

ID ML Project Technical Overview

Carla Takahashi

ML ID Project Overview

Background

- Joint initiative between Insertion Devices group and Software Group, driven by the ID team as a CPO project.
- First fully fledged project carried out with Machine Learning as main tool.

Motivation

- Beam orbit distortions by Insertion Devices currently corrected with measured feed-forward tables
 - Very time consuming
- Tables might be invalidated over time, or with different machine parameters (e.g. machine upgrades)
- New modes of ID operation can require several hundreds of measurement points e.g. universal mode.
- Prototyping of ML models that effectively replace measured feed-forward tables

Beam Orbit Distortion from Insertion Devices

- **Current Challenge**
 - Insertion Devices (IDs) introduce **orbit distortions** due to residual field integrals.
 - Compensation relies on **feed-forward tables**, which are:
 - Time-consuming to measure
 - Sensitive to changes in accelerator settings
 - Difficult to scale for complex ID configurations (e.g., Universal Mode)
- **Motivation for ML**
 - Automate compensation without extensive measurements
 - Improve maintainability and adaptability of orbit correction
 - Prepare for future operational scenarios and optical changes
- **Approach**
 - Use **archived operational data** (ID gaps, BPMs, correctors)
 - Train **neural network models** to replicate and eventually improve feed-forward behavior
 - Embed ML model into control loop via Tango Device Server

train_model_pipeline

2025-08-05, 21:05:59

Options

Home

Dags

Assets

Browse

Admin

Security

Docs

User

Create Run Folders

Extract Data

Transform Data

Split into Training and Testing Datasets

Train Model

Evaluate Model

Clean up run folders

React Flow

2025-08-05, 21:05:59

success

Add a note

Clear Run

Mark Run as...

Logical Date

Run Type

Start

End

Duration

Dag Version(s)

2025-08-05, 21:05:59

manual

2025-08-05, 21:35:24

2025-08-05, 21:40:23

00:04:58

v6

Task Instances

Audit Logs

Code

Details

Asset Events

Q Search Tasks

All States

Task ID	Map Index	State	Start Date	End Date	Try Number	Operator
Clean up run folders		success	2025-08-05, 21:40:21	2025-08-05, 21:40:22	1	_PythonDeci
Evaluate Model		success	2025-08-05, 21:40:02	2025-08-05, 21:40:20	1	_PythonDeci
Train Model		success	2025-08-05, 21:36:29	2025-08-05, 21:40:00	1	_PythonDeci
Split into Training and Testing Datasets		success	2025-08-05, 21:36:23	2025-08-05, 21:36:28	1	_PythonDeci
Transform Data		success	2025-08-05, 21:36:21	2025-08-05, 21:36:22	1	_PythonDeci
Extract Data		success	2025-08-05, 21:35:28	2025-08-05, 21:36:20	2	_PythonDeci
Create Run Folders		success	2025-08-05, 21:35:25	2025-08-05, 21:35:27	2	_PythonDeci

IDML Achromat 04

Provide Feedback

Experiment ID: 79

Artifact Location: mlflow-artifacts/79

Description Edit

Q metrics.rmse < 1 and params.model = "tree"

