0.59903	0.57280	0.55391	0.93986	0.92962	**0.94303**	0.94224
	(0.13573)	(0.16889)	(0.10406)	(0.10436)	(0.11887)	(0.11147)	(0.12408)	(0.13320)	(0.14437)	(0.11849)	(0.13258)	(0.13171)	(0.02069)	(0.02686)	(0.02072)	(0.02119)
Mix/Mix/All	0.40923	0.35902	0.46608	0.47032	0.54112	0.55196	0.55000	0.53402	0.60786	0.59264	0.58401	0.56602	0.92953	0.92131	**0.94352**	0.93523
	(0.12303)	(0.21586)	(0.11812)	(0.10754)	(0.12843)	(0.11381)	(0.12744)	(0.12562)	(0.14665)	(0.11363)	(0.13397)	(0.12704)	(0.02488)	(0.03473)	(0.02069)	(0.02277)

Parameter and model selection

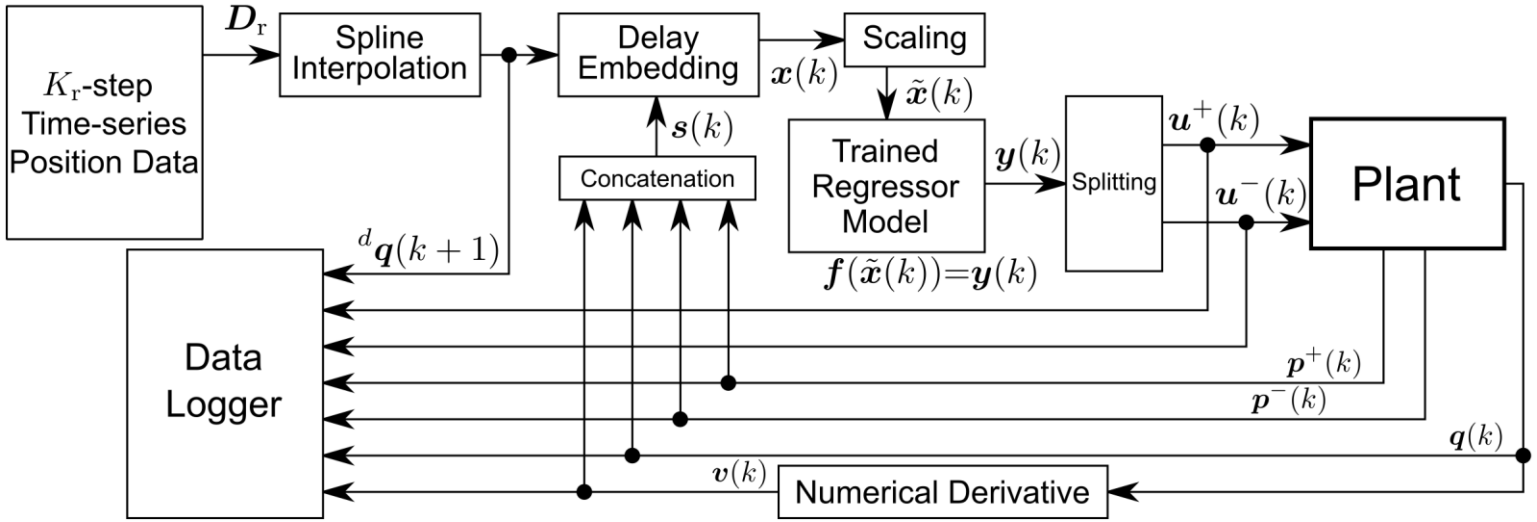
- Compute R^2 scores for whole body (13 DOF)
  - Delay embed: Preview reference (pattern 3)
  - Scaler: Standard
  - Regressor: MLP (#400, tanh)
  - Data set: Continuous-Asynchronous-All time scales
  - # of data sets for training: 60s × 180
  - # of data sets for testing: 60s × 36
  - Calculate R^2 scores for each preview/delay step
  - Calculate average of R^2 scores among joints

Prediction results for whole body learning



- Average of R^2 scores for 36 test data sets: 0.969 ± 0.008

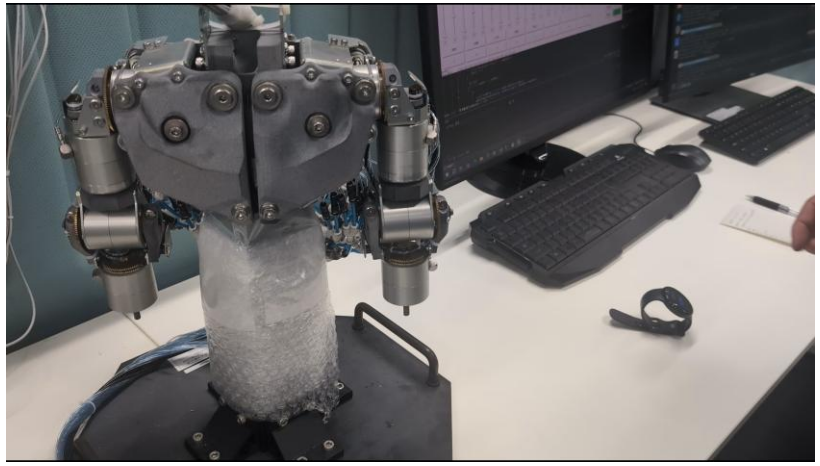
Control framework



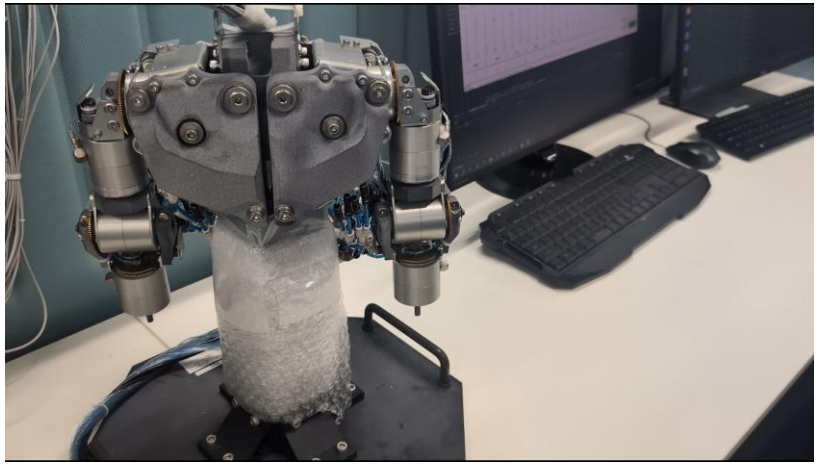
- Trajectory-tracking control with trained model
  - Recorded motion reference trajectory is interpolated using spline interpolation
  - Compare tracking performance among various patterns of components
    - Delay embedding patterns: States delay, Recursive states delay or Reference preview
    - Scaler patterns: No scaler, MinMax, Std or Robust
    - Regressor models: Linear, MLP with ReLU or MLP with tanh

Recording and re-playing (left arm)

