





Politecnico di Torino

Master's Degree in Mechanical Engineering

Contact Body Classification in Human–Robot Collaboration Using Domain Randomization

A Sim-to-Real Study on a Planar 3-RRR Parallel Robot

Master's Thesis

Sasan Alibabaei

Supervisor:	Prof. Giuseppe Bruno Averta
External Supervisor:	M. Sc. Jan Piosik
Academic Advisor:	Prof. Dr.-Ing. Thomas Seel

September 2026

Dedication

To my parents, thank you for always standing by me, believing in me, and supporting me through every stage of this journey. Whenever I felt tired, uncertain, or close to giving up, your words gave me the strength to keep going. Your patience, sacrifices, and unconditional love have made it possible for me to come this far. Whatever I achieve will always carry a part of everything you have given me.

To my brother, you have always been much more than a brother to me. You supported me in every way you could and never hesitated to be there whenever I needed help, encouragement, or simply someone to rely on. Knowing that I could always count on you made even the most difficult moments easier. I will always be grateful for everything you have done for me and for the confidence you have always placed in me.

I dedicate this thesis to you with all my heart.

Abstract

Safe physical interaction is a key requirement in human–robot collaboration. After detecting an external contact, the robot should also determine which part of its structure is involved so that an appropriate reaction can be selected. This thesis investigates the sim-to-real transfer of a seven-class contact-body classifier for a planar 3-RRR parallel robot.

All classifier training data are generated in simulation using MuJoCo and MATLAB. A feedforward neural network (FFNN) uses twelve physically interpretable features based on the external wrench, actuator torques, and the geometry of the external-force line of action. Physical data are not used for network training or feature alignment. Active- and passive-joint friction and damping are randomized around a reference model, while observation noise and actuator latency are studied separately. A common scenario bank, rollout-level data splitting, and a consistent observation model are used throughout the controlled experiments.

A dimensionless factor λ controls the width of the randomized ranges: larger λ values mean broader ranges around the reference model. The initial exploratory sweep tested $\lambda = 0.5$, 1, and 2, while physical ranges, observation noise, and maximum actuator latency were widened together. The average physical-benchmark accuracy increased across these three settings, so noise and latency were separated next at $\lambda = 2$. The controlled ablation showed that observation noise provided the clearest transfer gain, whereas the tested latency model did not improve the noise-enabled setting. Observation noise was therefore fixed and actuator latency was fixed at 0 ms before extending the physical ranges to $\lambda = 3$ and $\lambda = 4$. Neither wider range improved on $\lambda = 2$, so $\lambda = 2$ was retained for the final friction–damping ablation. Friction alone and damping alone did not improve the physical benchmark average, while their combination produced the strongest result.

The nominal simulation model achieves 63.51% and 70.87% accuracy on two independently recorded physical benchmark datasets, averaging 67.19%. The final friction-and-damping model with observation noise reaches 66.29% and 76.99%, increasing the average to 71.64%. The results show that careful design of the simulated source distribution and observation model can improve sim-to-real contact-body classification while keeping training entirely simulation-based. Unlike earlier work on the same robot using domain-adversarial neural networks (DANNs), the approach does not use physical target-domain data during training.

Keywords: Human–Robot Collaboration, contact-body classification, parallel robots, domain randomization, sim-to-real transfer, machine learning.

Contents

Abstract ii

List of Figures viii

List of Tables x

Abbreviations xi

1 Introduction 1

1.1 Human–Robot Collaboration and Safety 1

1.2 Contact Detection and Contact-Body Classification 1

1.3 Data-Driven Methods and the Need for Simulation 2

1.4 The Sim-to-Real Challenge 3

1.5 Domain Randomization 3

1.6 Research Gap 4

1.7 Objectives and Research Questions 4

1.8 Contributions of the Thesis 5

1.9 Structure of the Thesis 6

2 State of the Art 7

2.1 Contact Handling, Localisation, and Safety in HRC 7

2.2 Data-Driven Contact Classification 8

2.3 Transfer Learning and Sim-to-Real in Robotics 9

2.4 Domain Adaptation and Domain Randomization 10

2.5 Previous Work on the Investigated Parallel Robot 12

2.6 Research Gap and Positioning of This Thesis 12

3 Fundamentals **15**

3.1 The Planar 3-RRR Parallel Robot 15

3.2 Jacobian-Based Force and Wrench Mapping 16

3.2.1 Differential Mapping and Virtual Work 16

3.2.2 External-Force Estimation on the Physical Robot 17

3.3 Geometric Features for Contact-Body Classification 18

3.3.1 Force-Generated Moment 18

3.3.2 External-Force Line of Action 18

3.3.3 Distance Features 20

3.3.4 Angle Features 20

3.4 Final Physical Feature Representation 21

3.5 Feedforward Neural Networks 22

3.6 Classification and Evaluation Metrics 22

3.7 Domain Randomization 23

4 Methodology **24**

4.1 Overview of the Proposed Framework 24

4.2 Robot, Test Bench, and MuJoCo Model 25

4.3 Development of the Final Simulation and Feature Pipeline 26

4.4 Canonical Scenario Bank and Contact Generation 27

4.5 Reference XML and Domain Randomization Strategy 29

4.6 Observation Noise and Actuator Latency 31

4.7 Consistent Noisy-Feature Definition 31

4.8 Numerical Implementation of the Geometric Features 32

4.9 Final Feature Set 34

4.10 Rollout Sampling and Dataset Splitting 34

4.11 Neural Network Architecture and Training 34

4.12 Controlled Experimental Sequence 35
4.12.1 Noise–Latency Ablation 35
4.12.2 Controlled Physical-Range Extension 36
4.12.3 Final Friction–Damping Ablation 36
4.13 Evaluation Procedure 37