Time created ▾

State: Active ▾

Datasets ▾

Sort: Created ▾

Table

Chart

Evaluation

Export model

Run Name	Created	It	Duration	Source	Models	Metrics	
						val_rmse	val_mse
pipeline_keras_seq_...	6 days ago	24.3s	mild	idml_achro_14	-	-	-
pipeline_keras_seq_...	6 days ago	4.5s	mild	-	-	0.085180...	0.0111201...
pipeline_keras_seq_...	6 days ago	6.0s	mild	-	-	0.086746...	0.0129365...
pipeline_keras_seq_...	6 days ago	4.3s	mild	-	-	0.0781664...	0.0095889...
pipeline_keras_seq_...	6 days ago	4.7s	mild	-	-	0.0738204...	0.0104365...
pipeline_keras_seq_...	6 days ago	4.8s	mild	-	-	0.0736116...	0.0087836...
pipeline_keras_seq_...	6 days ago	5.0s	mild	-	-	0.1062127...	0.0181608...
pipeline_keras_seq_...	6 days ago	4.4s	mild	-	-	0.0908139...	0.0126278...
pipeline_keras_seq_...	6 days ago	4.2s	mild	-	-	0.0756769...	0.0098915...
pipeline_keras_seq_...	6 days ago	5.0s	mild	-	-	0.0813032...	0.0077125...
pipeline_keras_seq_...	6 days ago	5.4s	mild	-	-	0.0637358...	0.0073360...
pipeline_keras_seq_...	6 days ago	4.6s	mild	-	-	0.0521001...	0.0056932...
pipeline_keras_seq_...	6 days ago	4.4s	mild	-	-	0.0445703...	0.0032643...
pipeline_keras_seq_...	6 days ago	4.6s	mild	-	-	0.0529950...	0.0052946...
pipeline_keras_seq_...	6 days ago	4.9s	mild	-	-	0.0744292...	0.0095616...
pipeline_keras_seq_...	6 days ago	4.3s	mild	-	-	0.0578107...	0.0083248...
pipeline_keras_seq_...	6 days ago	4.9s	mild	-	-	0.0586113...	0.0062639...
pipeline_keras_seq_...	6 days ago	4.6s	mild	-	-	0.0729531...	0.0095238...
pipeline_keras_seq_...	6 days ago	4.7s	mild	-	-	0.0646460...	0.0071010...
pipeline_keras_seq_...	6 days ago	4.5s	mild	-	-	0.0576587...	0.0080991...
pipeline_keras_seq_...	6 days ago	4.3s	mild	-	-	0.0878104...	0.0130100...
pipeline_keras_seq_...	6 days ago	4.3s	mild	-	-	0.0617558...	0.0065199...
pipeline_keras_seq_...	6 days ago	5.3s	mild	-	-	0.0816275...	0.0115264...
pipeline_keras_seq_...	6 days ago	4.1s	mild	-	-	0.0531424...	0.0058797...
pipeline_keras_seq_...	6 days ago	4.8s	mild	-	-	0.0824222...	0.0134435...
pipeline_keras_seq_...	6 days ago	25.4s	airflow	idml_achro_13	-	-	-
pipeline_keras_seq_...	6 days ago	6.4s	airflow	-	-	0.0785222...	0.0106323...

100 matching runs

TestModelFromMLFlow.py

data_analysis (Python 3.11.8)

Data

Model

```
data: Dataframe = pl.read_csv(file_path/processes/data_file_name", use_python=True)
X: Dataframe = data.select(interest_features + normalization_features)
y: Dataframe = data.select(interest_targets)
```

```
model: Any | None = None
if flow.achromat_load_model(model_name):
    model = flow.achromat_load_model(model_name)
else:
    model = None
```

Pipeline

Model Prediction

```
fig: Figure, task, act = plt.subplots(2, 2, figsize=(16, 16),
    plt.tight_layout()
    act.set_title("test")
    )
fig: Figure, task, act = plt.subplots(2, 2, figsize=(16, 16),
    plt.tight_layout()
    act.set_title("predict")
    )
fig: Figure, task, act = plt.subplots(2, 2, figsize=(16, 16),
    plt.tight_layout()
    act.set_title("test")
    )
fig: Figure, task, act = plt.subplots(2, 2, figsize=(16, 16),
    plt.tight_layout()
    act.set_title("predict")
    )
```