Record

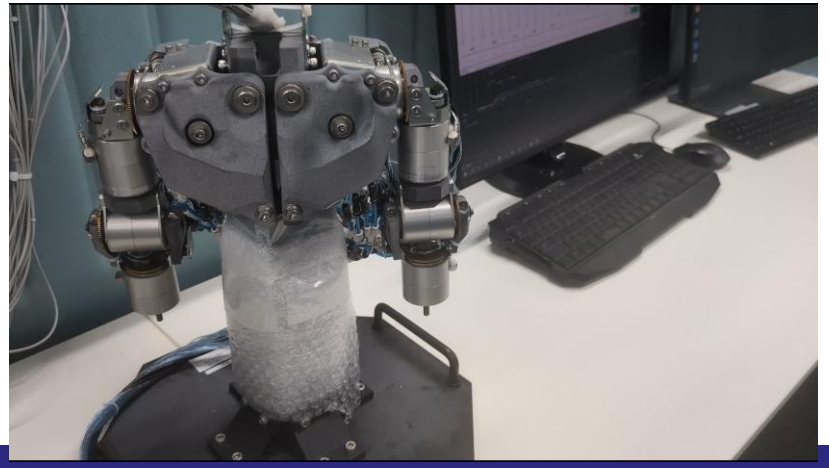


PID



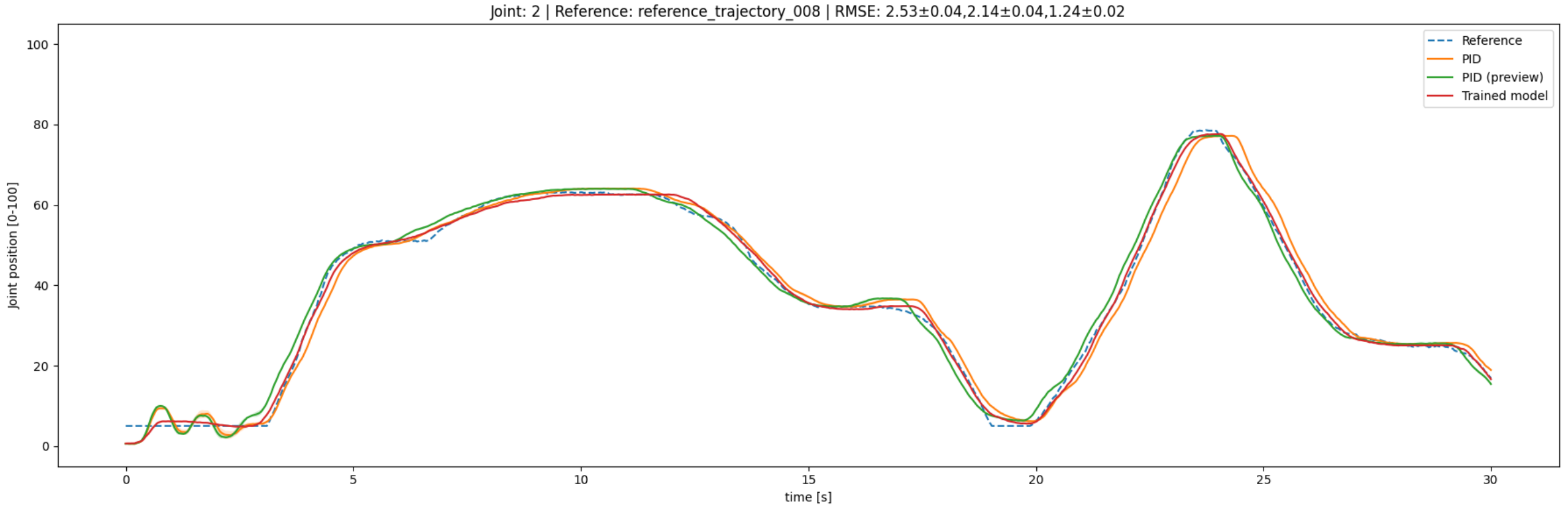
PID (preview)

Trained model



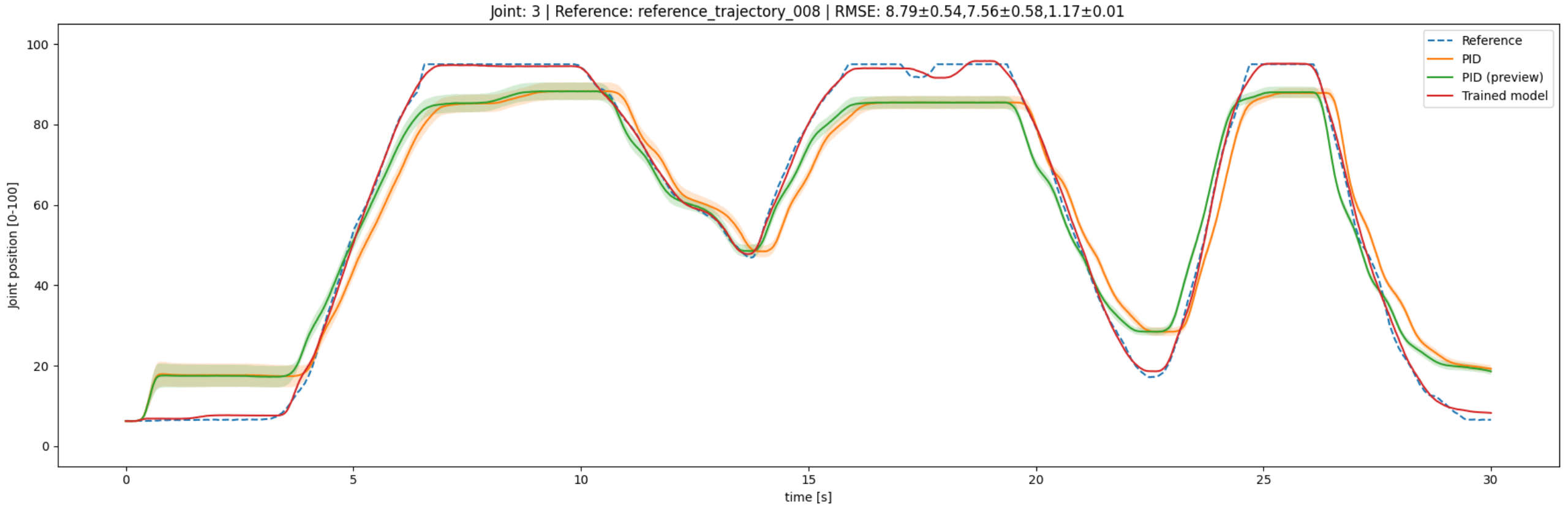
Tracking results for left arm

- Result of reference trajectory #8
  - Trained NN: MLP (#200,logistic), Standard scaler, Cont-async-all



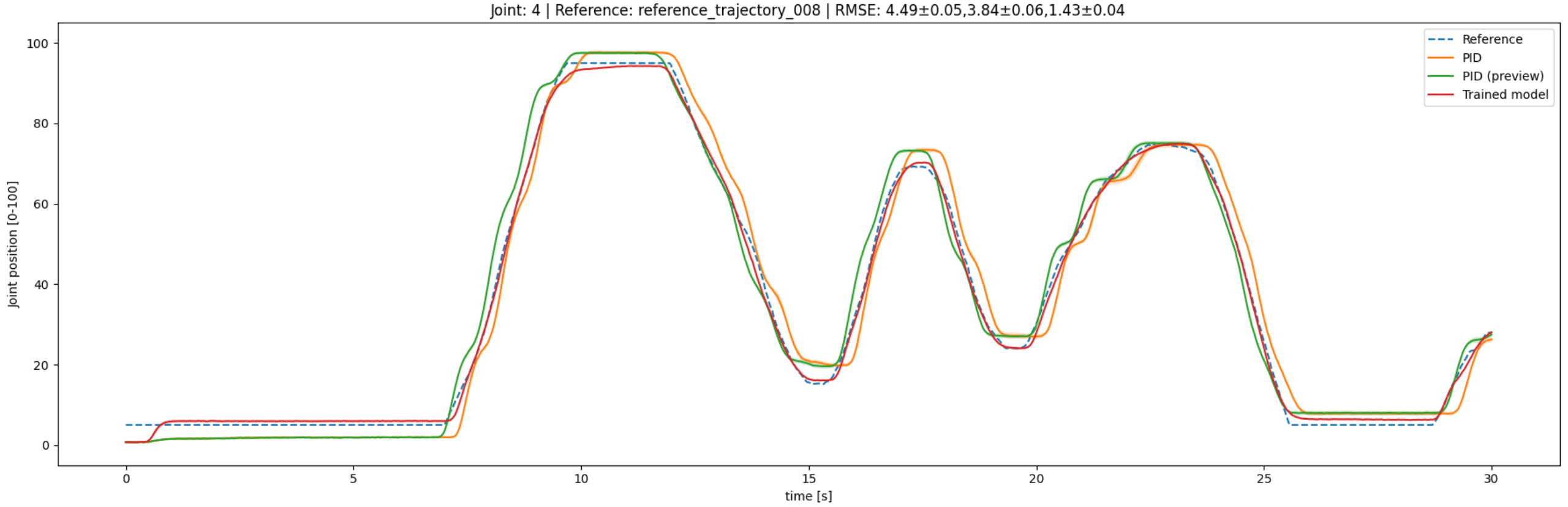
Tracking results for left arm

- Result of reference trajectory #8
  - Trained NN: MLP (#200,logistic), Standard scaler, Cont-async-all



