Walmart

The Alphabot Dynamics Library

A Challenge Project Presentation by Fletcher Cavanagh

July 19th, 2024

Agenda

1
Introduction

2
Business Case

3
Development Approach

4
Results

5
Leadership

6
Conclusion

Introduction

Fletcher Cavanagh

BS Mech. Eng. – NEU ‘23

Walmart ASR – SWE Intern

Alphabot Dynamics Library (ADL)

Walmart Advanced Systems and Robotics

- Formerly Alert Innovation
- Founded in 2013 by John Lert and Bill Fosnight to **reinvent retail through robotics**
- The Alphabot System: A warehouse automation system to increase Walmart's e-Grocery capacity and throughput

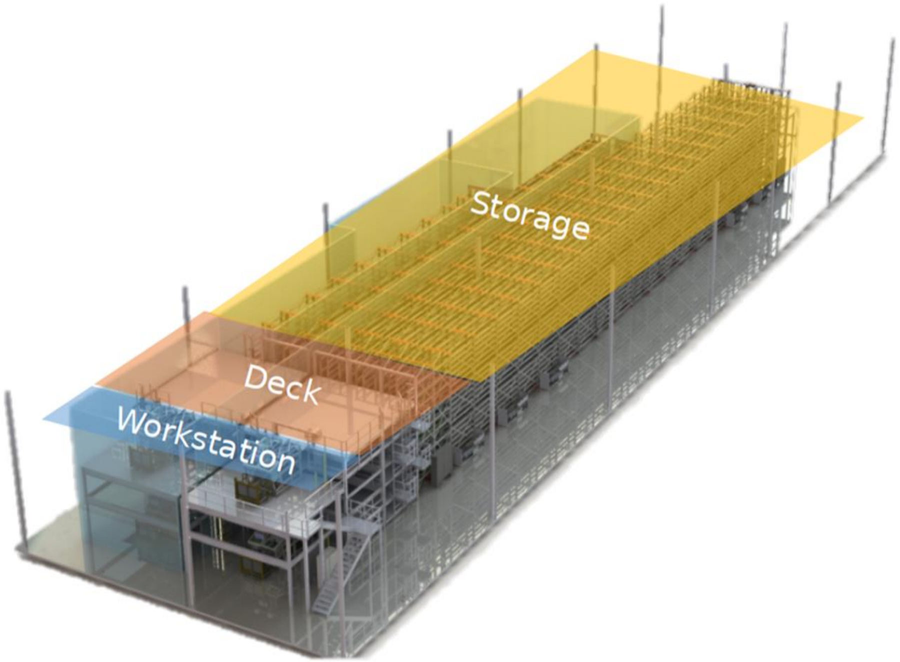

The Alphabot System in use at Walmart's Bentonville Location [1]

Walmart Advanced Systems & Robotics

The Alphabot System

- Active at **24+** Walmart sites
- Bots pick totes from the Structure’s **storage** area
- Navigate traffic via the **deck**
- Associates add and remove product from totes at **workstations**
- Increases a site's order capacity and throughput by **50%**

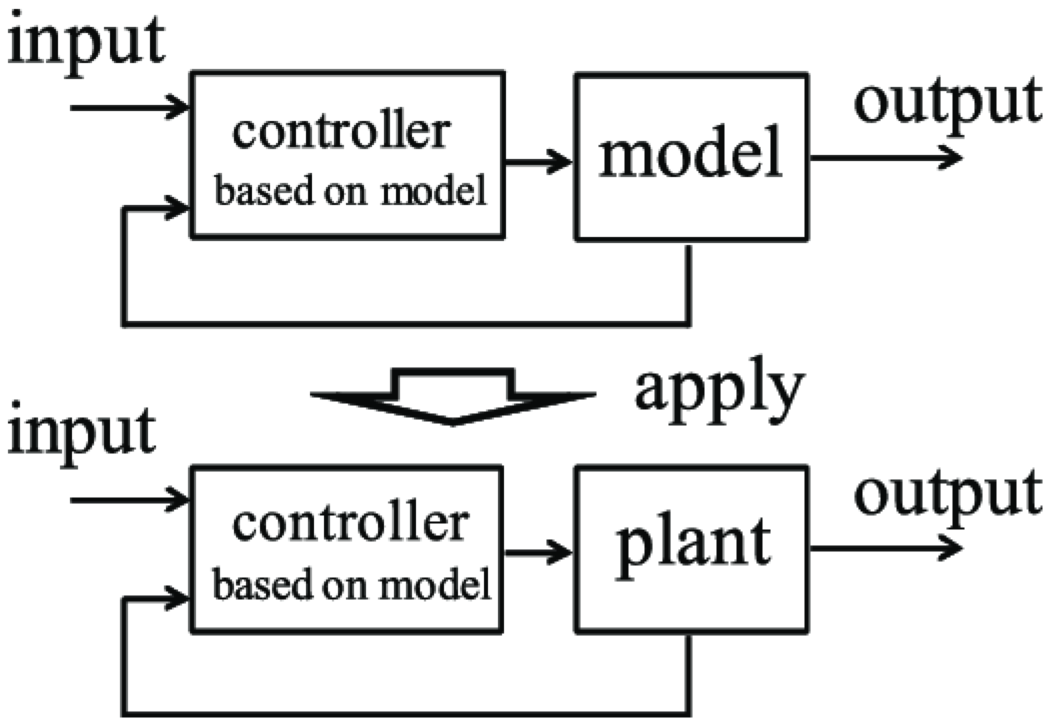
The Alphabot with a Tote [2]



The Alphabot Structure [3]

The Alphabot System

- Throughput can be increased by reducing system downtime
- Motion-related faults lead to **REDACTED%** of downtime
- Significant portion of motion faults are due to **wheel slip**
- Model-based control (MBC) can improve move accuracy and efficiency



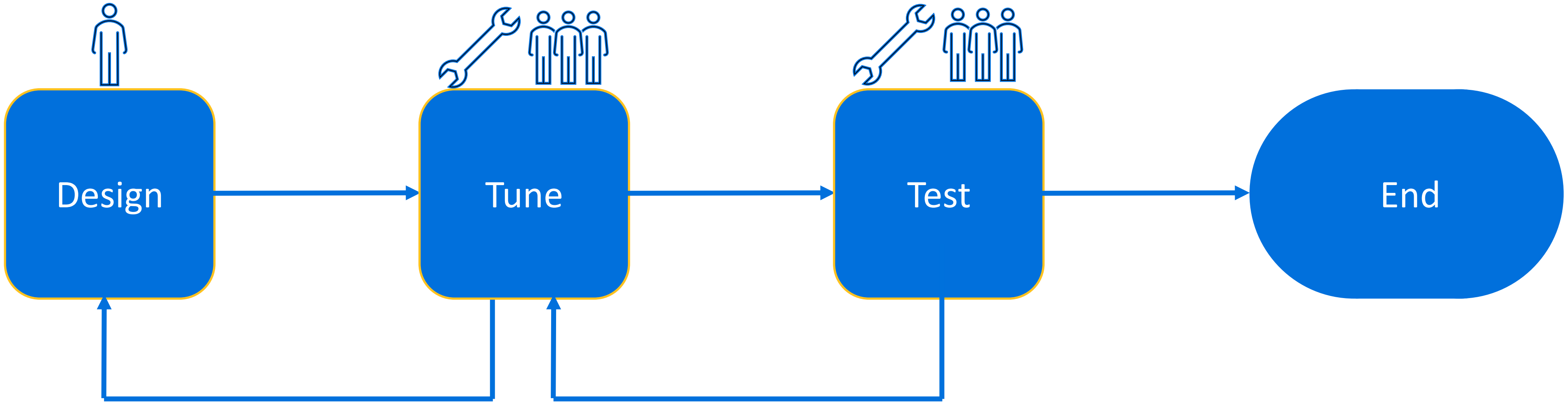
Basic Model-Based Control Schematic [4]

The Controls Team

- Designs and implements the Alphabot’s control systems
- Workflow **relies on** in-house **hardware tests**
- Requires support from mechanical, electrical, and testing teams
- Testing time **limited** and **valuable**

Controls Team Blocked Time per Sprint[1]