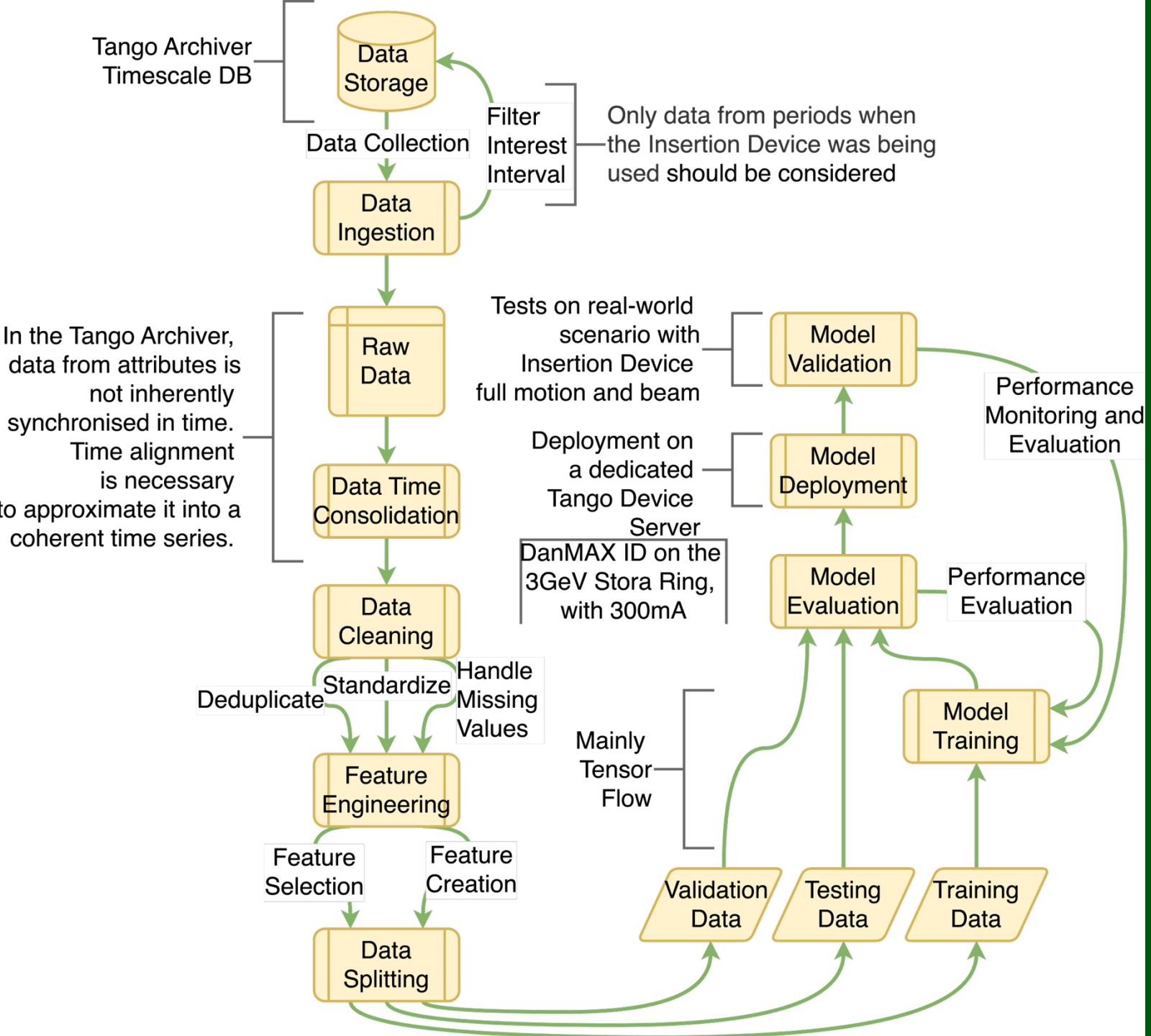
Extract Data schema

```
model_info.signature_dict
```

Infrastructure for Scalable and Trackable ML Development

- MLFlow for Experiment Tracking and Model Registry
 - Used to log training runs, parameters, metrics, and model versions
 - Enables reproducibility and collaboration
- Hosted on OKD (OpenShift)
 - Initial deployment for testing and integration
 - Provides containerized environment for scalable ML workflows
- Airflow for ETL and Training Pipelines
 - Currently running in a local Docker image for prototyping
 - Plans to deploy on OKD for production-grade orchestration
 - Supports automated data extraction, preprocessing, training, and model registration