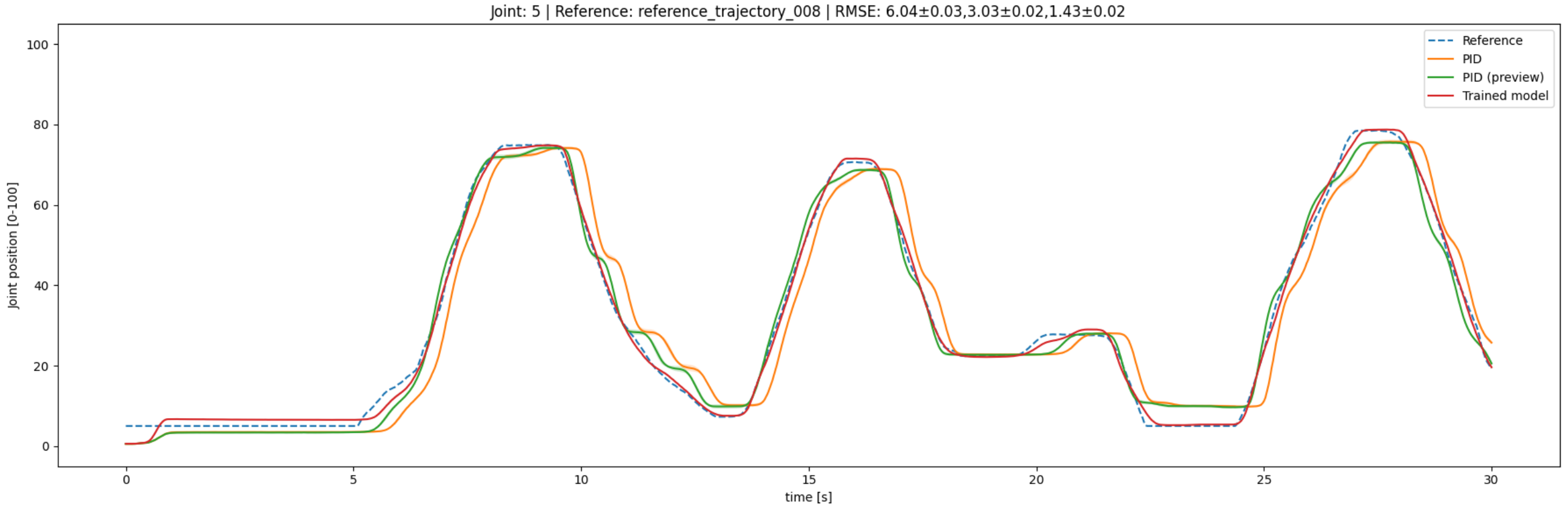
Tracking results for left arm

- Result of reference trajectory #8
  - Trained NN: MLP (#200,logistic), Standard scaler, Cont-async-all



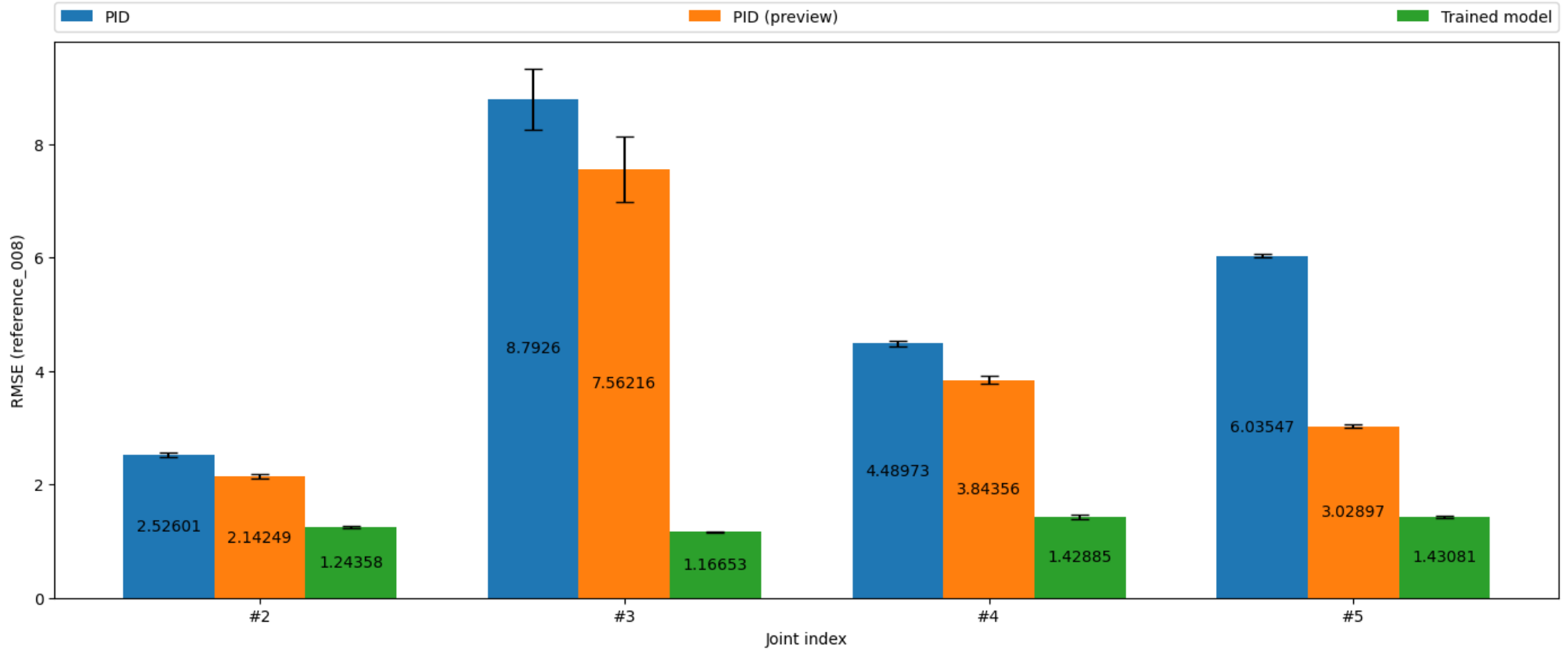
Tracking results for left arm

- Result of reference trajectory #8
  - Trained NN: MLP (#200,logistic), Standard scaler, Cont-async-all



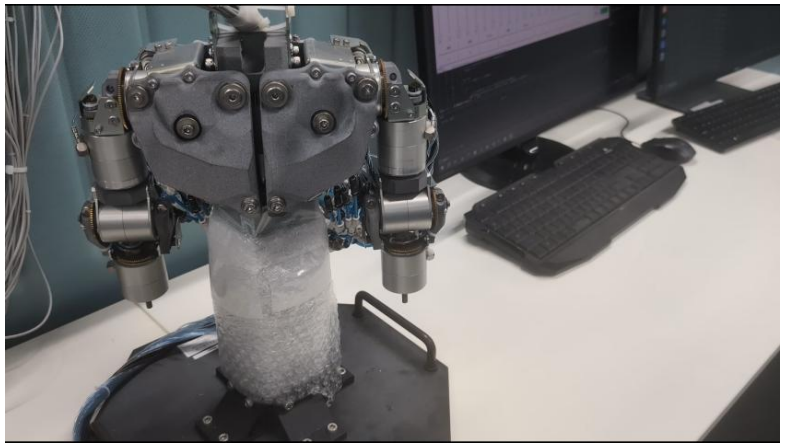
Comparison of RMSE

- Result of reference trajectory #8
  - Trained NN: MLP (#200,logistic), Standard scaler, Cont-async-all

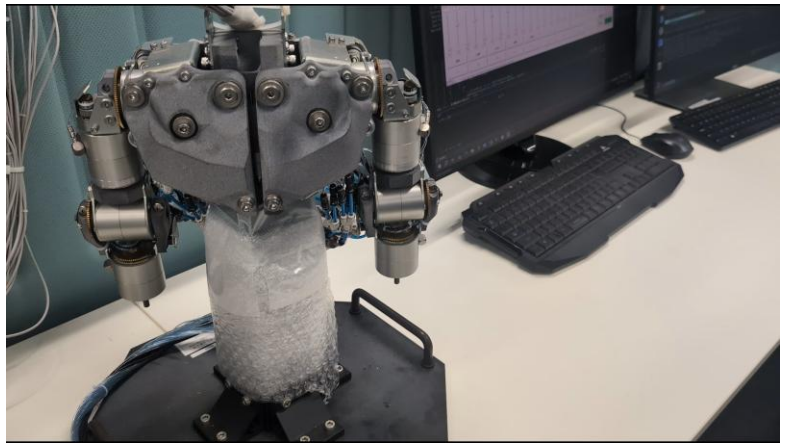


Reactions against external disturbances

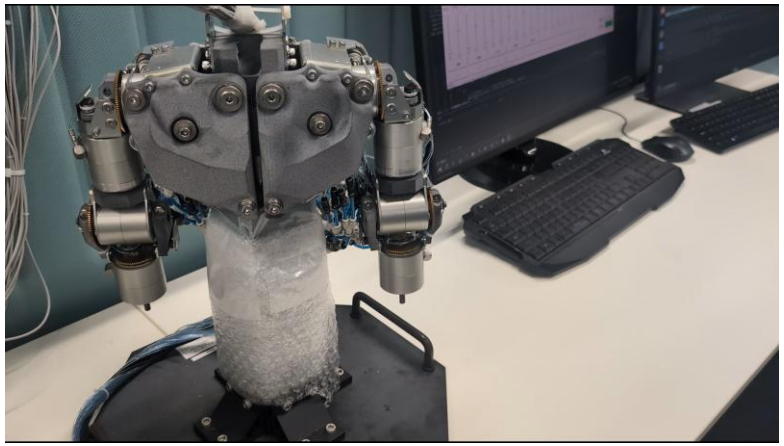
No disturbance
(trained model)