Average	Maximum
2 days	2.5 weeks



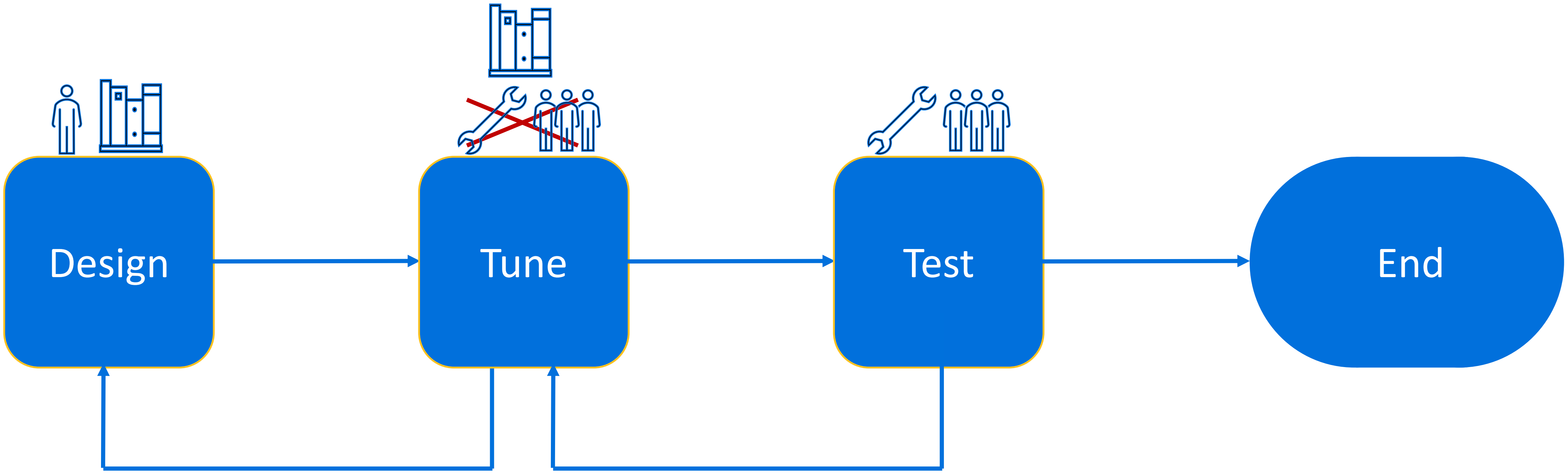
Simplified Controls Team Workflow

1: One sprint is equal to two weeks.

The Problem:

The Control Team's workflow is **dependent on teams and resources,** and engineers are **blocked** as a result.

Solution: The Alphabot Dynamics Library



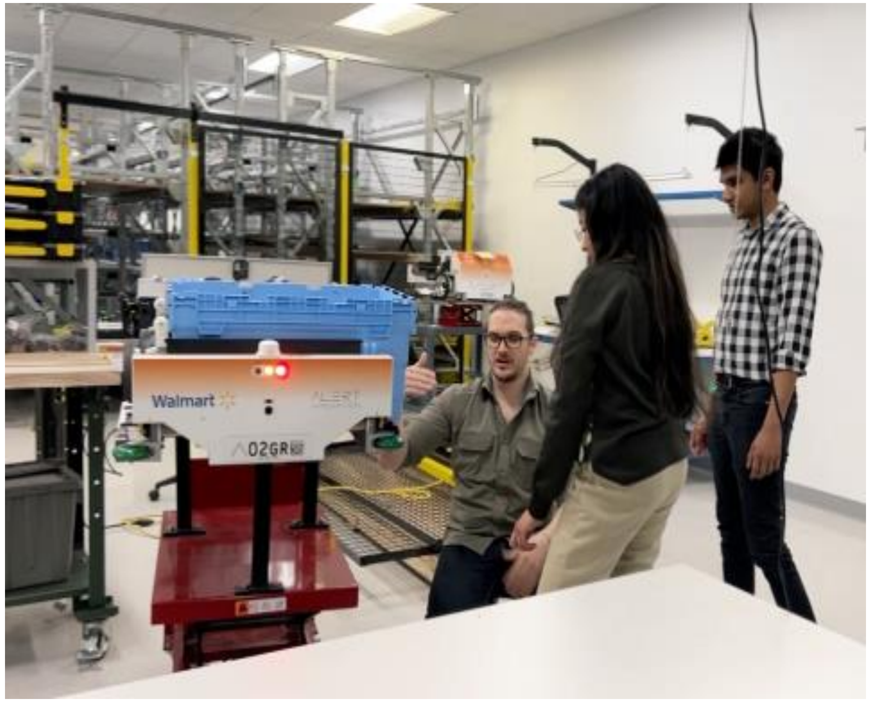
Dynamic models of the Alphabot with software for users to simulate and analyze models

Business Case

Why did we do this?

Internal Benefit

- Reduced hardware and personnel dependencies for key Controls team tasks
- Provided a workspace to modify, design and implement Alphabot control algorithms
- Foundation for future MBC development
- Informs bot’s physical limits in terms of physical parameters



Engineers inspecting an Alphabot at an in-house lab

ADL Time Savings per User per Sprint

Task	Savings (time)
Controller design	3 Hours
Controller tuning	15 Hours
Fault investigation	1 Hour

External Benefit

Real-time state estimation

Foundation for MBC

Slip and Force Estimation

Return on Investment	
Expenses	**Cost per project**
Candidate Labor	$56,288
PL #1 Licenses	$3,440
Total Expenses:	**$59,278**
Savings	**Gain per project**
Hardware Resources	$1,000
Controls User Labor	$75,000
Interdisciplinary Support Labor	$7,280
Total Savings:	**$83,280**
ROI	**40.5%**

$$\mathbf{ROI} = \frac{\mathbf{savings - cost}}{\mathbf{cost}} \times \mathbf{100} = \mathbf{40.5\%} = \mathbf{0.2\ FTE\ saved\ per\ year!}$$

Development Approach

How did we do it?

Customer Needs

- Applied methods from Ulrich and Eppinger's *Product Design and Development* [7]
- Critical step - informed the structure, functionality, and need for the project
- Conducted interviews, surveys, and research to define needs
- Some needs based on legacy constraints

ADL Model Summarized Customer Needs

Need	Importance (1-5)
Accurate	5
Robust	5
Portable	4

ADL Internal Software Summarized Customer

Need	Importance (1-5)
Configurable	5
Integrable	5
Documented	4

Product Specifications

- Drew from customer needs, research, and theory to create **quantifiable metrics** that determined ADL's success
- Focused on **accuracy** and **compatibility** with existing workflow

ADL Model Specifications

Specification	Target Value
Max prediction error	5%
Considers wheel slip	Yes
Represented in single file	Yes