5 Results **39**

5.1 Headline Comparison 39
5.2 Initial Coupled Randomization-Strength Sweep 40
5.3 Noise–Latency Ablation 41
5.4 Controlled Physical-Range Extension 41
5.5 Final Friction–Damping Ablation and Selected Model 42

6 Discussion **45**

6.1 Interpretation of the Sim-to-Real Results 45
6.2 Influence of Feature Selection and Consistency 46
6.3 Influence of Domain Randomization 46
6.4 Comparison with Previous Work 47
6.5 Limitations 49
6.6 Future Work 49

7 Conclusion **51**

7.1 Summary and Main Findings 51
7.2 Answers to the Research Questions 52
7.3 Outlook 53

A Additional Dataset Information **54**

A.1 Canonical Class-Label Order 54
A.2 Scenario-Bank Summary 54

A.3 Controlled Experiment Summary 55

B Neural Network Hyperparameters **57**

B.1 Final Input Representation 57

B.2 Network Architecture 57

B.3 Training and Preprocessing Settings 58

C Additional Evaluation Results **59**

C.1 Full Aggregate Metric Tables 59

C.2 Class-Wise Effect of the Final Friction–Damping Condition 60

C.3 Confusion Matrices 60

References **64**

List of Figures

3.1 Definition of the force line of action and the associated geometric features used for contact-body classification. The figure illustrates the line of action, the minimum distance to the coupling joints, and the angle features. Reproduced from [2]. 20

4.1 Overview of the sim-to-real workflow. Training and model selection are performed on simulation data; the two physical datasets are used afterwards as fixed external benchmarks. 25

4.2 Physical planar 3-RRR test bench and corresponding MuJoCo model. Both domains represent the same seven contact-body classes used by the classifier. 26

4.3 Construction and validation of the canonical scenario bank. Removing invalid initial configurations and one singular rollout leaves 80,499 common rollout identities that are reused in every controlled experiment. 29

5.1 Headline accuracy comparison for the principal controlled models. Observation noise provides the clearest gain for the friction–damping model, while the tested noise–latency combination falls below the nominal baseline. 40

5.2 Final friction–damping ablation with observation noise fixed ON. Randomizing friction or damping alone does not improve the physical benchmark average, whereas their joint randomization produces the strongest result on both datasets. 43

6.1 Classification-accuracy comparison with the physical-to-physical FNN result of Mohammad et al. [2]. The present-thesis bars use the average of Bench Set 1 and Bench Set 2. 48

C.1 Row-normalized confusion matrices for the Base-XML nominal baseline and the final selected friction + damping + noise model on both physical benchmarks. A common 0–1 scale is used in all panels. 61

C.2 Row-normalized confusion matrices for the clean friction + damping condition on Bench Set 1 and Bench Set 2. The class-wise performance remains uneven without observation noise. 62

C.3 Row-normalized confusion matrices for the friction + damping + latency condition on Bench Set 1 and Bench Set 2. The tested latency changes some class-level errors but does not materially improve the average physical accuracy. 62

C.4 Row-normalized confusion matrices for the friction + damping + noise + latency condition on Bench Set 1 and Bench Set 2. The combined noise–latency setting produces broader errors than the selected noise-only model, consistent with its lower average accuracy. 63

List of Tables

2.1 Selected literature and its relation to the present thesis. The present approach differs from target-informed adaptation by keeping physical samples outside network training and feature alignment. 14

4.1 Main development decisions leading to the final controlled pipeline. These choices are fixed before the controlled comparisons reported in the Results chapter. 26

4.2 Base-XML values and modified-reference centre values for the parameters retained in the initial study. 30

4.3 Parameter ranges used in the initial coupled randomization-strength sweep. 30

4.4 Observation-noise and latency settings in the initial coupled λ-sweep. Both effects increase with λ, and all Gaussian noise is truncated at $\pm 3\sigma$. 31

4.5 Controlled rollout-level split used in all experiments. Samples from one rollout cannot appear in both training and validation subsets. 34

4.6 Preprocessing and training settings held fixed across the controlled experiments so that changes in physical performance are not caused by changes in the learning procedure. 35

4.7 Definitions of the controlled noise–latency ablation experiments. The design separates observation noise and latency while keeping the $\lambda = 2$ friction-and-damping ranges fixed. 36

4.8 Evaluation protocol. Physical data are used only for external benchmarking after simulation-side model selection. 37

5.1 Headline comparison of the principal controlled models on the two physical benchmarks. The friction + damping + noise condition gives the highest accuracy on both datasets and on average. 39

5.2 Physical accuracy in the initial coupled randomization-strength sweep. Physical parameter ranges, observation noise, and latency all scale with λ. 40

5.3 Controlled noise–latency ablation. Observation noise improves both physical benchmarks; the tested latency model provides little benefit alone and removes the gain when combined with noise. 41

5.4 Controlled physical-range extension with frozen observation noise and actuator latency fixed at 0 ms. Among the tested ranges, $\lambda = 2$ gives the strongest transfer on both physical benchmarks. 42

5.5 Final friction–damping ablation at $\lambda = 2$ with observation noise fixed ON and actuator latency fixed at 0 ms. Neither factor improves the physical average alone, while their combination gives the strongest result. 42

6.1 Contextual comparison with the two closest sim-to-real results reported by Giourgas [5]. The protocols and averaging procedures are not identical. 47

6.2 Comparison of the present simulation-trained models with the physical-to-physical FNN result reported by Mohammad et al. [2]. 48