SOFTWARE GROUP

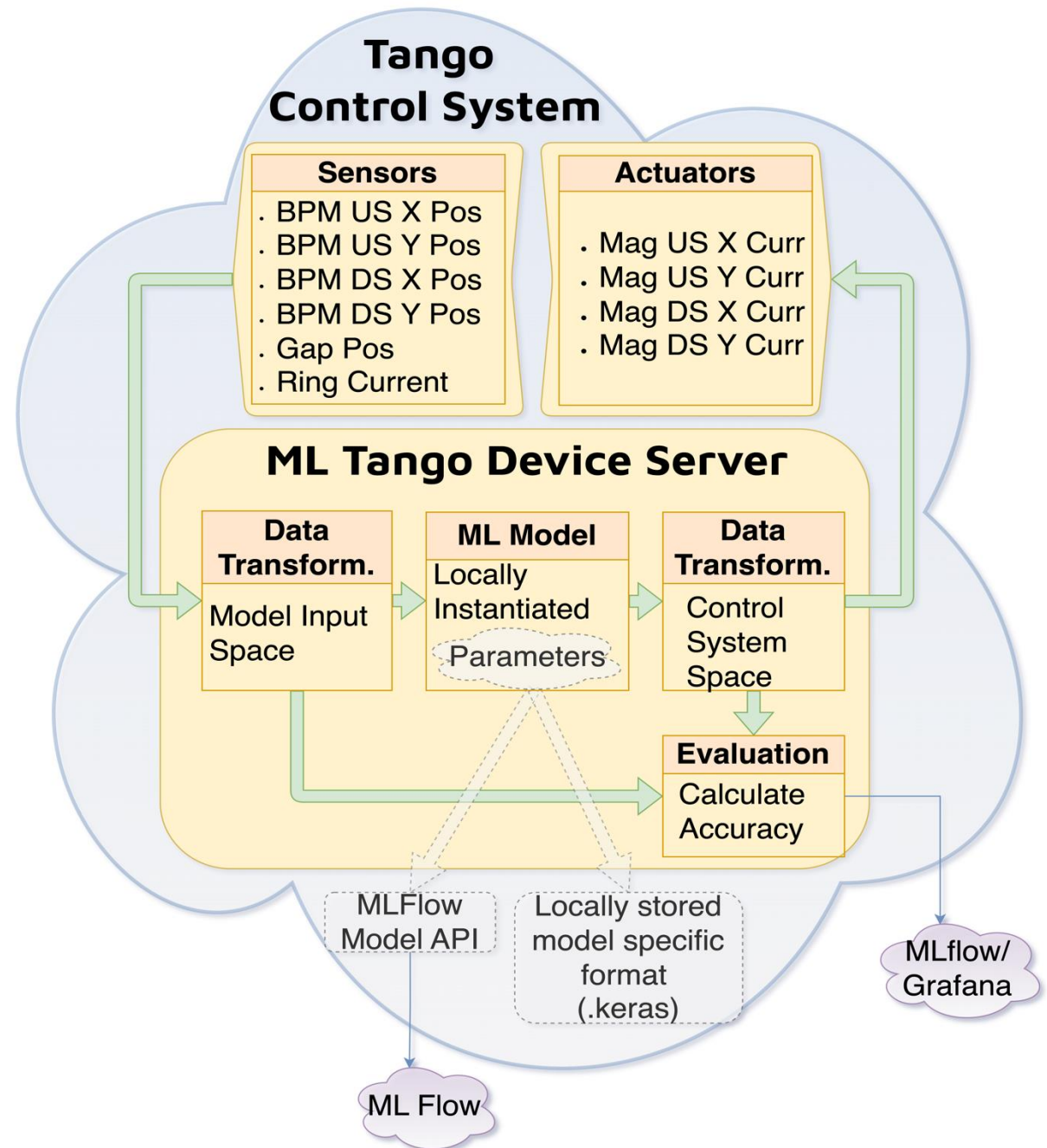


Retrieving Archived Data for ML Training

- **Data Source**
 - Historical data from MAX IV Archiver (TimescaleDB)
 - Includes: BPM readings, corrector currents, ID gaps, ring current
- **Filtering Strategy**
 - Focused on intervals when the ID was actively in use
 - Detected motion and position thresholds in ID gap time series
 - Reduced dataset size for efficient training and retrieval
- **Preprocessing Pipeline**
 - Time consolidation to align asynchronous attribute timestamps
 - De-duplication, completeness filtering, normalization
 - Feature engineering: e.g., BPM readings normalized by ring current to reduce noise
- **Outcome**
 - Enabled non-disruptive training using existing operational data
 - Avoided need for dedicated measurement campaigns