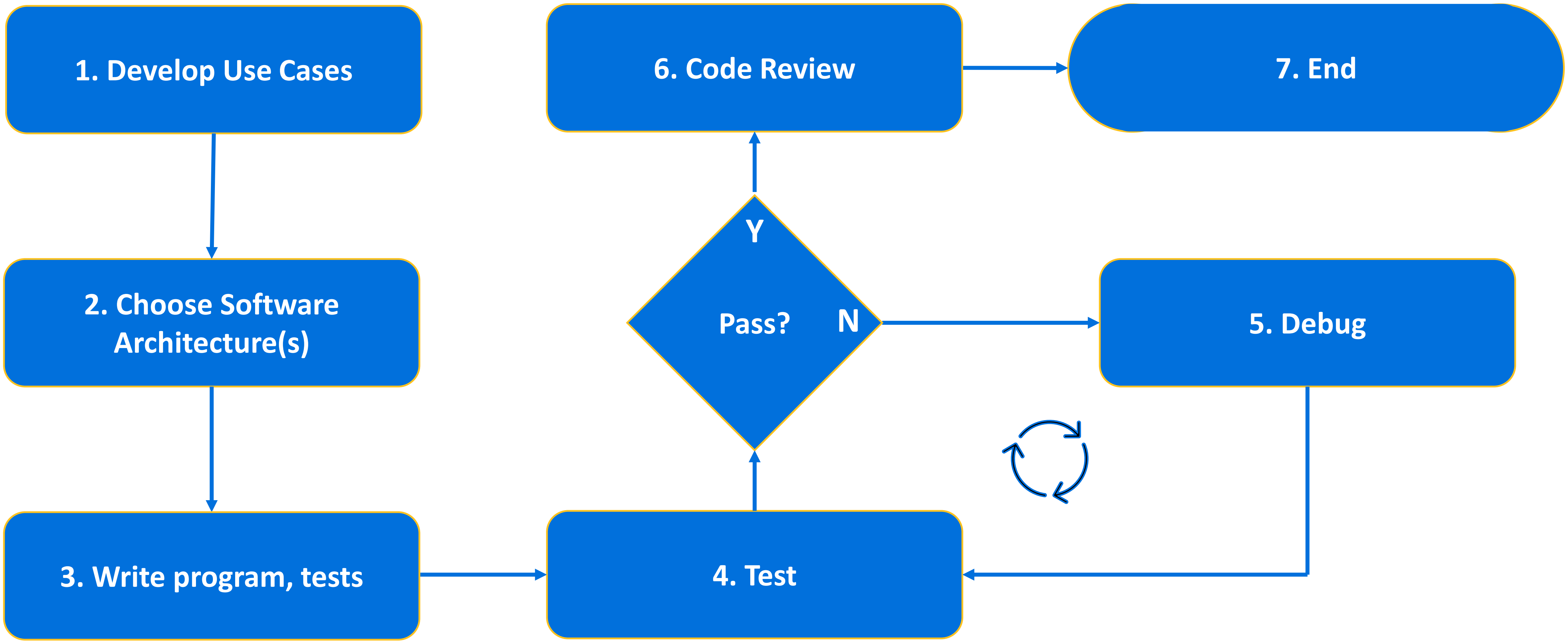
ADL Internal Software Specifications

Specification	Target Value
Accepts Alphabot data	Yes
Inject faults, noise	Yes
Unit tested	Yes

Software Development

Development Approach

Software Development Process



ADL Software Packages

Internal Software

- For Controls team engineers
- Provides user low-level control over model structure and simulation environment

External Software

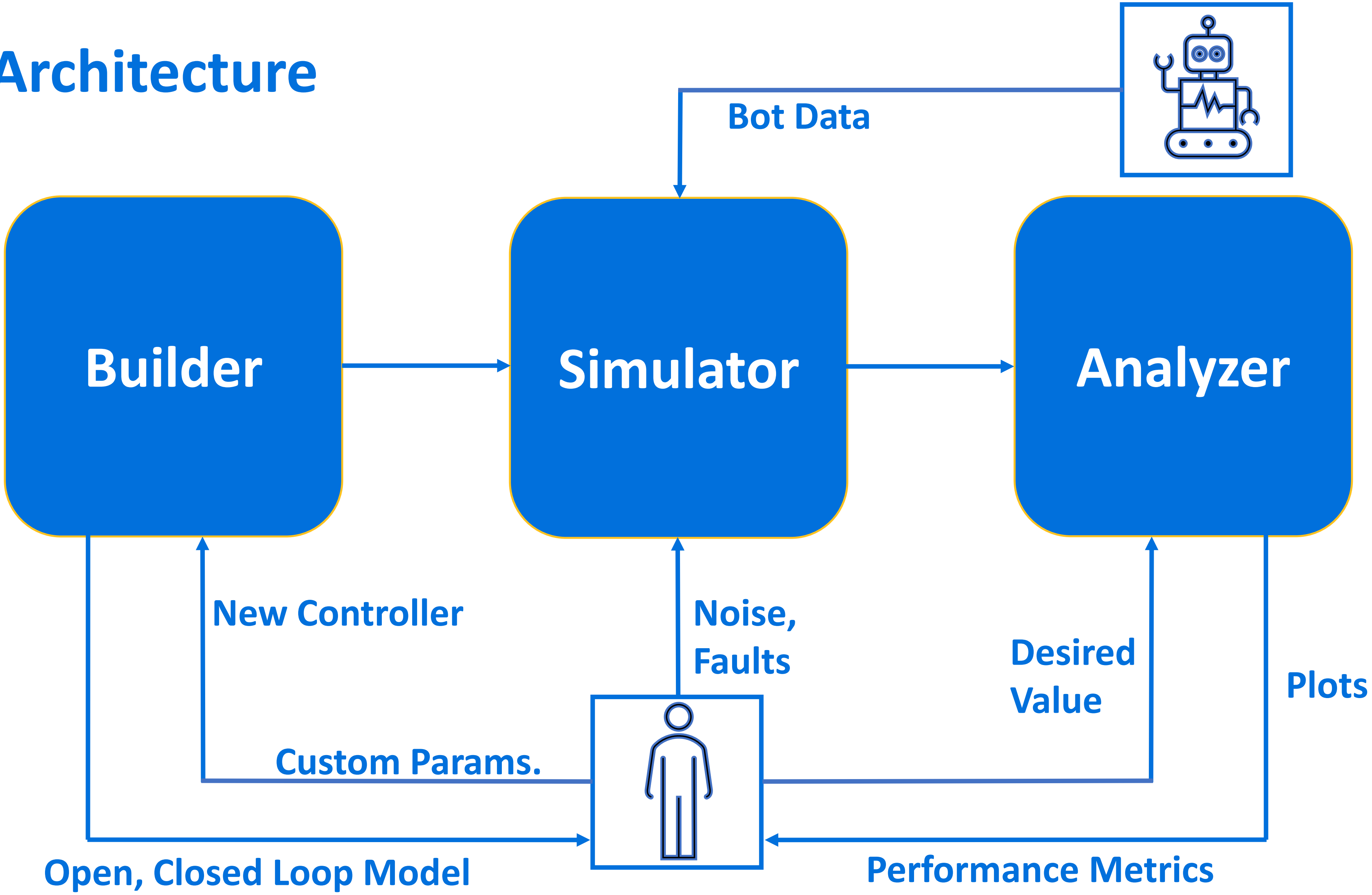
- For Alphabot application
- Memory and time efficient module that integrates with existing control software

Simulink® Toolbox

- For Controls team engineers
- Streamlined toolbox for rapid prototyping and development

Use Cases and Architecture

- Used product specs. to define use cases
- Architecture defined based on use cases and research
- Chose a class-based architecture



ADL Internal Software Architecture Diagram

Testing

1

Unit Testing

- Isolate functions to root-cause errors
- Validate performance of each submodule

2

Code Review

- Detailed analysis from coworkers
- Catch faulty design patterns and unnecessary dependencies