PID



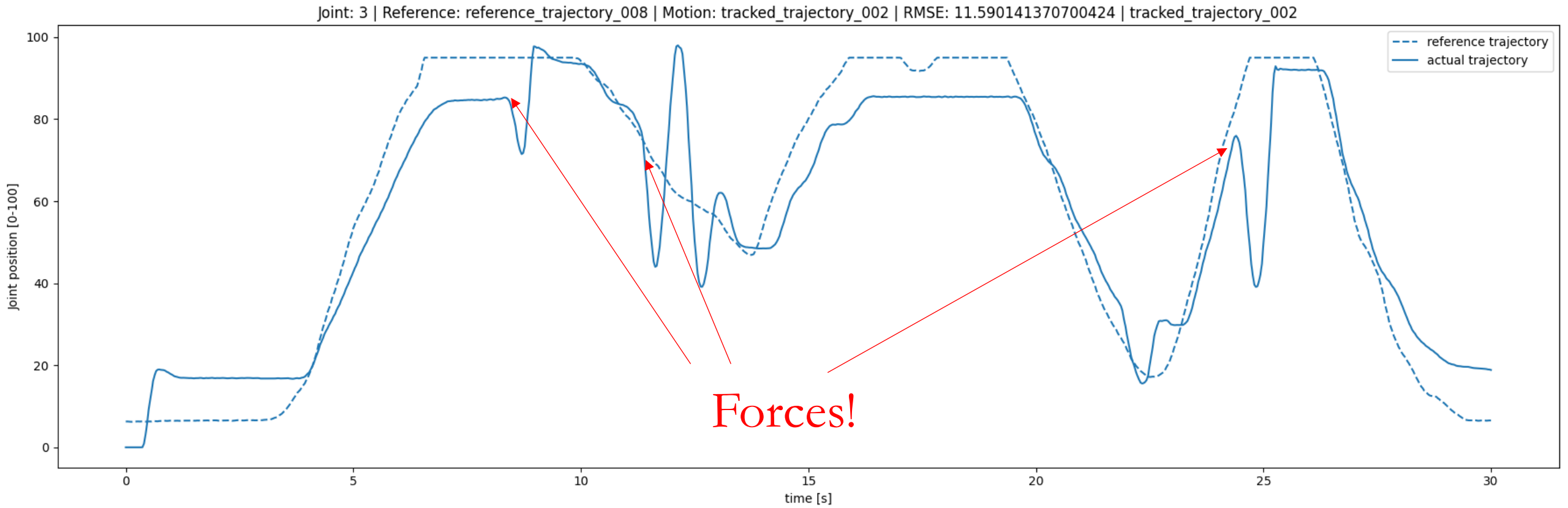
Trained model



- Supply external disturbances during motion at random

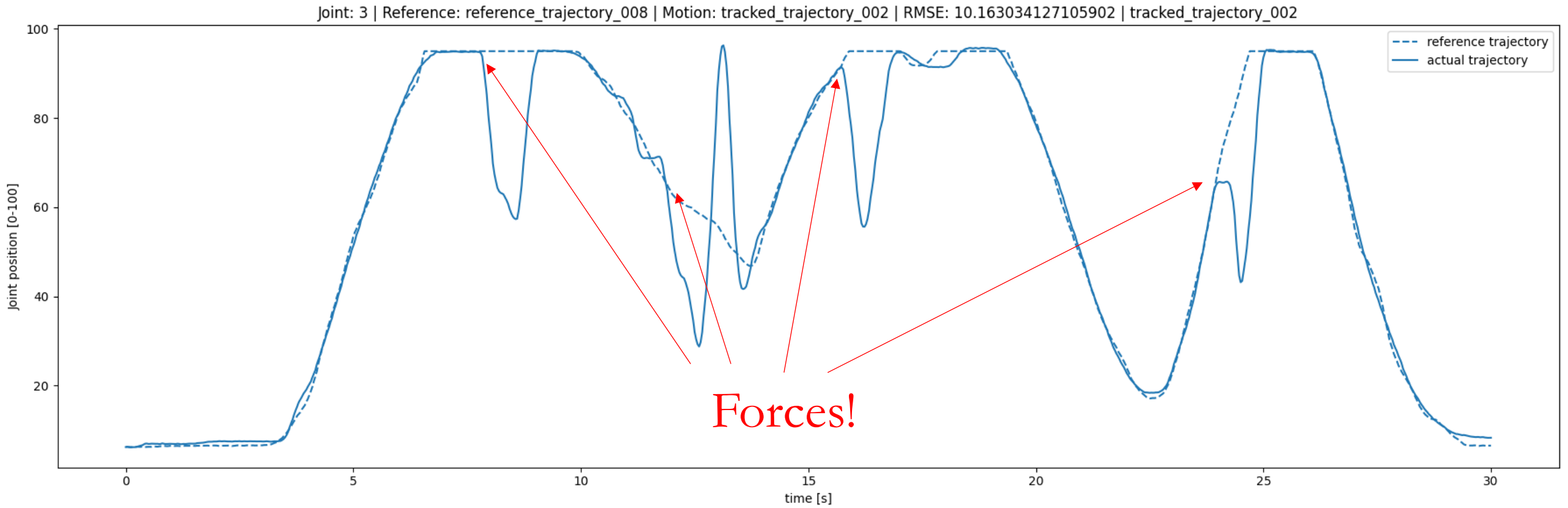
Reactions against external disturbances

- PID



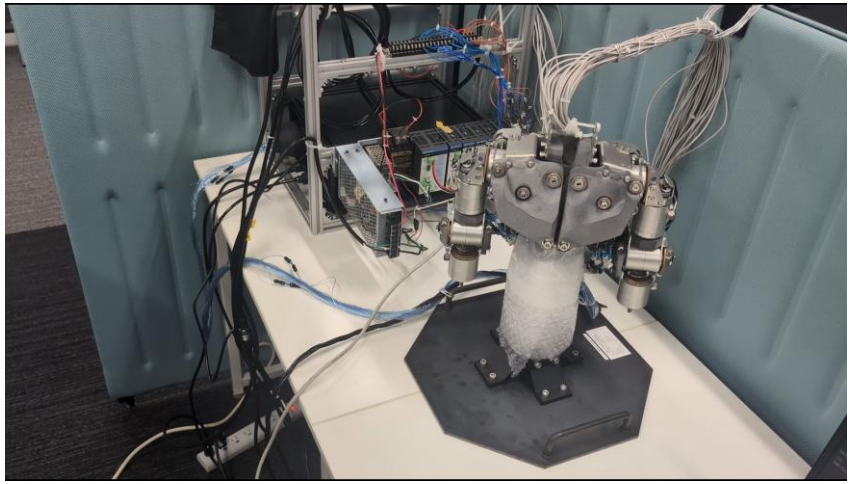
Reactions against external disturbances

- Trained model

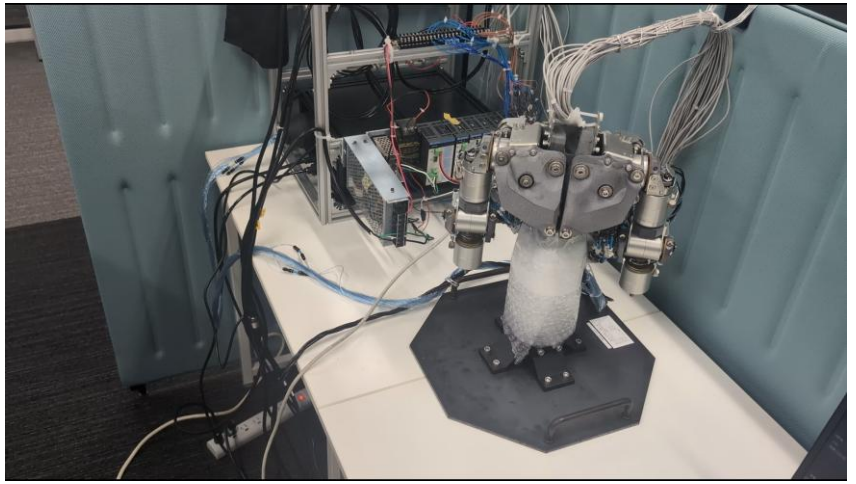


Recording and re-playing (whole body)