A.1 Canonical class-label order used throughout the thesis. 54

A.2 Canonical common scenario bank used for all controlled experiments. . . 55

A.3 Descriptive summary of the controlled experiments used in the thesis. . 56

B.1 Neural-network architecture. 57

B.2 Controlled preprocessing and training settings. 58

C.1 Complete metrics for the initial coupled randomization-strength sweep. . 59

C.2 Complete metrics for the controlled noise–latency ablation at the $\lambda = 2$ friction-and-damping ranges. 59

C.3 Complete aggregate metrics for the controlled physical-range extension. Observation noise is frozen and actuator latency is fixed at 0 ms. 59

C.4 Complete aggregate metrics for the final friction–damping ablation. Observation noise is fixed ON and actuator latency is fixed at 0 ms for every condition. 60

C.5 Class-wise F1 changes from the friction/damping-off condition to the final friction + damping condition on the two physical benchmarks. 60

Abbreviations

COM	Centre of Mass
DANN	Domain-Adversarial Neural Network
DR	Domain Randomization
FFNN	Feedforward Neural Network
HRC	Human–Robot Collaboration
ML	Machine Learning
PKM	Parallel Kinematic Machine
Sim-to-Real	Transfer from simulation to the physical system

CHAPTER 1

Introduction

1.1 Human–Robot Collaboration and Safety

Human–Robot Collaboration (HRC) brings humans and robots into a shared workspace in order to combine human flexibility and decision-making with robotic precision, repeatability, and physical capability. HRC has long been recognised as an interdisciplinary field in which robot design, sensing, planning, learning, and safe control must be considered together [15, 16]. Removing the traditional physical separation between the human and the robot makes safe handling of unintended contact a central requirement. Collision monitoring therefore forms an important part of the safety concept of collaborative robotic systems [1, 17].

Physical interaction may be intentional, for example during guidance or handover tasks, or unintended as a result of an unexpected motion, positioning error, or collision. For an appropriate reaction, it is useful not only to detect that an external interaction has occurred, but also to determine where it acts and how strongly it affects the robot. The literature accordingly distinguishes between collision detection, isolation, and identification [1].

Parallel robots are attractive for collaborative applications because their actuators can be located close to the fixed base, reducing the moving mass of the mechanism. At the same time, their closed-loop kinematic structure makes the interpretation of external contacts more challenging because several chains are mechanically coupled through the mobile platform [2]. This motivates the contact-body classification problem considered in this thesis.

1.2 Contact Detection and Contact-Body Classification

Contact detection determines whether an external interaction is present, whereas contact localisation aims to determine where the interaction occurred. For the planar 3-RRR robot investigated here, localisation is formulated first as a seven-class contact-body classification problem: the classifier distinguishes the mobile platform and the two links of each

of the three kinematic chains.

Previous work on the same robot showed that the affected body can be classified from proprioceptive information by combining external-force estimates, actuator torques, and geometric descriptors derived from the line of action of the external force [2]. A later study by Giourgas addressed the same classification problem with transfer learning and, in particular, domain-adversarial neural networks (DANNs), for which target-domain information is used during domain alignment [5]. The present thesis does *not* repeat that transfer mechanism. It retains a conventional feedforward classifier and instead investigates whether the simulated source domain itself can be designed to transfer more reliably through reference-centred domain randomization, a realistic observation model, and controlled data generation. Physical samples are not used for feature alignment, network training, scaler fitting, or checkpoint selection.

The final input representation therefore contains external-wrench components, actuator torques, and force-line distance and angle features. Their mechanical interpretation is introduced in Chapter 3, while their final implementation is described in Chapter 4.

1.3 Data-Driven Methods and the Need for Simulation

Supervised machine-learning methods require labelled examples of the input quantities and their corresponding contact classes. Recording such datasets directly on a physical robot is possible, but large-scale data collection can be time-consuming, costly, and dependent on repeated interaction with the hardware. In robotics more generally, these limitations are an important motivation for simulation-based learning and benchmarking [19, 4, 22].

Simulation allows contact bodies, force directions, force magnitudes, robot configurations, and model parameters to be varied systematically while the corresponding class labels remain known. It also enables repeatable experiments and avoids repeatedly exposing the physical test bench to contact experiments. Previous work on the investigated platform likewise identified simulated data generation as a promising way to reduce the effort associated with real contact-data collection [5].

However, the availability of large simulated datasets does not by itself guarantee useful physical performance. A classifier may learn relationships that are highly predictive inside one simulator but do not remain valid when the same features are measured on the real robot.

1.4 The Sim-to-Real Challenge

The discrepancy between simulated and physical behaviour is commonly referred to as the sim-to-real gap. It can arise from differences in dynamics, sensing, actuation, control, contact modelling, and other approximations that separate a simulator from the physical system [4, 3, 22].

For the present contact-classification problem, relevant discrepancies include friction and damping errors, numerical contact behaviour, measurement noise, actuator timing, force-estimation errors, and differences between ideal simulated quantities and the signals available on the test bench. As a consequence, high classification accuracy on a simulation validation set does not imply comparable performance on physical measurements.

Sim-to-real transfer therefore requires more than increasing the capacity of the classifier. The simulated training distribution and the feature-generation pipeline must be constructed so that the network is encouraged to learn relationships that remain informative outside the nominal simulator.

1.5 Domain Randomization

Domain randomization addresses this problem by replacing training on a single deterministic simulator with training on a distribution of simulated environments. Selected parameters are varied during data generation so that the learned model is exposed to multiple plausible versions of the source domain [6, 7].

The principle has been demonstrated both for visual transfer and for robot dynamics. In dynamics randomization, quantities such as physical parameters, observation noise, and timing can be varied so that the resulting model becomes less dependent on one calibrated simulation instance [7]. Domain randomization is also commonly used in zero-shot sim-to-real settings in which training is performed entirely in simulation and the learned model is transferred directly to the physical system [3].