Dynamic Modeling

Development Approach

ADL Dynamic Models

Drive Wheel

- Equivalent to no-slip model suspended in air
- Open and closed-loop models

$$v = 0$$

No-Slip

- Builds on drive wheel model
- Considers full bot dynamics

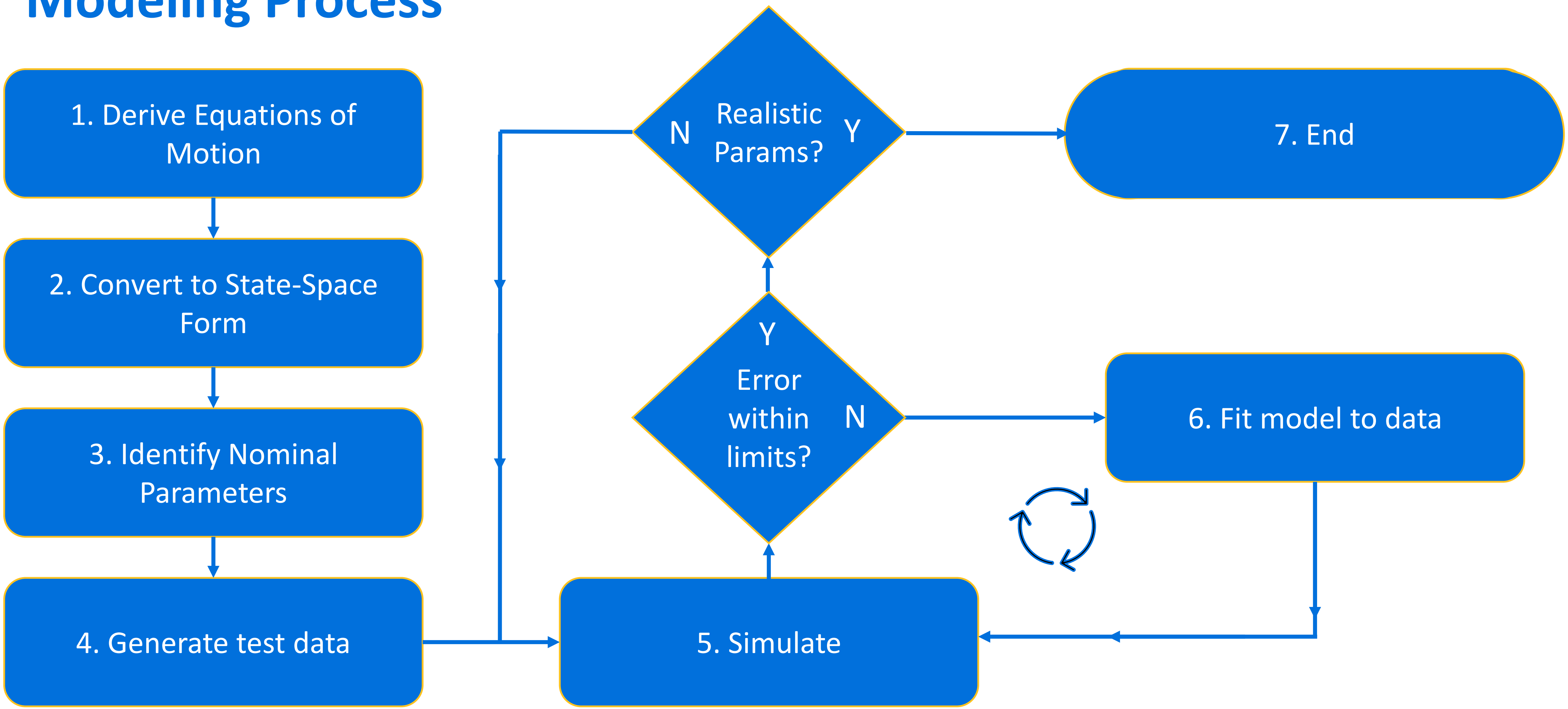
$$v = \omega r$$

Slip

- Extension of no-slip model without pure-rolling constraint
- Considers contact forces

$$v \neq \omega r$$

Modeling Process



Deriving Bot Models

- Applied **LaGrangian mechanics** [16] to avoid internal force interactions
- **Pfaffian matrix** [16] used to reduce model order
- Terms grouped into appropriate **state-space** matrices

General Nonholonomic Lagrange Dynamics

$$\mathbf{M}(q)\ddot{q} + \mathbf{C}(q,\dot{q})\dot{q} + \mathbf{F}(\dot{q}) + \mathbf{G}(q) + \tau_d = \mathbf{B}(q)\tau - \boldsymbol{\Lambda}^T(q)\lambda$$

$$\boldsymbol{S}^T(q)\boldsymbol{\Lambda}^T(q) = 0$$

No-Slip

$$\dot{v} = \dot{\phi}\mathbf{A}_{\boldsymbol{\phi}}v + \mathbf{A}v + \mathbf{B}u$$

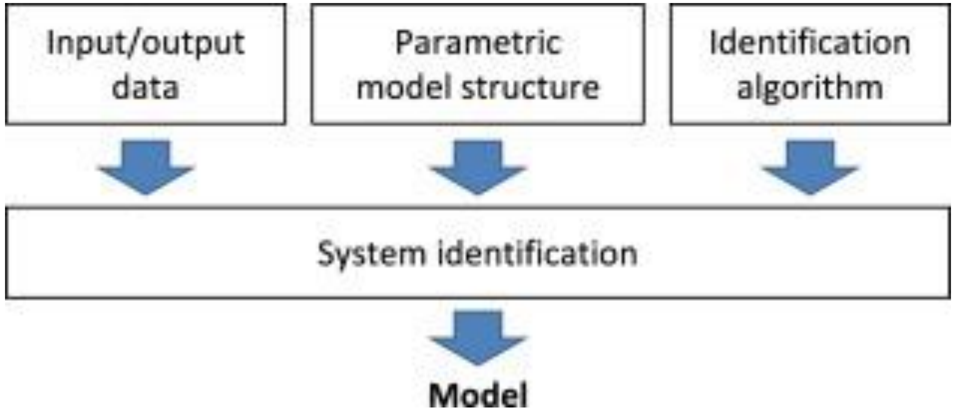
Slip

$$\dot{v} = \dot{\phi}\mathbf{A}_{\boldsymbol{\phi}}v + \mathbf{A}v + \mathbf{B}u$$

$$\ddot{\Theta} = \mathbf{I}^{-1}(\tau - rf)$$

System Identification and Validation

- Derived nominal values from multiple sources
  - Measurement
  - Experiment
  - Datasheets
  - Estimation
- Generated training and test data sets on the Alphabot
- Used Teflon Tape to **induce wheel slip**
- Tuned model with training data, measure performance vs. test data
- Applied iterative search methods to **fit model to data**



System Identification Process Inputs[9]

Results

How'd it go?

ADL Internal Software Results

Builder

- Modify model parameters and control structure
- Design and introduce new control methods

Simulator

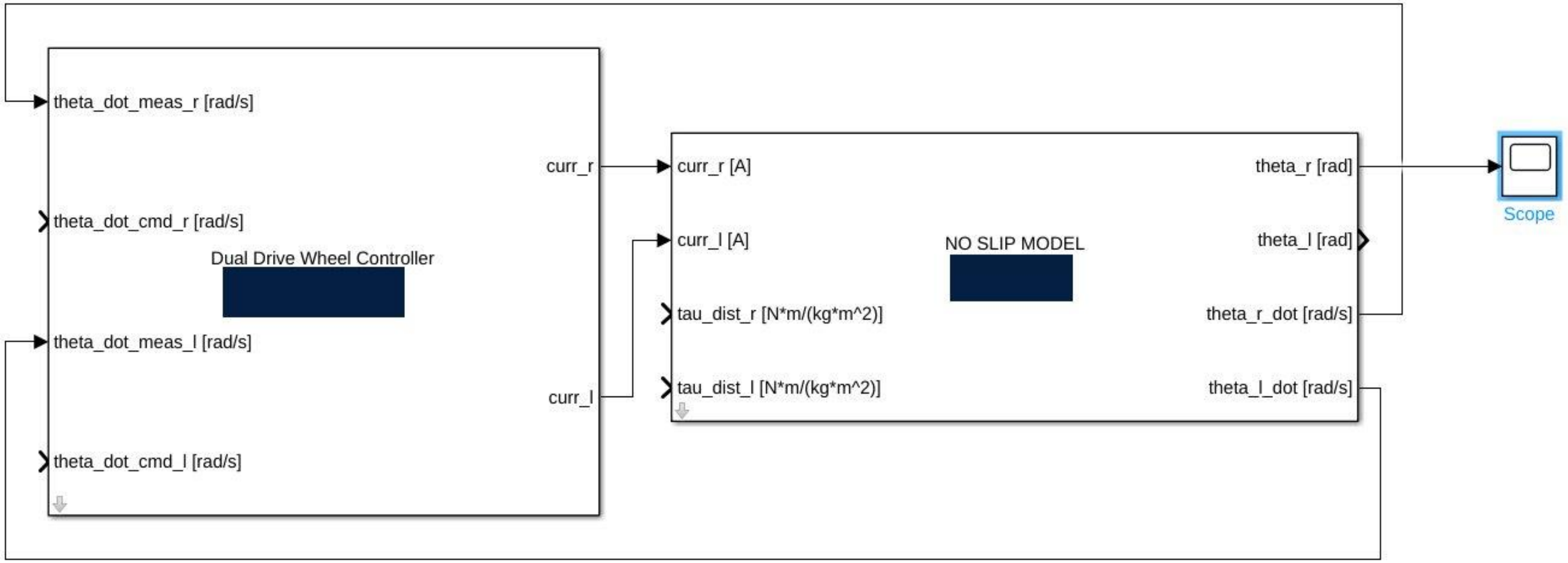
- Inject sensor noise and faults into simulation
- Compatible with Alphabot data

Analyzer

- Performance metrics for model and controller
- Graphical representation of model output

ADL Simulink® Toolbox

- All ADL models, controllers, and estimators represented
- Compatible with existing Simulink block sets



Closed-loop controller Implementation with the ADL Simulink Toolbox

ADL External Software Results

State Predictions

- Real-time state estimates on the Alphabot application

Context-Dependent

- Skips simulation when bot is in an unsupported move or location

Documented

- Graphical representation of model output
- Performance metrics for model and controller