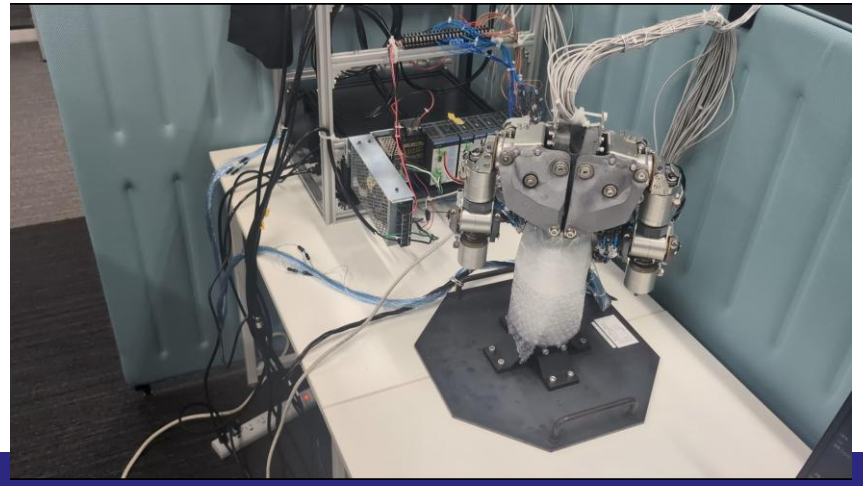
Record



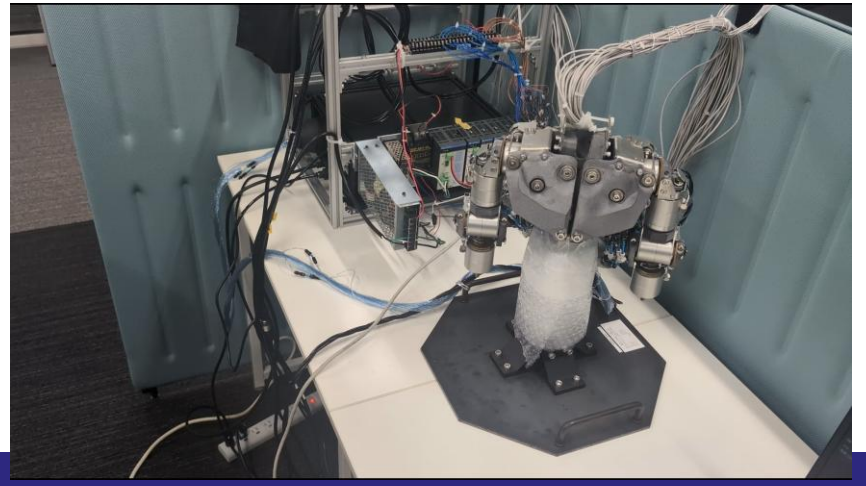
PID



PID (preview)

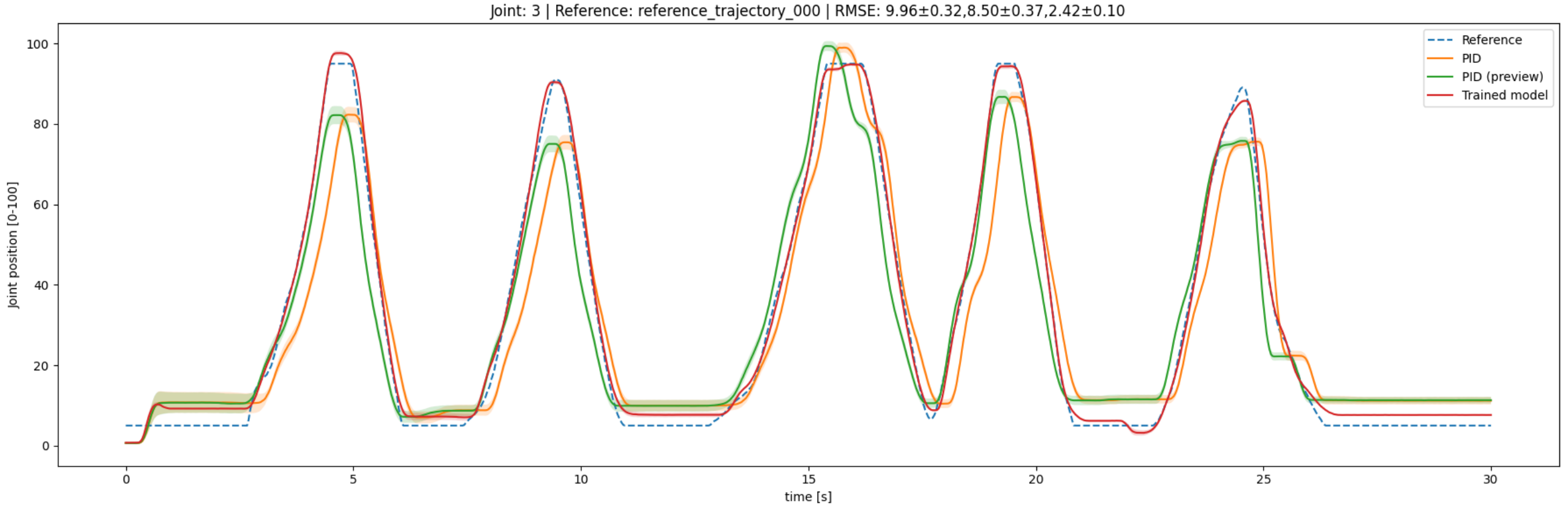


Trained model



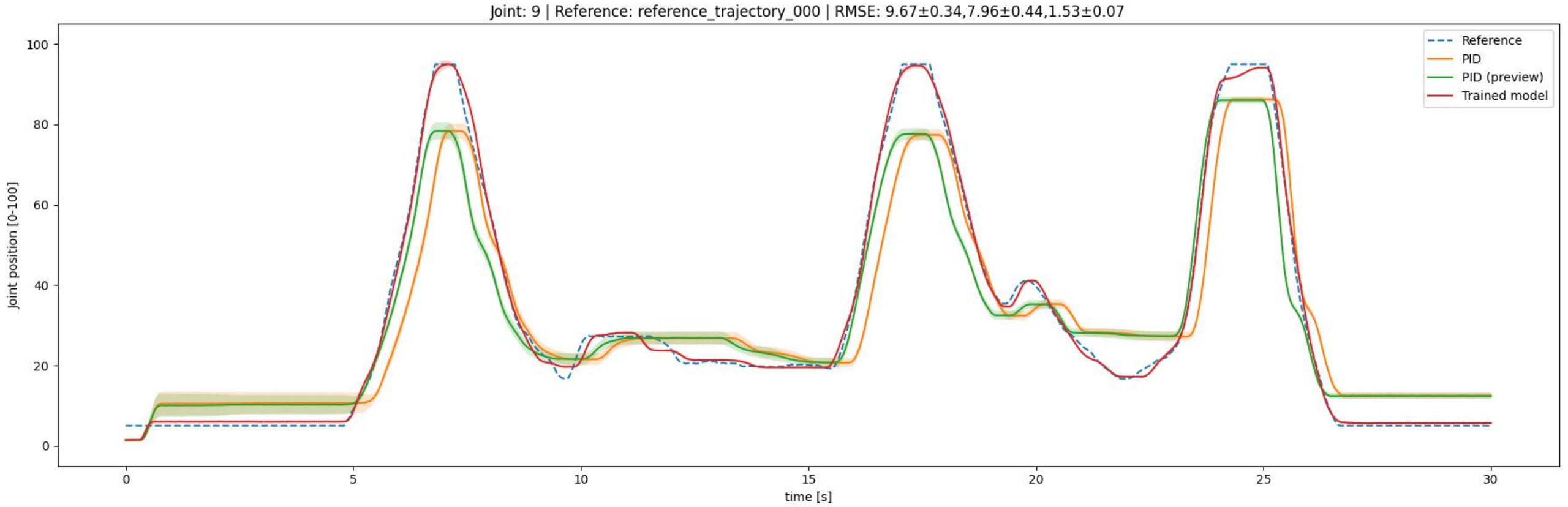
Tracking results for whole body

- Result of reference trajectory #0
  - Trained NN: MLP (#400), Standard scaler, Cont-async-all