In this thesis, randomization is centred around a modified reference model of the planar 3-RRR robot, defined explicitly in Section 4.5. A dimensionless range-scaling factor λ controls how far the randomized parameter ranges extend around that reference centre: larger λ values correspond to broader ranges. Active- and passive-joint friction and damping form the effective physical randomization, while observation noise and actuator latency are investigated separately. Mechanical stiffness is not randomized. The exact parameterization of λ is given in Equation (4.5). The contact scenarios, class definition,

preprocessing, and training protocol are kept fixed across the final controlled comparisons.

1.6 Research Gap

Previous work established the feasibility of proprioceptive contact-body classification on the investigated parallel robot and showed that physically derived force-line features are informative for this task [2]. Subsequent work investigated transfer learning on the same platform and demonstrated that domain-adversarial methods can improve physical classification performance when target-domain information is incorporated into the transfer procedure [5].

The present work considers simulation-only model training. The physical bench datasets are excluded from network training, feature alignment, scaler fitting, and checkpoint selection. They are retained as fixed external benchmarks for comparing simulation-trained models after each model has been selected using simulation validation data. The central question is therefore how far the sim-to-real gap can be reduced by engineering the simulated source distribution itself.

In addition to the randomization ranges, this requires careful control of the experimental pipeline. The contact scenarios used by different models must be comparable, physically dependent features should be constructed consistently, and samples originating from the same rollout should not be distributed across both training and validation sets. The thesis therefore studies domain randomization together with these simulation- and data-design choices rather than treating randomization as an isolated parameter-tuning step.

1.7 Objectives and Research Questions

The main objective of this thesis is to develop and evaluate a simulation-based contact-body classifier for a planar 3-RRR parallel robot and to determine which simulation-side uncertainty sources improve its transfer to physical test-bench data. Particular attention is given to reference-centred friction and damping randomization, observation noise, actuator latency, and the width of the randomized physical ranges.

The work addresses the following research questions:

1. Can a contact-body classifier trained exclusively on randomized simulation data generalise to measurements from a physical planar 3-RRR parallel robot?

2. Which contact-related input features and simulation-design choices are most suitable for reducing the sim-to-real gap?
3. To what extent do reference-centred physical randomization and the tested observation model improve robustness compared with training on the Base-XML nominal simulation model?
4. Does the final classification pipeline maintain comparable performance across two independently recorded test-bench datasets?

1.8 Contributions of the Thesis

The main contributions of this thesis are:

- the development of a complete sim-to-real workflow for seven-class contact-body classification on a planar 3-RRR parallel robot using simulation-only network training;
- the construction of a canonical paired scenario bank and common validity mask that allow nominal and randomized experiments to be compared on exactly the same rollout identities;
- the definition of a twelve-dimensional physics-based input representation combining external-wrench components, actuator torques, force-line distances, and force-line angles, together with a consistent observation model that preserves the mechanical coupling between these quantities;
- the development of a rollout-aware preprocessing and dataset-splitting procedure that prevents samples from the same physical scenario from appearing in both training and validation subsets;
- a staged experimental study that first examines coupled randomization strength, then separates observation noise and actuator latency, extends the physical randomization range to $\lambda = 3$ and $\lambda = 4$, and finally isolates the effects of friction and damping;
- evaluation of all final models on two independently recorded physical benchmark datasets while keeping physical samples outside network training, feature alignment, scaler fitting, and checkpoint selection.

Taken together, the thesis provides a simulation-only transfer study for the same seven-class problem previously investigated with domain-adversarial adaptation. The distinc-

tion is deliberate: instead of adapting the learned representation with physical target-domain samples, the simulated source distribution and observation model are modified while the classifier remains an FFNN. The final selected configuration combines friction and damping randomization at $\lambda = 2$ with the consistent observation-noise model and raises the average physical accuracy from 67.19% for the Base-XML nominal baseline to 71.64%, with improvement on both physical benchmark datasets. The final ablation shows that this gain is associated with the joint action of friction and damping randomization rather than either factor alone.

1.9 Structure of the Thesis

Chapter 2 reviews contact handling and data-driven classification in HRC, transfer learning and sim-to-real methods, and the distinction between domain adaptation and domain randomization.

Chapter 3 presents the Jacobian-based force and wrench mappings used in the thesis, external-force estimation, geometric contact features, neural-network classification, and evaluation metrics.

Chapter 4 describes the simulation model, common scenario bank, domain-randomization strategy, observation model, feature construction, rollout-level preprocessing, classifier training, and evaluation procedure.

Chapter 5 reports the controlled nominal baseline, the initial coupled randomization-strength sweep, the observation-noise and actuator-latency ablation, the controlled $\lambda = 2, 3, 4$ physical-range extension, and the final friction–damping ablation on both physical test-bench datasets.

Chapter 6 interprets the resulting sim-to-real behaviour, examines the influence of the selected design choices, and discusses remaining limitations.

Finally, Chapter 7 summarises the main findings and provides recommendations for future work.

CHAPTER 2

State of the Art

This chapter reviews the research areas that directly support the problem studied in this thesis. The discussion begins with contact handling and localisation in human–robot collaboration, with particular attention to proprioceptive methods and parallel robots. It then reviews data-driven contact classification, transfer learning and sim-to-real transfer, and finally the distinction between domain adaptation and domain randomization. The chapter closes by positioning the present work with respect to previous studies on the same planar 3-RRR research platform.

2.1 Contact Handling, Localisation, and Safety in HRC

Human–robot collaboration requires a robot to operate safely despite uncertainty in human motion and in the surrounding environment. Early HRC surveys already identified safety, sensing, planning, and learning as tightly coupled design problems [15]. Kulic and Croft showed that online safety cannot be reduced to mechanical design alone: real-time planning and control must also react to changing collision risk [16]. For physical contact itself, the literature commonly distinguishes between *detection*, which determines whether an external interaction is present, *isolation* or localisation, which determines where it acts, and *identification*, which estimates quantities such as the contact force [1].