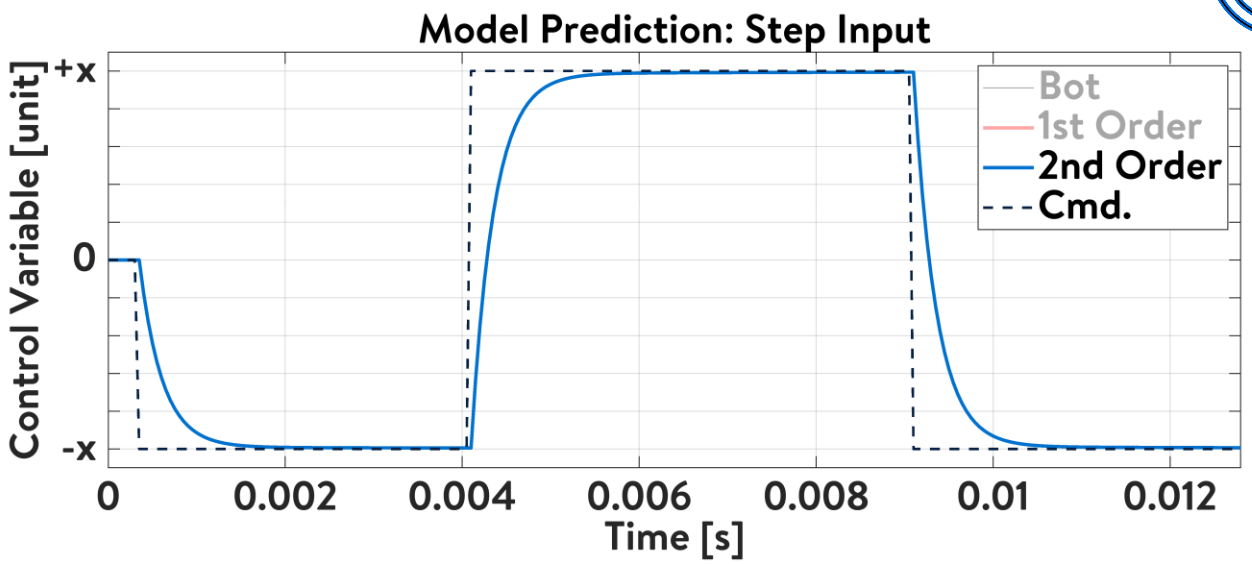
Results

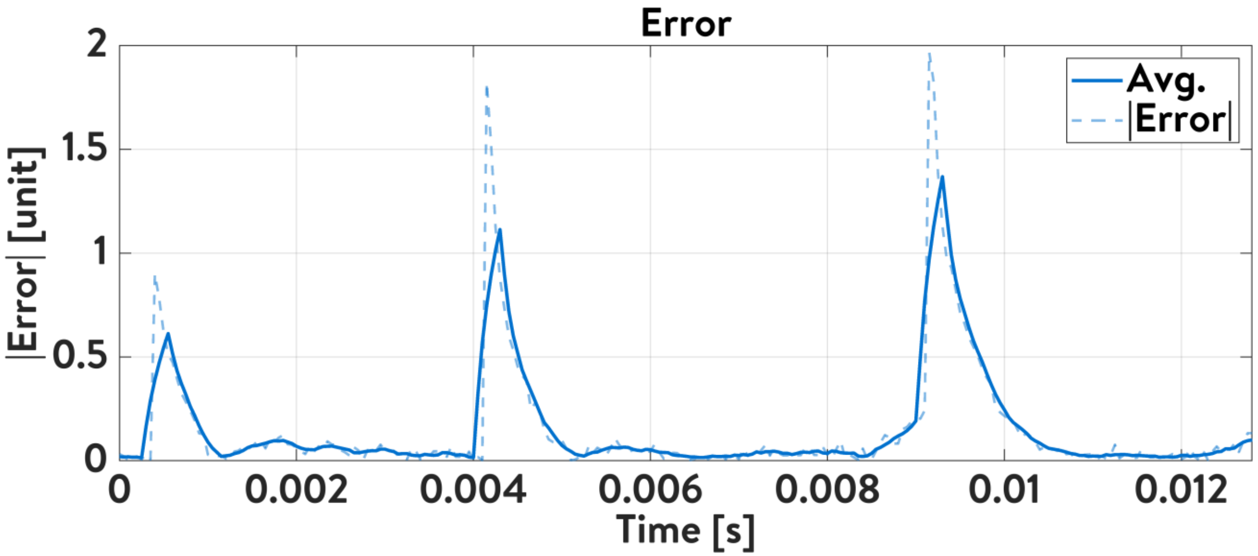
Dynamic Modeling

Drive Wheel

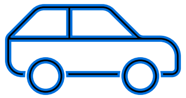
- Highest % accuracy of all models
- Predict motor's reaction to controller changes **without** hardware tests

Context	Avg |Error| [unit]	Avg Error [%]
Transient	0.51	-2.0%
Steady-State	0.03	0.3%
Full Move	**0.15**	**1.2%**



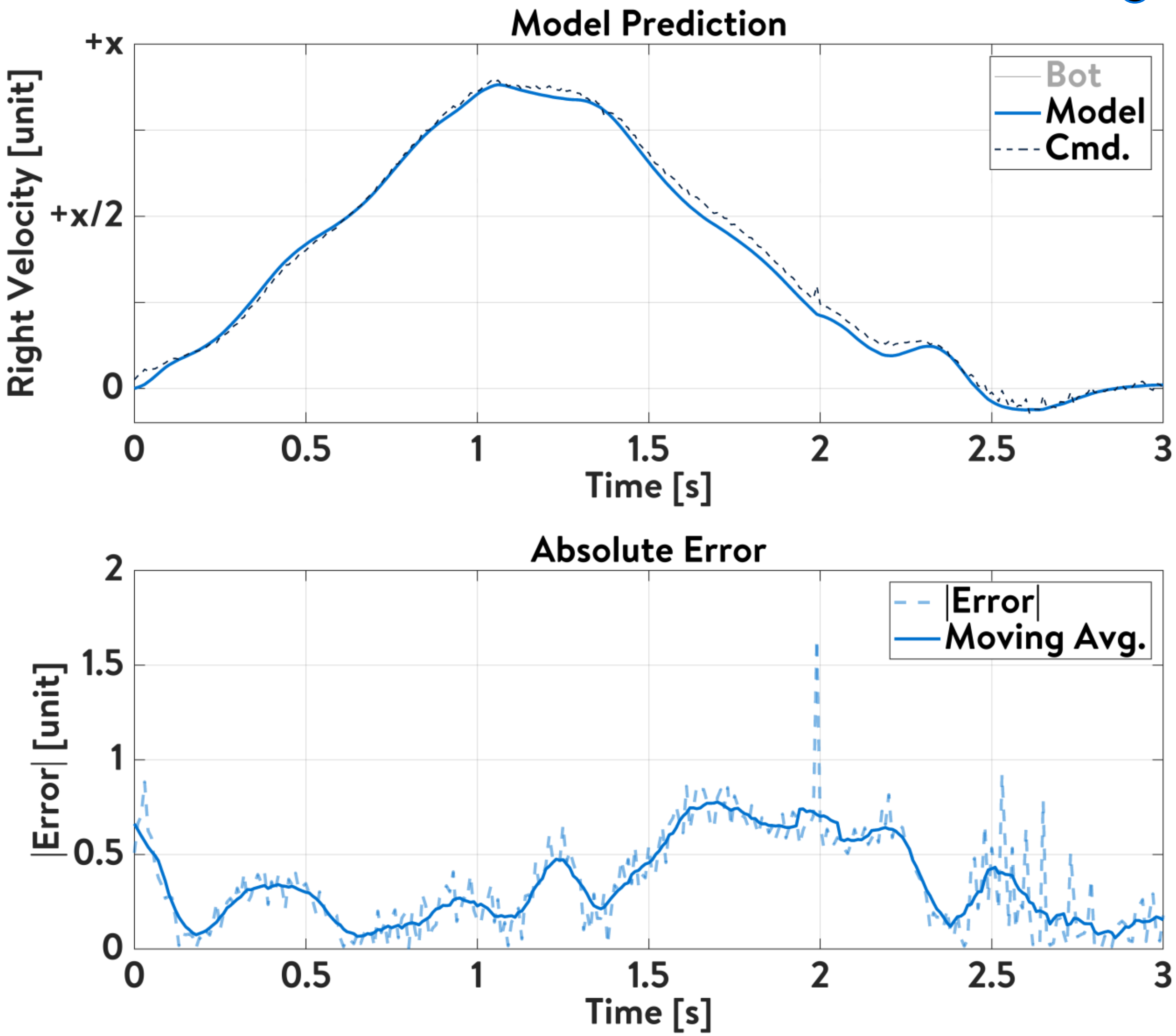


No-Slip: Translation



- Best performance during acceleration from rest
- Can be used to identify **disturbances** and **hardware faults**