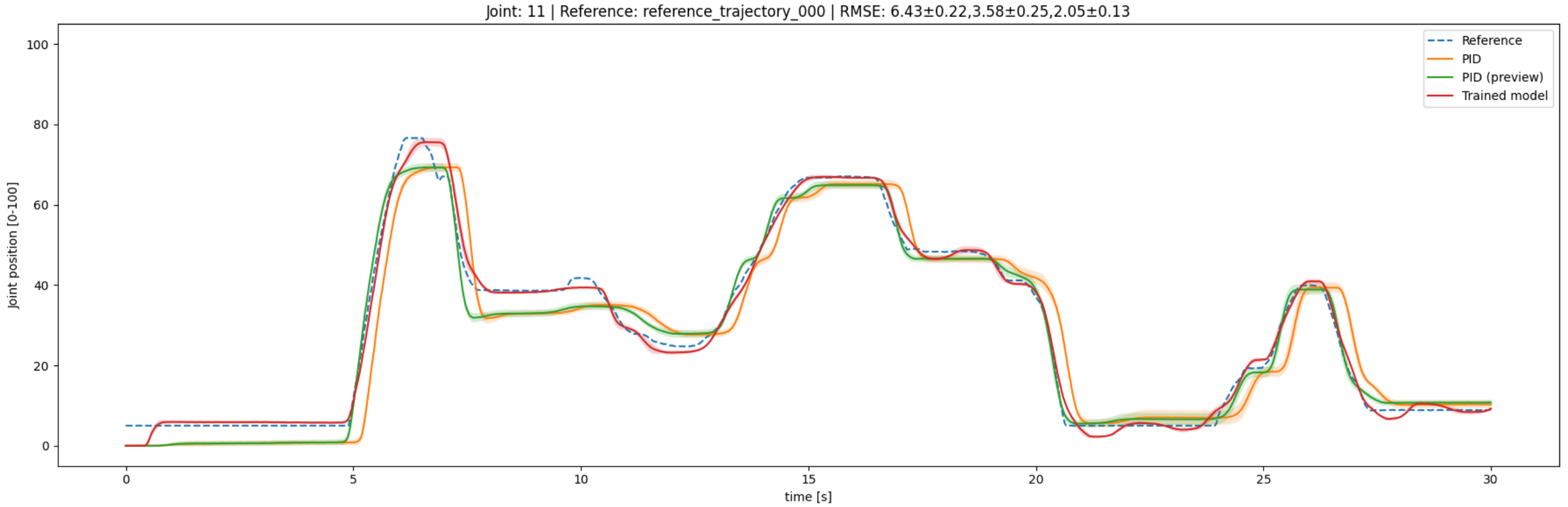
Tracking results for whole body

- Result of reference trajectory #0
  - Trained NN: MLP (#400), Standard scaler, Cont-async-all



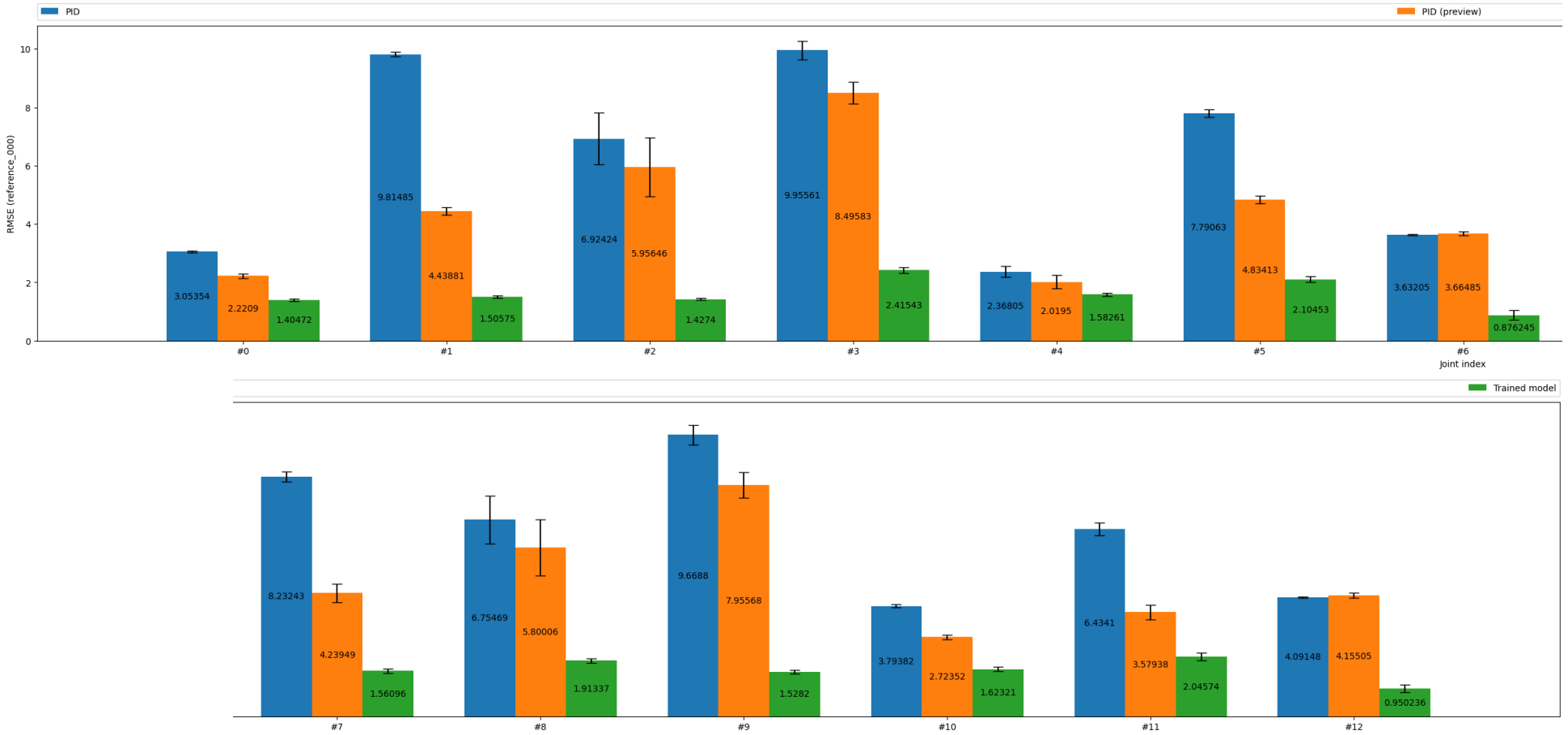
Tracking results for whole body

- Result of reference trajectory #0
  - Trained NN: MLP (#400), Standard scaler, Cont-async-all



Comparison of RMSE

- Result of reference trajectory #0
  - Trained NN: MLP (#400), Standard scaler, Cont-async-all