Contact information can be obtained through exteroceptive sensing, for example tactile skins or vision, or through proprioceptive quantities already available in the robot. Proprioceptive methods are attractive for whole-body monitoring because they do not require every potential contact surface to be covered by additional sensors. They instead infer external interaction from joint positions, actuator torques, motor currents, and model-based residuals or external-force estimates [1, 17]. The quality of such estimates is, however, tied to the accuracy of the robot model, friction compensation, sensing, and observer dynamics.

This issue becomes especially relevant for parallel robots. Their drives can be placed near the fixed base, which reduces moving mass and is attractive for collaborative operation, but their closed kinematic loops couple the response of several chains. Mohammad et

al. addressed these properties experimentally on the planar 3-RRR platform using kinetostatic transformations, impedance control, and momentum-observer-based contact detection. They demonstrated contact monitoring over the robot structure and reported detection delays in the range of 9–58 ms in their experiments [17]. These results establish the physical observer and contact-handling context on which the present classification task is built.

After contact detection, the location of the interaction must be resolved. For serial manipulators, Popov et al. combined joint-torque sensing with model-based reasoning for collision detection, localisation, and classification [8]. Manuelli and Tedrake proposed the contact particle filter, which represents uncertainty over possible contact locations using proprioceptive measurements [9]. Zwiener et al. showed that machine learning can also map proprioceptive measurements to contact location when an explicit analytical mapping is difficult to use robustly [10].

For the investigated parallel robot, Mohammad et al. formulated body classification as an intermediate step before continuous contact-point and force estimation [2]. The six links and the mobile platform form seven classes. Distances and angles derived from the line of action of the estimated external force, together with actuator-torque information, provide a physically interpretable way to distinguish these bodies despite the coupling of the parallel chains. This formulation is the direct mechanical basis of the classification problem studied in this thesis.

Contact classification is safety relevant because a wrong class can lead to an inappropriate reaction. Mohammad et al. later investigated classification uncertainty for collisions and clamping on a real parallel robot and explicitly distinguished dangerous from less dangerous misclassifications when the predicted class is used to select a reaction [18]. Their work reinforces that classification accuracy is not only a statistical objective; the reliability of the predicted contact state can affect the safety of the subsequent robot response.

2.2 Data-Driven Contact Classification

Data-driven methods are useful when the available mechanical relations contain important structure but are not sufficient to define a robust decision rule over all configurations and contact conditions. A neural classifier can combine several force-, torque-, and geometry-related quantities and learn nonlinear class boundaries from labelled examples. Deep-learning methods provide flexible nonlinear function approximation, but robotics also poses practical difficulties because training data from physical systems can be expensive and safety critical to collect [19].

For contact problems, purely black-box learning is not the only option. Robot-learning literature has repeatedly shown the value of combining mechanical knowledge with learned models. Nguyen-Tuong and Peters, for example, incorporated rigid-body dynamics as prior knowledge in semiparametric inverse-dynamics learning and showed that the learned correction can capture effects not represented by the nominal physical model [20]. Calandra et al. extended data-driven inverse-dynamics learning to contact-rich settings, emphasising that unknown contact locations, calibration errors, unmodelled dynamics, and sensor noise can substantially degrade purely analytical predictions [21]. Although these studies concern inverse dynamics rather than contact-body classification, they support the broader principle used here: physically meaningful quantities can structure the learning problem while the data-driven model absorbs relationships that are difficult to express analytically.

The feedforward classifier introduced by Mohammad et al. follows this hybrid philosophy directly. Its inputs are not raw simulator states but quantities derived from robot mechanics and force-line geometry [2]. This gives the feature representation a physical meaning that can, in principle, be reproduced in both simulation and experiment. More recent work at the same institute has also investigated temporal convolutional networks for contact detection and body classification, showing that temporal information can provide an additional source of discriminative structure [11]. The present thesis deliberately retains the simpler feedforward architecture so that changes in physical transfer can be attributed primarily to the simulation and observation pipeline rather than to a simultaneous change of classifier family.

A central distinction is therefore between *in-domain accuracy* and *transferability*. A classifier can reach very high validation accuracy when training and validation data are drawn from the same simulator distribution and still perform substantially worse on physical measurements. For the present problem, the quality of the training distribution and the consistency of the feature construction are consequently as important as the capacity of the network itself.

2.3 Transfer Learning and Sim-to-Real in Robotics

Transfer learning reuses knowledge acquired in one setting in order to improve performance in a related setting. Jaquier et al. organise robotic transfer around changes in the robot, task, and environment, and emphasise that useful transfer depends on the similarities and differences between the source and target settings [12]. The present thesis is primarily an environment transfer problem: the robot structure and seven-class task

remain the same, while the source environment is a MuJoCo simulation and the target environment is the physical test bench.

Simulation is attractive because it supports scalable, repeatable, and safe data generation. Large labelled datasets can be produced without repeated physical contact experiments, and parameters or contact cases can be varied systematically. These advantages are widely recognised in robot learning and sim-to-real research [19, 4]. They are particularly useful for the present application because the contact body, force direction, force magnitude, and robot configuration are known exactly during simulation.

The limitation is the *sim-to-real gap*. Zhao et al. discuss mismatches in sensing and actuation as key components of sim-to-real transfer [4]. The more recent survey by Aljalbout et al. provides a broader decomposition into dynamics, perception and sensing, actuation and control, and system design [22]. For a contact-rich parallel robot, relevant discrepancies include friction, damping, constraint and contact modelling, force-estimation errors, measurement noise, timing, actuator dynamics, and numerical effects. The target-domain feature distribution can therefore differ from the source even when the underlying robot geometry and classification task are nominally identical.