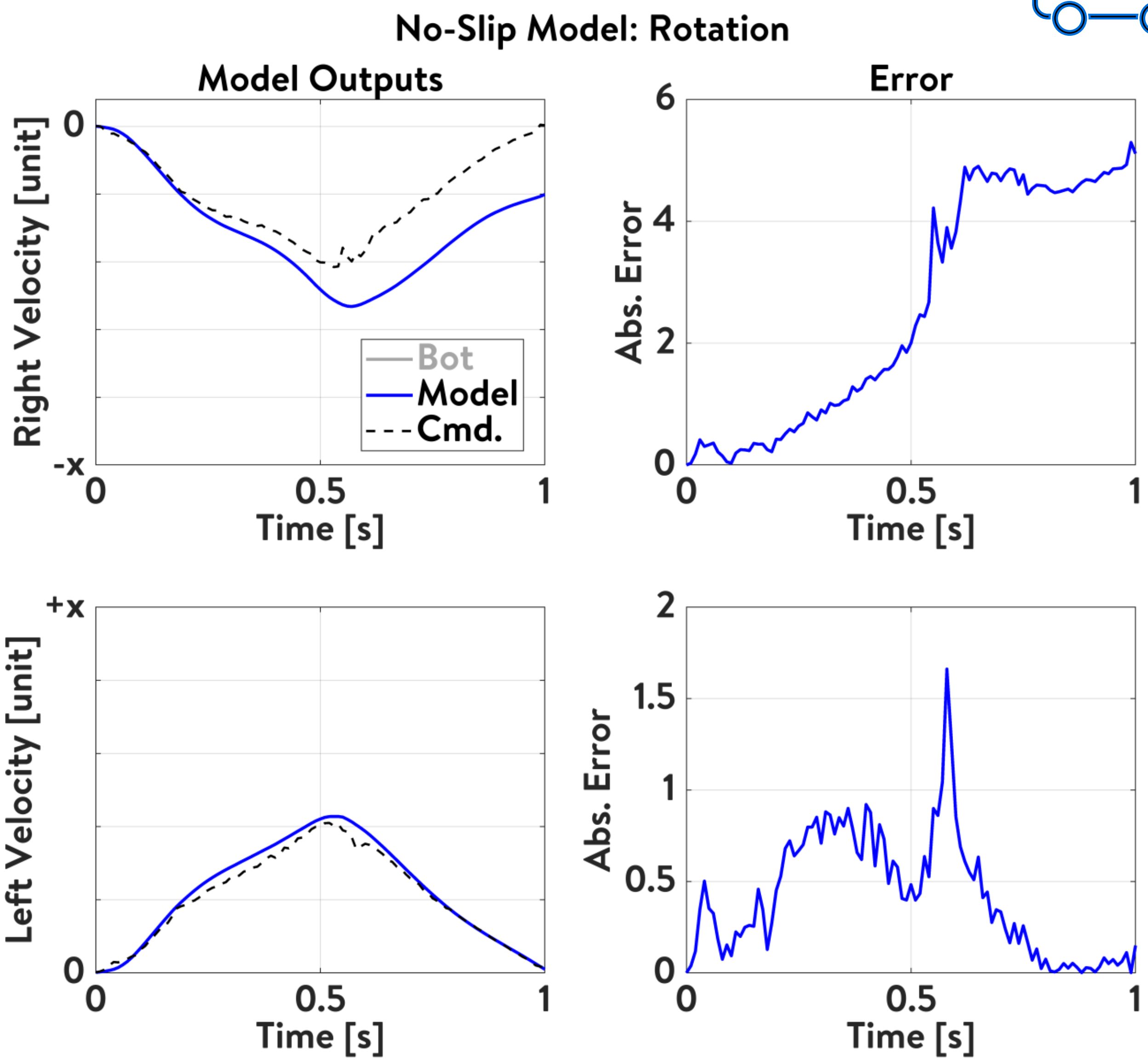
Context	Avg |Error|	Avg % Error
Accel.	0.24	-0.09%
Steady-State	0.48	-2.74%
Decel.	0.57	-8.46%
Full Move	**0.43**	**-4.48%**



No-Slip: Rotation

- Bulk of modeling effort
- Nonlinearity via mass eccentricity
- Better results with drive wheel model, but not analytically correct

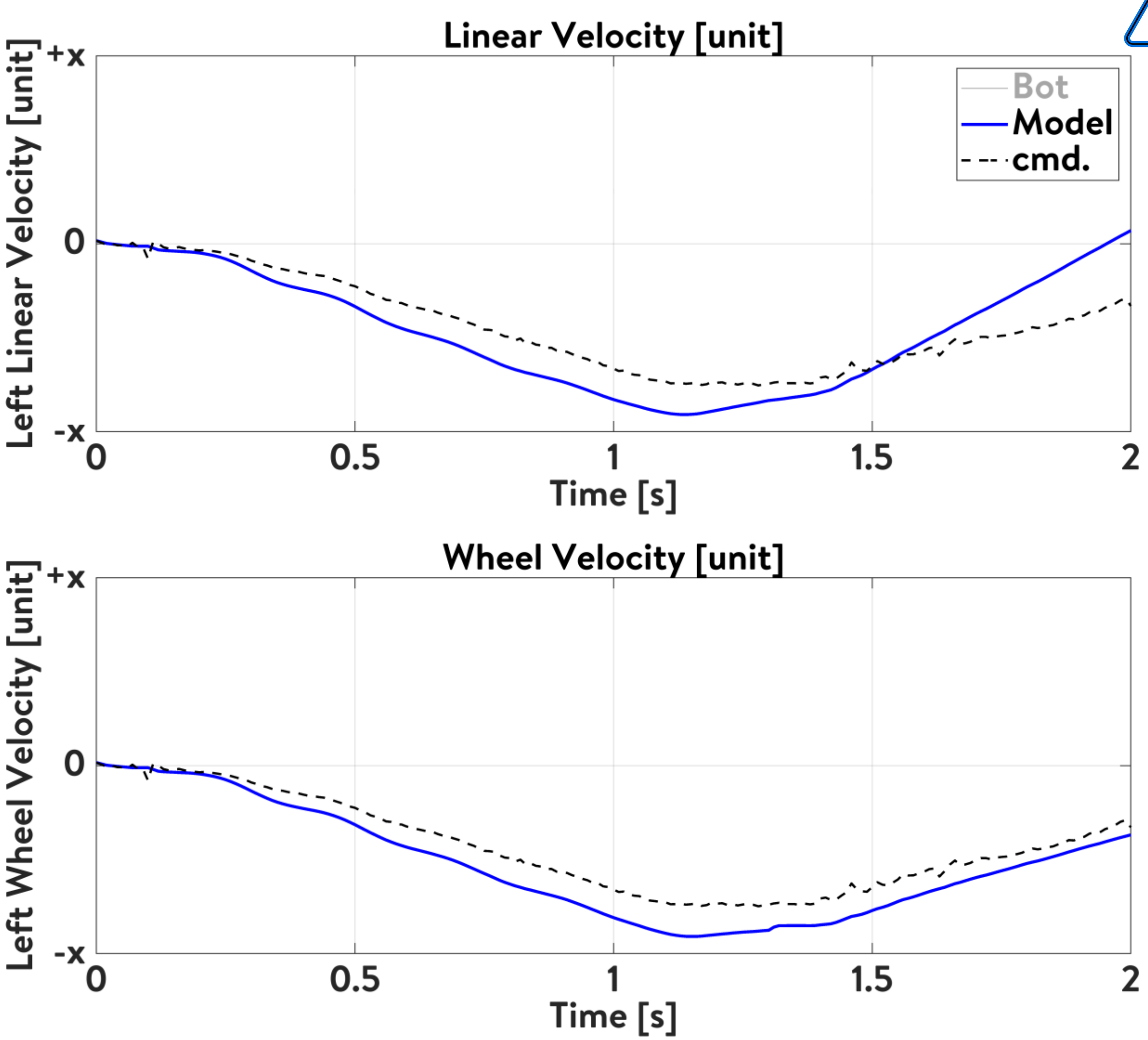
Channel	Avg |Error|	Avg % Error
Leading Wheel	0.50	8.6%
Trailing Wheel	3.22	31.3%
Average	**1.86**	**20.0%**