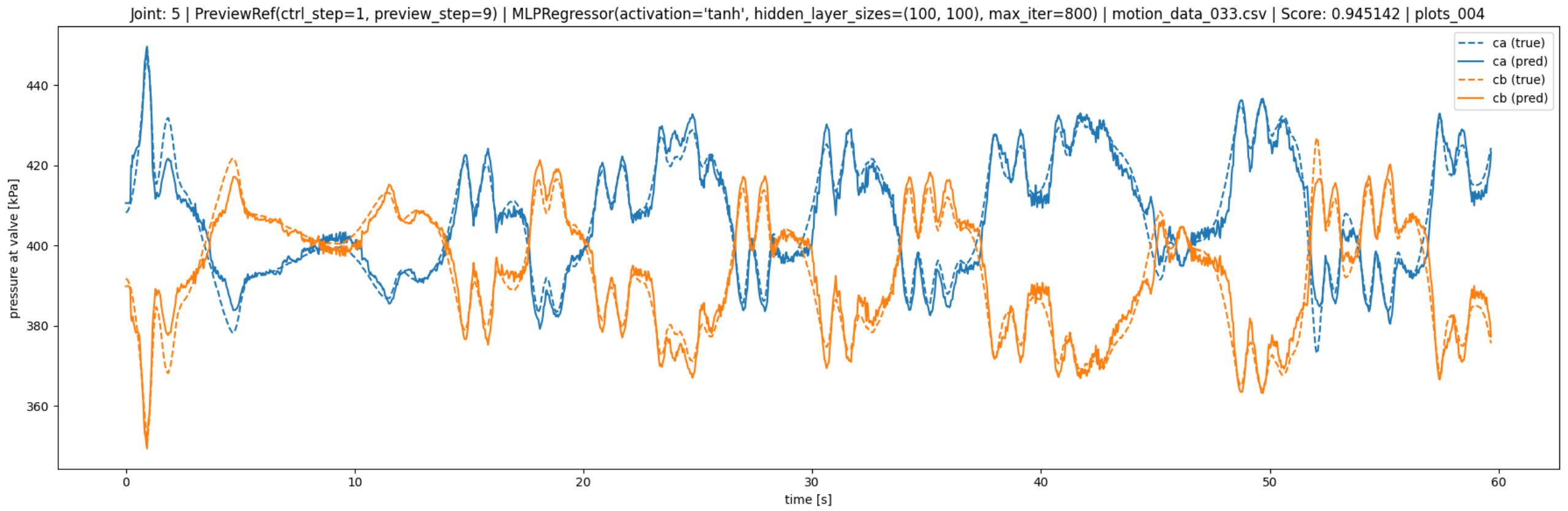


Preliminary ESN results

- ESN parameters (sampled 40 times from TPE algorithm (Optuna))
  - Neuron #: 2000, density: 0.0280, leaking rate: 0.916, ρ: 0.540, input scaling: 0.01555
- MLP
  - Neuron: #100-#100, activation: ReLU
- Preprocessing: Reference preview

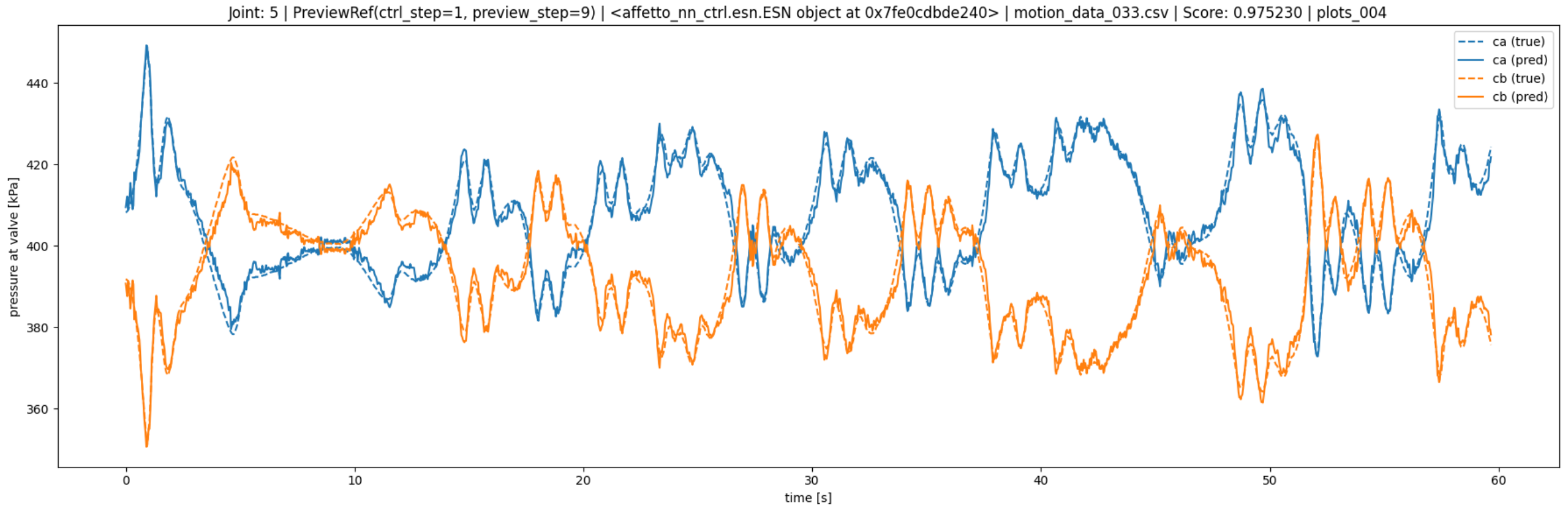
Preliminary ESN results

- MLP
  - Neuron: #100-#100, activation: ReLU
  - Reference preview, 72 data sets



Preliminary ESN results

- ESN
  - Neuron #: 2000, density: 0.0280, leaking rate: 0.916, ρ: 0.540, input scaling: 0.01555
  - Reference preview, 72 data sets

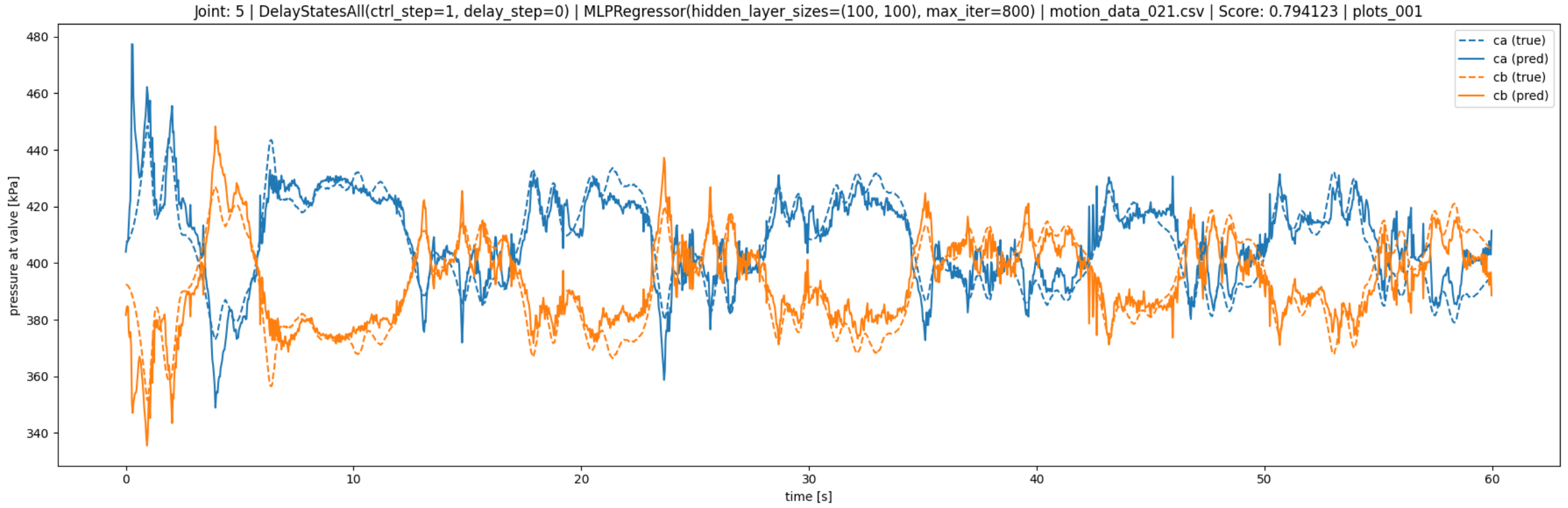


Preliminary ESN results

- ESN parameters (sampled 40 times from TPE algorithm (Optuna))
  - Neuron #: 2000, density: 0.183, leaking rate: 0.723, ρ: 1.1284, input scaling: 0.01055
- MLP
  - Neuron: #100-#100, activation: ReLU
- Preprocessing: Without delay embedding