Several strategies address this gap. System identification attempts to make the simulated model more closely match measured physical behaviour. Domain adaptation uses target-domain information to reduce the discrepancy between source and target representations. Domain randomization instead trains over a distribution of simulated models so that success depends less strongly on one particular simulator instance. These strategies are complementary rather than mutually exclusive [4, 12, 22].

Transfer can also fail when the source encourages the model to exploit features that are not stable in the target domain. Jaquier et al. refer to this possibility as negative transfer [12]. This is particularly relevant here because highly predictive simulator-specific cues can make the simulated classification problem easier while making the learned decision boundary less useful on the physical robot. The selection of the input representation and the design of the source distribution are therefore central elements of the present study.

2.4 Domain Adaptation and Domain Randomization

Domain adaptation and domain randomization address the same general problem of distribution shift but use different information. A prominent domain-adaptation method is the domain-adversarial neural network (DANN) of Ganin et al. [13]. Its feature extractor is trained both to predict the source task labels and to make source and target samples

difficult for a domain discriminator to distinguish. Consequently, target-domain samples participate in the learning procedure even when their task labels are not required.

Giourgas applied this principle to contact-body classification on the same planar 3-RRR research platform [5]. The study compared a conventional FFNN, DANN, and multi-domain variants and showed that domain-adversarial alignment improved physical classification over the corresponding simulation-trained FFNN. It also investigated sim/real-to-real cases in which labelled physical data were included in training. This line of work demonstrates that target-informed transfer is effective for the investigated task, but it answers a different question from the one considered here.

Domain randomization changes the source distribution instead of aligning source features with target samples. Tobin et al. demonstrated the principle for visual synthetic-to-real transfer by training over strongly varied rendered scenes [6]. Peng et al. extended the idea to robot dynamics, randomising physical and timing-related parameters and showing that observation noise and action timing can be important for successful physical transfer [7]. Chen et al. formalised domain randomization as learning over a family of simulated MDPs with latent environment parameters, providing a theoretical view of why training over diverse simulators can improve robustness [3].

A practical difficulty is selecting both the parameters and the width of their distributions. Too little variation may fail to cover the physical discrepancy, while excessive variation can make the source distribution unnecessarily difficult or physically implausible. Adaptive methods such as Bayesian Domain Randomization use sparse real-world feedback to tune the randomization distribution [14]. Such methods can be data efficient, but they no longer represent a purely simulation-defined source distribution.

The present thesis uses a controlled, reference-centred randomization scheme instead. Friction and damping parameters, observation noise, and actuator latency are varied in paired experiments while contact scenarios, feature definitions, preprocessing, and training remain fixed. The network is trained only on simulation data. Bench Set 1 and Bench Set 2 do not enter gradient-based training, feature alignment, scaler fitting, or checkpoint selection, although their repeated evaluation informs the sequence of controlled simulation experiments. The study should therefore be interpreted as simulation-only network training with fixed external physical benchmarks rather than as an untouched one-shot zero-shot benchmark.

2.5 Previous Work on the Investigated Parallel Robot

The present thesis follows a sequence of studies on safe contact handling and learning on the same research platform. Mohammad et al. first established the experimental basis for low-impedance operation, external-force observation, and whole-structure contact detection on the planar 3-RRR robot [17]. Their subsequent work on collision isolation and identification introduced the seven-body classification problem and the force-line distance and angle features used in this thesis [2]. A later study examined the uncertainty of data-driven contact classifications and showed why misclassification reliability matters when the predicted class selects a physical reaction [18].

Giourgas then investigated transfer learning for contact-body classification on this platform [5]. The important methodological distinction is that the main DANN experiments use physical target-domain samples during adversarial feature alignment, and the Sim/Real-to-Real experiments additionally include labelled physical training data. In contrast, the present thesis keeps the classifier as a conventional FFNN and asks how much transfer can be obtained by improving the simulated source distribution and observation model without inserting physical samples into the network-learning pipeline.

The difference is not merely architectural. The present work also uses a fixed bank of contact scenarios, a common validity mask, consistent feature reconstruction from one feature wrench, and a rollout-level train–validation split. These choices are intended to ensure that changes in physical performance can be attributed to the tested source distribution rather than to changes in scenarios or leakage of related samples between training and validation.

2.6 Research Gap and Positioning of This Thesis

The literature establishes three points that motivate this thesis. First, proprioceptive sensing and physically derived force-line features can support whole-structure contact localisation on the investigated parallel robot [17, 2]. Second, data-driven contact models can benefit from physically meaningful representations while still being sensitive to modelling and sensing errors [20, 21]. Third, transfer-learning methods can compensate source–target mismatch either by using target data for adaptation or by broadening the simulated source domain [13, 6, 7, 12].

Previous work on the same robot demonstrated the value of target-informed DANN transfer [5]. What remains less clear is how far a plain feedforward contact-body classifier can

be transferred when the network itself is trained only on simulation and the emphasis is placed on constructing the source data carefully. This is the research gap addressed here.

Accordingly, the thesis does not propose a new general transfer-learning algorithm. It provides a controlled sim-to-real study of contact-body classification in which nominal and randomized datasets reuse the same scenario bank, related samples remain together through rollout-level splitting, and friction-and-damping randomization is separated from observation noise and actuator latency. The resulting models are evaluated on two independently recorded physical datasets. This design makes it possible to study which simulation-side choices actually improve transfer and which merely preserve high in-domain accuracy.