Slip: Teflon Tape Tests

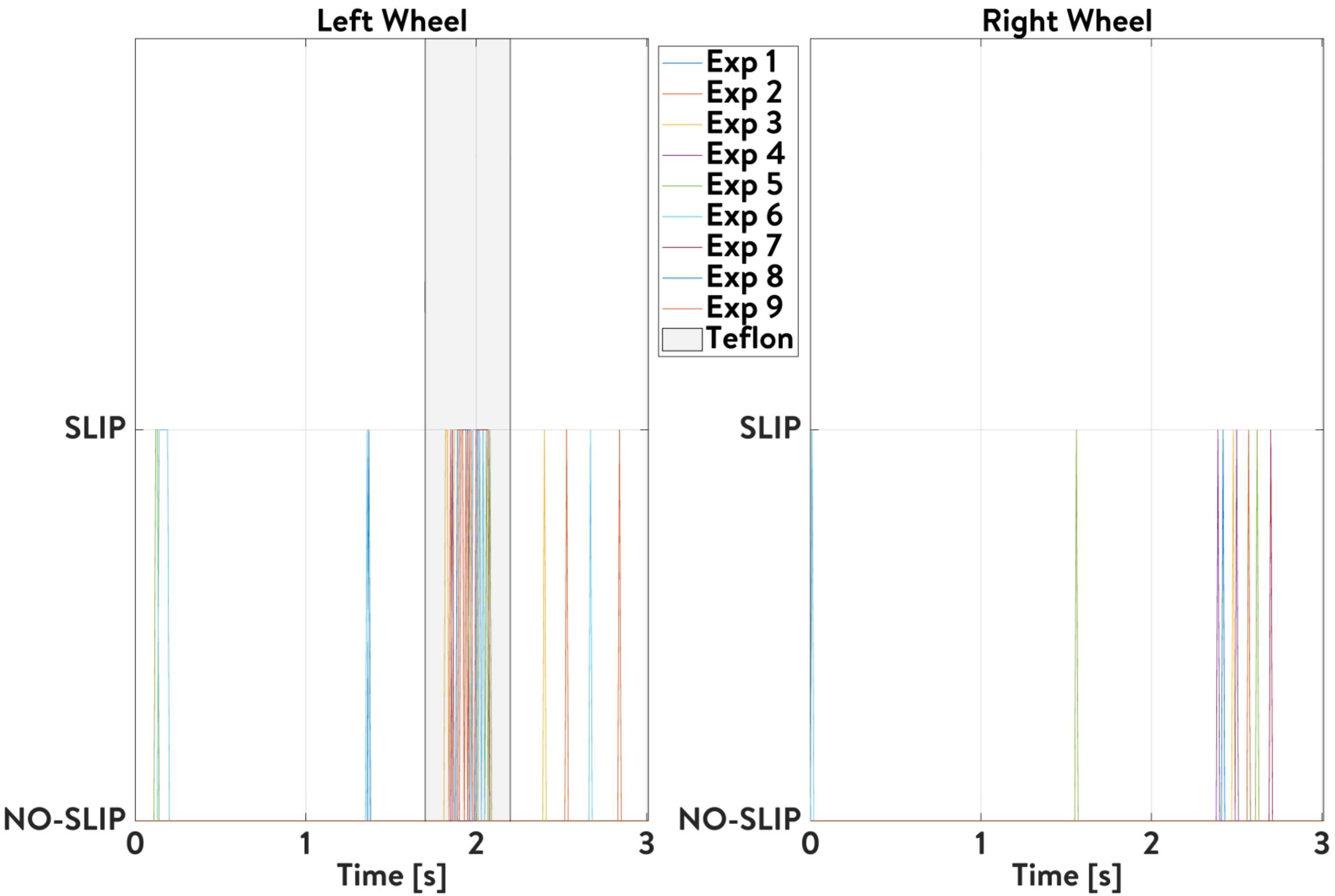
- Linear velocity drifts from test data
- Difficulties identifying the return to no-slip conditions
- Slightly better results on other Alphabots

Channel	Avg |Error| [unit]	Avg % Error
Linear Velocity	0.50	25.0%
Angular	2.90	27.4%
Average	**1.70**	**26.2%**



Slip: Slip Detection

- Slip detection algorithm tested with **Teflon tape data**
- Successfully identified slip in all experiments
- Some false identifications

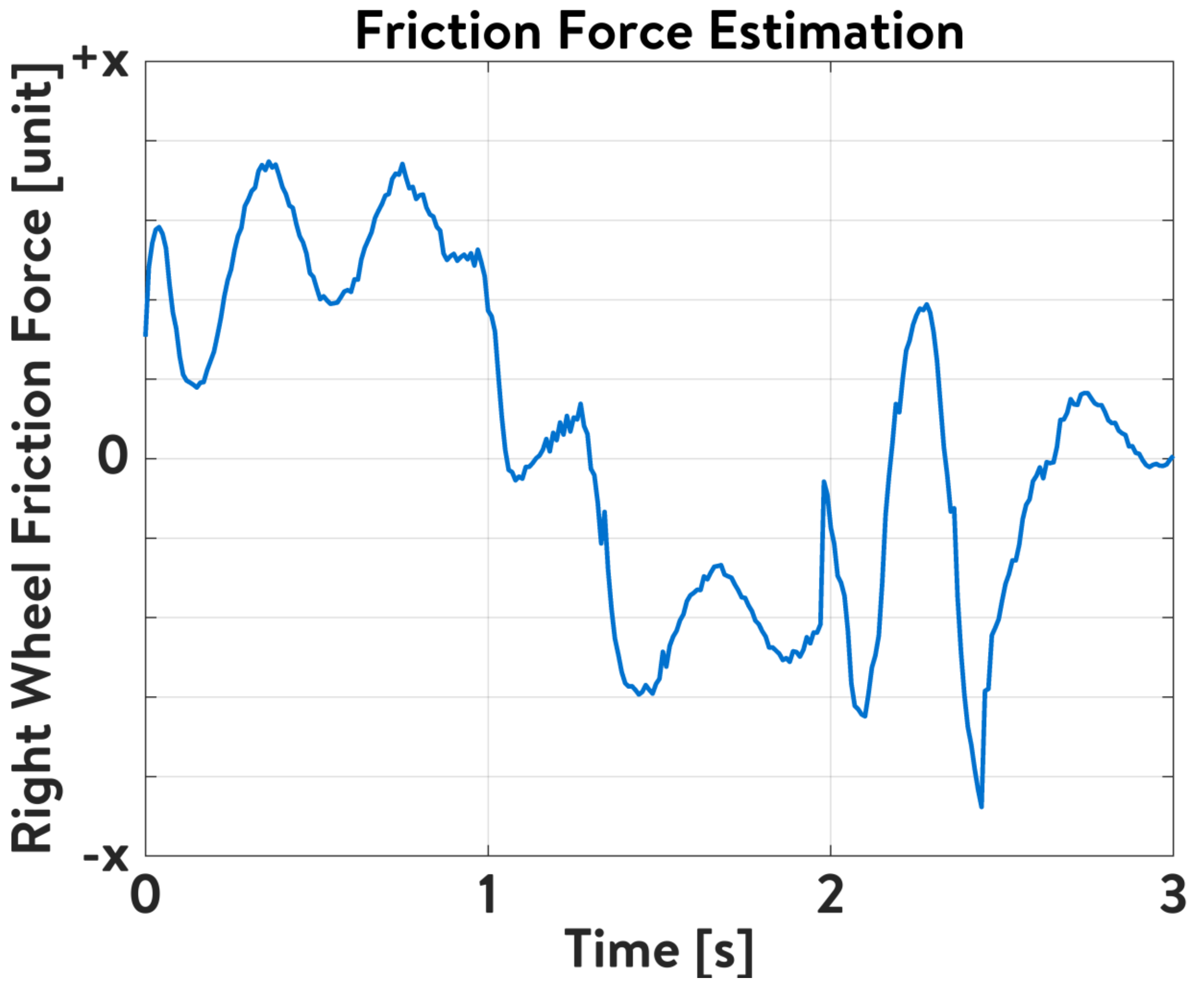


Slip: Force Estimation

- Maximum acceleration determined as a function of physical parameters
- Informs **context-specific limits** between chilled, ambient, and frozen storage zones