Embedding ML into the Orbit Control Loop

- **Control System Integration**
 - ML model hosted in a Tango Device Server
 - Interfaces with MAX IV control system for real-time inference
- **Inputs & Outputs**
 - Inputs: BPM positions (up/downstream X/Y), ID gap, ring current
 - Outputs: Corrector magnet currents (up/downstream X/Y)
- **Functionality**
 - Synchronizes and transforms incoming data
 - Performs periodic inference
 - Translates predictions back to control system space
 - Logs predictions and computes evaluation metrics(TBD)
- **Model Management**
 - Models loaded from local files or MLFlow registry(Need refactoring)
 - Supports flexible versioning and deployment



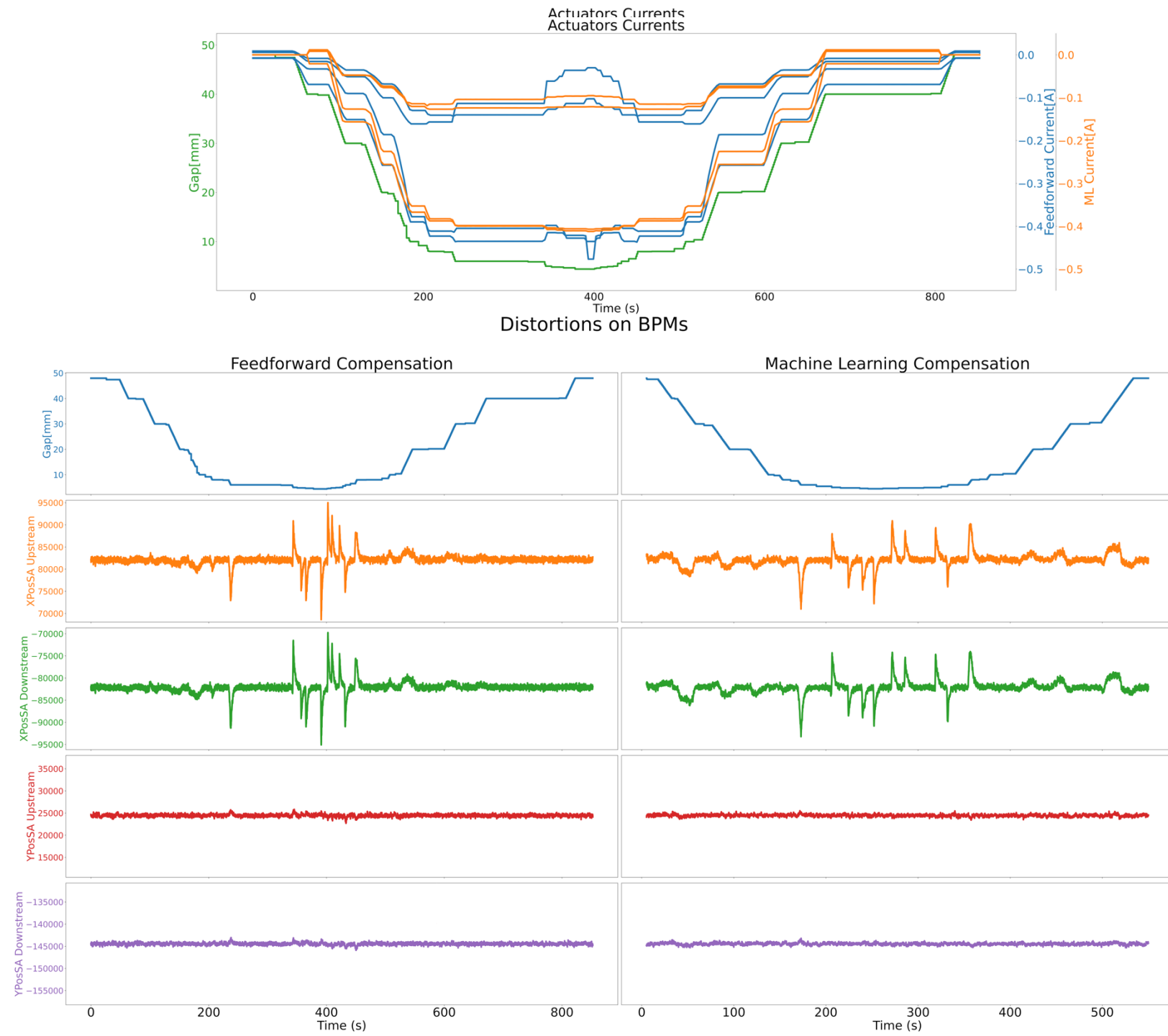
Experimental Results

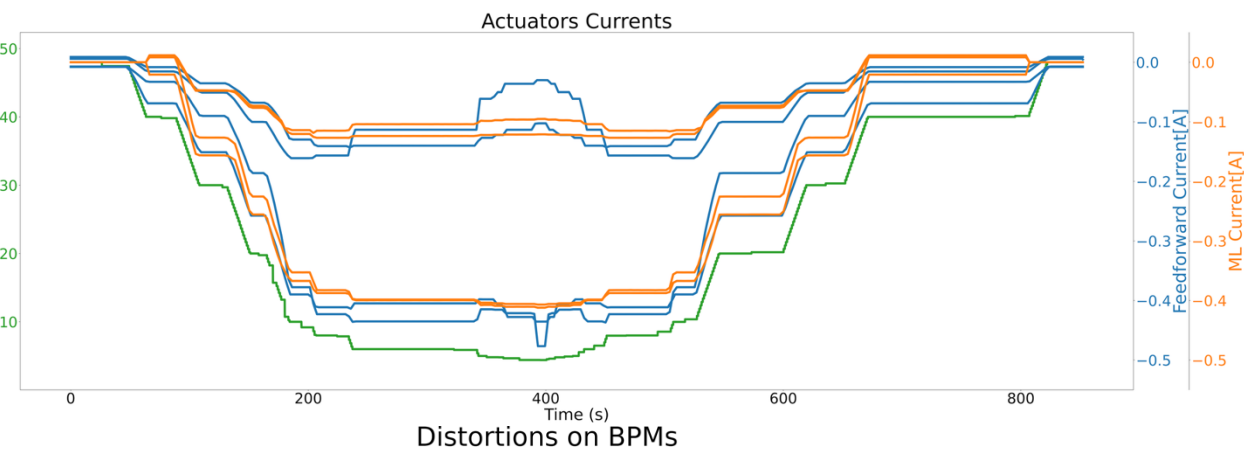
• Experimental Results

- ML model closely approximates feedforward correction across the ID gap range
- Shows reduced disturbance peaks at smaller gaps → improved local adaptation
- Some divergence at larger gaps (>30 mm), likely due to limited training data

• Impact

- Demonstrates feasibility of ML-based orbit compensation
- Indicates potential for adaptive control and reduced noise





Toward Robust and Adaptive ML-Based Orbit Control

• Live Evaluation Metrics (TBD)

- Integrate real-time performance metrics into monitoring tools
- Track model accuracy, prediction drift, and orbit stability
- Enable alerts for performance degradation

• Retraining and Deployment Policies (TBD)

- Define triggers for retraining:
 - Data shift detection
 - Degeneration metrics
 - Orbit deviation thresholds
- Establish rollback mechanisms and fallback models
- Automate model registration and deployment via MLFlow

• Goal

- Build a **resilient control loop** that adapts to changing conditions
- Ensure **safe and maintainable integration** of ML in operations