For orientation, Table 2.1 summarises the studies most directly related to the present work and highlights the distinction between target-informed adaptation and the simulation-only approach used here.

Table 2.1: Selected literature and its relation to the present thesis. The present approach differs from target-informed adaptation by keeping physical samples outside network training and feature alignment.

Study	Core approach	Real/target data used in learning?	Relation to this thesis
Mohammad et al. [2]	Physics-based force-line features with an FFNN for contact-body classification, followed by contact localisation.	Yes	Defines the seven-class problem and the distance/angle features used as the mechanical basis of this work.
Ganin et al. [13]	Domain-adversarial learning for source–target feature alignment.	Yes, unlabelled target data	Provides the principal domain-adaptation reference and a contrast to simulation-only network training.
Tobin et al. [6]	Visual domain randomization for synthetic-to-real transfer.	No	Establishes the idea that deliberately broad source variation can promote direct transfer.
Peng et al. [7]	Dynamics randomization including physical parameters, observations, and timing effects.	No	Motivates the separation of friction-and-damping randomization, observation noise, and actuator latency.
Giourgas [5]	FFNN and DANN-based sim-to-real contact-body classification on the same research platform.	DANN uses target-domain data	Direct predecessor for the same classification task; provides the most relevant numerical comparison for the present work.
Present thesis	**FFNN with reference-centred domain randomization, consistent feature reconstruction, and controlled observation modelling.**	**No physical samples in network training or feature alignment**	**Studies how far the same task can be transferred by improving the simulated source distribution rather than adapting the network to target data.**

CHAPTER 3

Fundamentals

This chapter introduces only the mechanical and machine-learning relations that are used later in the thesis. Detailed direct- or forward-kinematics derivations are omitted because they are not solved as part of the proposed classification pipeline. Instead, the focus is on the Jacobian-based mappings required to relate an external contact force to platform wrenches and actuator torques, the observer used on the physical test bench, and the force-line geometry from which the classifier features are constructed. The mechanical formulation follows the modelling framework developed for the investigated parallel robot in [17, 2].

3.1 The Planar 3-RRR Parallel Robot

The investigated mechanism is a planar 3-RRR parallel robot composed of three kinematic chains connecting a fixed base to a common mobile platform. Each chain contains one actuated revolute joint, one passive revolute joint, and one revolute coupling joint at the mobile platform. The mechanism has three operational degrees of freedom: two translations and one rotation in the plane.

The mobile-platform pose is represented by

$$\boldsymbol{x} = \begin{bmatrix} x_E & y_E & \varphi_E \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^3, \tag{3.1}$$

and the three active-joint coordinates are collected in

$$\boldsymbol{q}_a = \begin{bmatrix} q_{a,1} & q_{a,2} & q_{a,3} \end{bmatrix}^{\mathsf{T}}. \tag{3.2}$$

The passive and coupling joints are required internally for the robot model and for evaluating contact-point geometry, but their explicit direct-kinematics solution is not part of the classification method. For the present thesis, the relevant mechanical quantity is the local Jacobian that relates differential motion and generalized forces at the current robot configuration.

3.2 Jacobian-Based Force and Wrench Mapping

3.2.1 Differential Mapping and Virtual Work

At a valid robot configuration, the differential relation between active-joint velocity and mobile-platform velocity is written as

$$\dot{\boldsymbol{x}} = \boldsymbol{J}_{x,q_a}\dot{\boldsymbol{q}}_a, \tag{3.3}$$

where $\boldsymbol{J}_{x,q_a} \in \mathbb{R}^{3\times 3}$ is the configuration-dependent platform Jacobian. For a point C on a robot body, corresponding contact-point Jacobians can be used to relate the velocity of that point either to the platform coordinates or directly to the active joints,

$$\dot{\boldsymbol{x}}_C = \boldsymbol{J}_{x_C,x}\dot{\boldsymbol{x}} = \boldsymbol{J}_{x_C,q_a}\dot{\boldsymbol{q}}_a. \tag{3.4}$$

The exact form of these matrices is evaluated by the robot model and depends on the current configuration, contacted body, and contact location. Their detailed derivation is not required for the later learning pipeline; their role here is to define how the same physical contact appears in platform and actuator coordinates.

For the planar robot, a generalized external wrench at the mobile platform is written as

$$\boldsymbol{F}_{\text{ext}} = \begin{bmatrix} F_x & F_y & M_z \end{bmatrix}^{\mathsf{T}}. \tag{3.5}$$

Using the principle of virtual work, a force or wrench acting at a contact point can be projected to platform and actuator coordinates as

$$\boldsymbol{F}_{\text{ext,mP}} = \boldsymbol{J}_{x_C,x}^{\mathsf{T}}\boldsymbol{F}_{\text{ext,link}}, \tag{3.6}$$

and

$$\boldsymbol{\tau}_{a,\text{ext}} = \boldsymbol{J}_{x_C,q_a}^{\mathsf{T}}\boldsymbol{F}_{\text{ext,link}}. \tag{3.7}$$

When the external wrench has already been expressed in platform coordinates, the corresponding actuator-torque contribution is

$$\boldsymbol{\tau}_{a,\text{ext}} = \boldsymbol{J}_{x,q_a}^{\mathsf{T}}\boldsymbol{F}_{\text{ext,mP}}. \tag{3.8}$$

These relations are directly relevant to the classifier because the same contact can produce different actuator-torque patterns at different robot configurations, while the force-line geometry provides additional information about which body is intersected by the external interaction.