Maximum Torque Equation

$$\tau_{\max} = \frac{\mu_s g}{R}\left(\mathrm{mR}^2 + \mathrm{I}_{\mathrm{wy}}\right)$$



Parameter Estimation

- Estimates of bot inertia under varying tote weights
- Derived expected friction and motor parameters values
- More data needed across Alphabot fleet to gain a statistically significant average
- **Accurate parameters** inform **accurate physical limits** of the Alphabot

Tuned Parameter Deviation from Nominal

Bot Inertia	Wheel Inertia	Center of Mass	Torque Constant	Viscous Friction
2.5%	0.00%	-15.6%	-4.5%	-57.7%

Key Takeaways:

The ADL (generally) met performance specifications.

Future work is needed to resolve rotations and force estimation errors.

Leadership

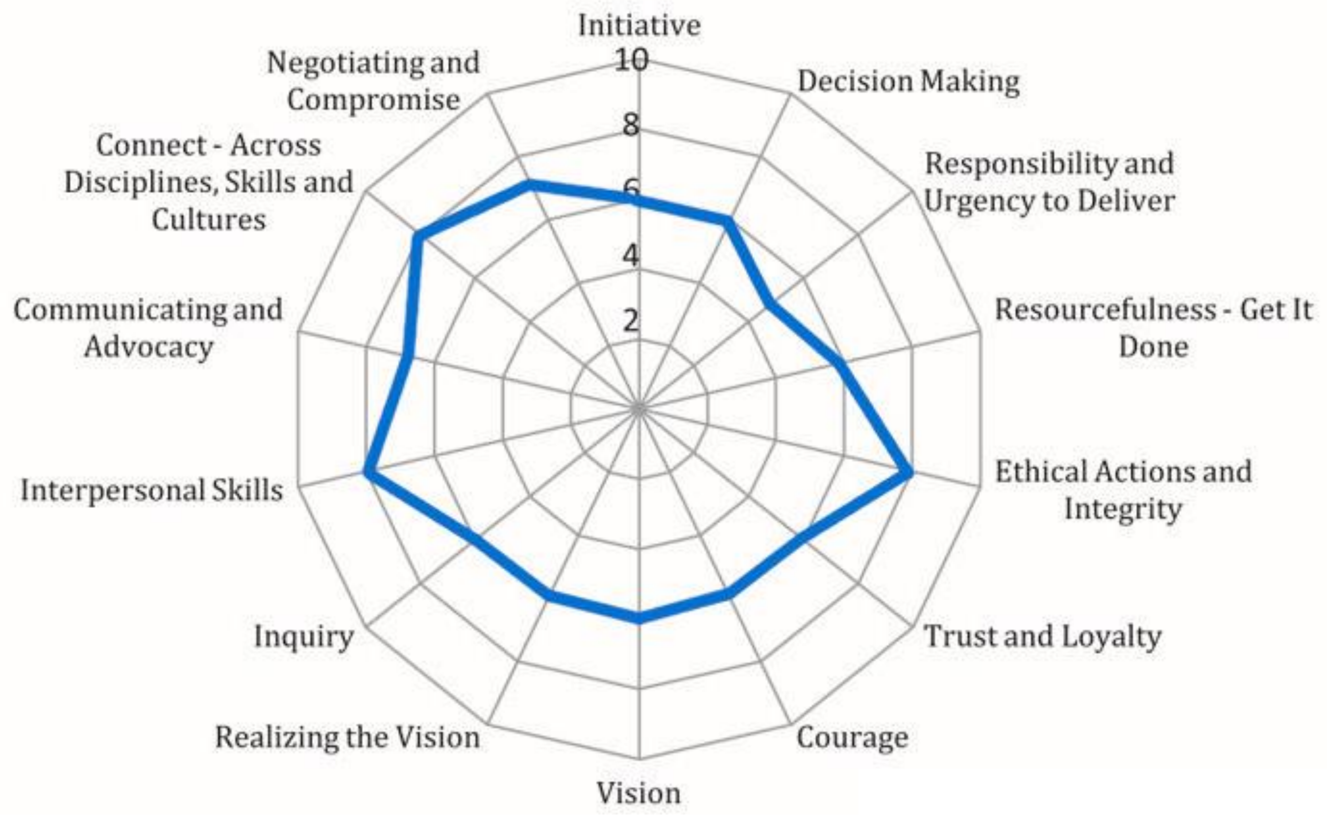
What Did We Learn?

Spider Chart: Pre-Test

- GIEL's 14 Leadership traits mapped with the **Spider Chart** [10]
- Pre-test completed at start of internship (9/1/2024)
- Improvement areas identified, action plan developed

Candidate's Key Leadership Qualities

Trait	Pre-Test Score
Resourcefulness	6
Responsibility	5
Vision	6



Candidate's Spider Chart Pre-Test

SOAR Story #1

Key Skills
Initiative
Communication
Negotiation

Key Frameworks
Influence without Authority [11]

S	The candidate **needed support** from WASR's reliability team to determine motor parameters. Meetings revealed **unusual** and **undesired** motor behavior.
O	Create a **mutually beneficial connection** across departments to advance the project.
A	**Hosted meetings** with user to define fault functionality. Negotiated with my ISA to include extra work in the project. **Designed a warning** in the Alphabot application that **detected unusual motor behavior**
R	Successfully characterized behavior using my warning. Inspired a meeting between **department** leads to discuss Alphabot's error detection capabilities. Gained the reliability team's support in determining parameters

SOAR Story #2

Key Skills
Vision
Communication
Negotiation
Responsibility

Key Frameworks
Getting to Yes
Eisenhower Matrix

S

The candidate wanted to use his software to implement model-based control schemes. Due to previous **schedule delays**, he **didn't have time** to develop controllers for his project

O

Negotiate a best-case scenario to keep controller design within the project scope.

A

Candidate used Fisher's *Getting to Yes* [13] to create **best-case** and **best-alternative** scenarios. He prepared and practiced his argument and presented to his advising team.

R

The candidate **realized the vision** and understood the importance of **developing core features** before increasing the project scope. He applied the **Eisenhower Matrix** [14] to identify important tasks.

SOAR Story #3

Key Skills
Initiative
Resourcefulness

Key Frameworks
Be-Know-Do [11]

S	The candidate needed to collect data for model tuning, but his tests **blocked other teams** from testing and required **additional support** from WASR's testing teams.
O	Develop the skills to test bots **without external support.**
A	The candidate studied WASR's **testing workflow** and solutions to common faults and errors. He **modified his schedule** to run tests during off-peak hours.
R	Candidate collected data **independently** without impeding WASR's workflow. New skills increased candidate's sense of **confidence** and **psychological safety.**

Spider Chart: Post-Test

- Peers confirmed the candidate's development in leadership skills

Trait	Pre	Post	ISA	CW
Resourcefulness	6	8	8	7
Responsibility	5	7	7	7
Initiative	6	10	10	8

Spider Chart Post Test

Conclusion

The candidate’s project took Controls team tasks offline, laid a foundation for MBC, and made the candidate a stronger engineer and leader.

Thank You!

Fletcher Cavanagh

cavanagh.f@northeastern.edu | fletchercavanagh.com

Q&A

30 Minutes

References

Alphabot.png (1280×720) (condyne.com)

https://talkbusiness.net/2023/05/walmart-opens-automated-market-fulfillment-center-in-bentonville/

https://www.researchgate.net/figure/Model-based-control-sequence_fig1_284280775

https://www.sciencedirect.com/topics/engineering/system-identification-approach