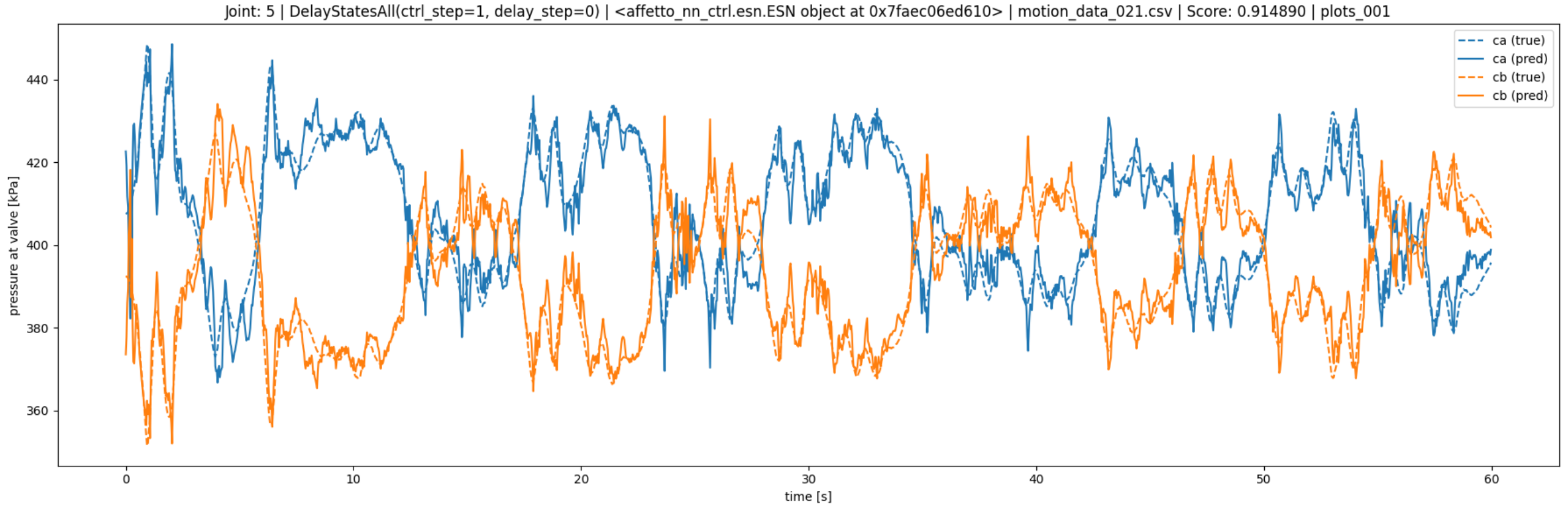
Preliminary ESN results

- MLP
  - Neuron: #100-#100, activation: ReLU
  - Without delay embedding, 72 data sets



Preliminary ESN results

- ESN
  - Neuron #: 2000, density: 0.183, leaking rate: 0.723, ρ: 1.1284, input scaling: 0.01055
  - Without delay embedding, 72 data sets

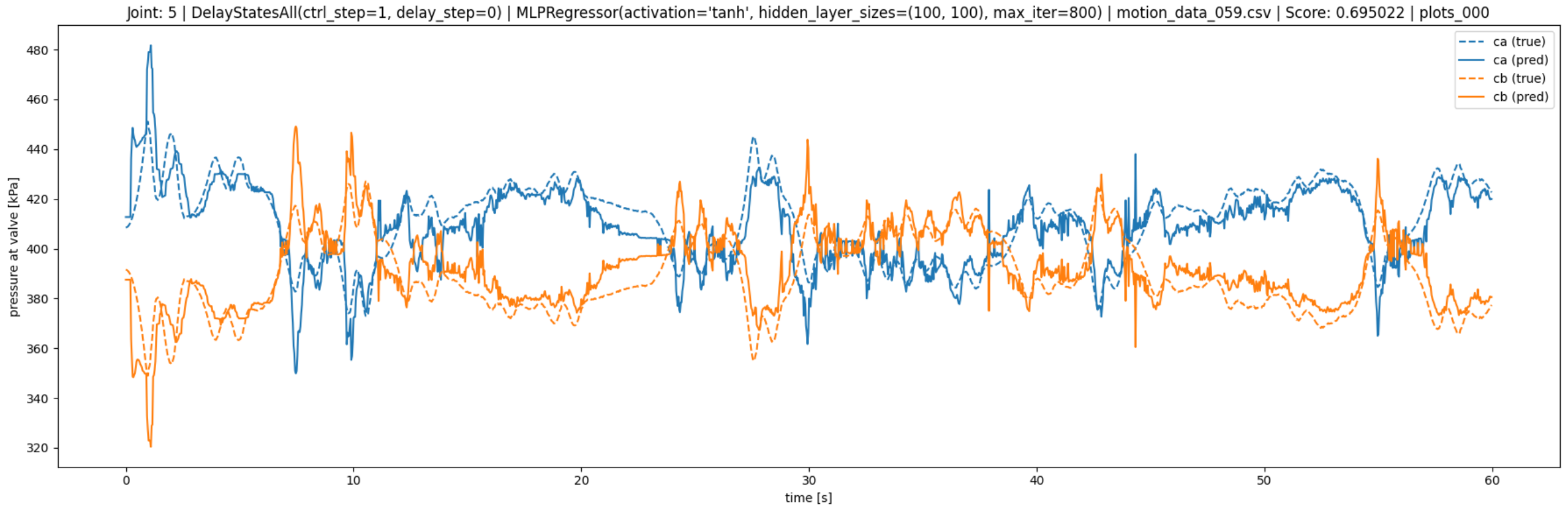


Preliminary ESN results (small dataset)

- ESN parameters (sampled 40 times from TPE algorithm (Optuna))
  - Neuron #: 2000, density: 0.1995, leaking rate: 0.704, ρ: 0.872, input scaling: 0.0134
- MLP
  - Neuron: #100-#100, activation: ReLU
- Preprocessing: Without delay embedding

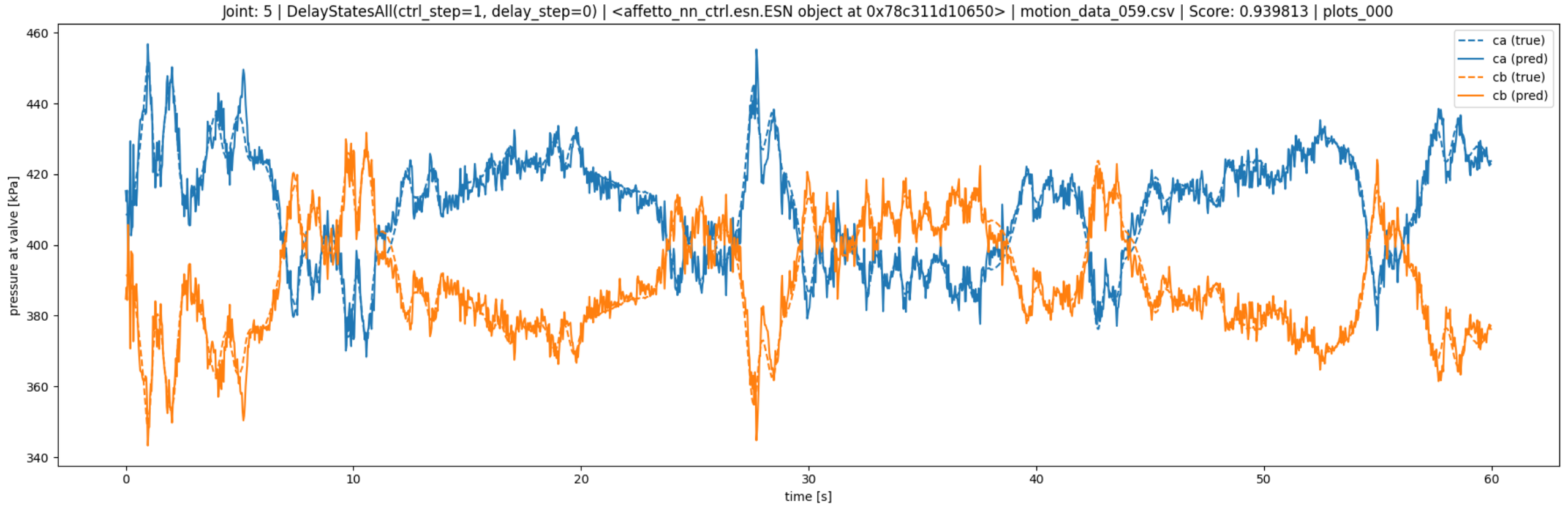
Preliminary ESN results

- MLP
  - Without delay embedding, 9 data sets



Preliminary ESN results

- ESN
  - Neuron #: 2000, density: 0.1995, leaking rate: 0.704, ρ: 0.872, input scaling: 0.0134
  - Without delay embedding, 9 data sets



Conclusion

- Random motion data collection
  - Data set properties may affect learning performance
- Parameter selection for preprocessing and multi-layered perceptron
  - Reference preview embedding works best
  - Hard to optimize parameters. Starting evaluation from small DOF
- Trajectory-tracking performance
  - Confirm enhancement of trajectory-tracking performance with reference preview embedding
- Journal title (tentative)
  - Trajectory-tracking performance enhancement of a 13-DOF pneumatically-actuated robot based on data-driven control with reference preview embedding
    - Note that the paper includes mechanical design and dynamic property investigation