3.2.2 External-Force Estimation on the Physical Robot

On the physical test bench, the externally applied wrench is not known as a ground-truth simulation input. It is estimated from the robot motion and model using a generalized-momentum observer. In operational coordinates, the robot dynamics can be written in the compact form

$$\boldsymbol{M}_x(\boldsymbol{x})\ddot{\boldsymbol{x}} + \boldsymbol{c}_x(\boldsymbol{x}, \dot{\boldsymbol{x}}) + \boldsymbol{g}_x(\boldsymbol{x}) + \boldsymbol{F}_{\mathrm{fr},x} = \boldsymbol{F}_m + \boldsymbol{F}_{\mathrm{ext}}, \tag{3.9}$$

where $\boldsymbol{M}_x$ is the operational-space inertia matrix, $\boldsymbol{c}_x$ contains centrifugal and Coriolis terms, $\boldsymbol{g}_x$ contains gravitational effects, $\boldsymbol{F}_{\mathrm{fr},x}$ represents friction, and $\boldsymbol{F}_m$ is the motor-generated generalized wrench.

With generalized momentum $\boldsymbol{p}_x = \boldsymbol{M}_x\dot{\boldsymbol{x}}$, the observer used for the platform can be represented as

$$\hat{\boldsymbol{F}}_{\mathrm{ext}} = \boldsymbol{K}_o \left[\hat{\boldsymbol{M}}_x\dot{\boldsymbol{x}} - \int_0^t \left(\boldsymbol{F}_m - \hat{\boldsymbol{\beta}} + \hat{\boldsymbol{F}}_{\mathrm{ext}} \right) \mathrm{d}\tilde{t} \right], \tag{3.10}$$

where $\boldsymbol{K}_o$ is the observer-gain matrix and $\hat{\boldsymbol{\beta}}$ collects modelled gravitational, frictional, and velocity-dependent terms. Under sufficiently accurate model assumptions, its error dynamics are approximately first order,

$$\boldsymbol{K}_o^{-1}\dot{\hat{\boldsymbol{F}}}_{\mathrm{ext}} + \hat{\boldsymbol{F}}_{\mathrm{ext}} = \boldsymbol{F}_{\mathrm{ext}}. \tag{3.11}$$

This observer is important for sim-to-real transfer because the physical classifier does not receive the ideal force known to the simulator. Model errors, friction, sensing, filtering, and observer dynamics modify the measured feature distribution. The observation model introduced in Chapter 4 is intended to make the simulation-side features less ideal while preserving their mechanical relationships.

3.3 Geometric Features for Contact-Body Classification

The closed-loop structure of a parallel robot prevents the contacted body from being determined solely by identifying the last affected actuator. A force applied to one chain can influence multiple actuators through the mobile platform. For this reason, the classification method uses geometric features derived from the line of action of the external force [2].

3.3.1 Force-Generated Moment

For the planar mechanism, the force and its point of application can be embedded in three-dimensional coordinates as

$$\boldsymbol{f} = \begin{bmatrix} F_x & F_y & 0 \end{bmatrix}^{\mathsf{T}}, \qquad \boldsymbol{r} = \begin{bmatrix} r_x & r_y & 0 \end{bmatrix}^{\mathsf{T}}. \tag{3.12}$$

The moment generated by the force is

$$\boldsymbol{m} = \boldsymbol{r} \times \boldsymbol{f} = \begin{bmatrix} 0 & 0 & M_z \end{bmatrix}^{\mathsf{T}}, \tag{3.13}$$

with

$$M_z = r_x F_y - r_y F_x. \tag{3.14}$$

Equation (3.14) shows that an out-of-plane moment can arise naturally from a planar force applied at a non-zero lever arm. This induced moment must be distinguished from an independently imposed external moment. The treatment of M_z in the final simulation pipeline is described in Chapter 4.

3.3.2 External-Force Line of Action

The geometric construction requires an equivalent planar wrench at the mobile platform. On the physical test bench this quantity is obtained from the external-force observer, whereas the simulation pipeline uses the corresponding mapped feature wrench. The same line-of-action construction applies in both cases. Using hats to denote the estimated case, the wrench is written as

$$\hat{\boldsymbol{F}}_{\text{ext}} = \begin{bmatrix} \hat{\boldsymbol{f}}^{\mathsf{T}} & \hat{M}_z \end{bmatrix}^{\mathsf{T}}. \tag{3.15}$$

The moment generated by a lever arm $\boldsymbol{r}$ satisfies

$$\hat{\boldsymbol{m}} = \boldsymbol{r} \times \hat{\boldsymbol{f}} = \boldsymbol{S}^{\mathsf{T}}(\hat{\boldsymbol{f}})\boldsymbol{r}, \tag{3.16}$$

where $\boldsymbol{S}(\cdot)$ denotes the skew-symmetric cross-product matrix.

A point of minimum distance between the mobile-platform coordinate origin and the force line of action can be obtained using the Moore–Penrose pseudoinverse:

$$\boldsymbol{r}_{\text{mP,LoA}} = \left[\boldsymbol{S}^{\mathsf{T}}(\hat{\boldsymbol{f}})\right]^{\dagger} \hat{\boldsymbol{m}}. \tag{3.17}$$

The normalized force direction is

$$\hat{\boldsymbol{n}}_f = \frac{\hat{\boldsymbol{f}}}{\left\|\hat{\boldsymbol{f}}\right\|_2}. \tag{3.18}$$

The line of action is then parameterized by a scalar s as

$$\boldsymbol{r}_{\text{LoA}}(s) = \boldsymbol{r}_{\text{mP,LoA}} + s\hat{\boldsymbol{n}}_f, \qquad s \in \mathbb{R}. \tag{3.19}$$

The scalar s is used here to avoid confusion with the domain-randomization range factor λ introduced later in the thesis; the reproduced notation in Figure 3.1 follows the original source.

For clarity, the geometric interpretation of the line of action and the resulting distance and angle features is illustrated in Figure 3.1.

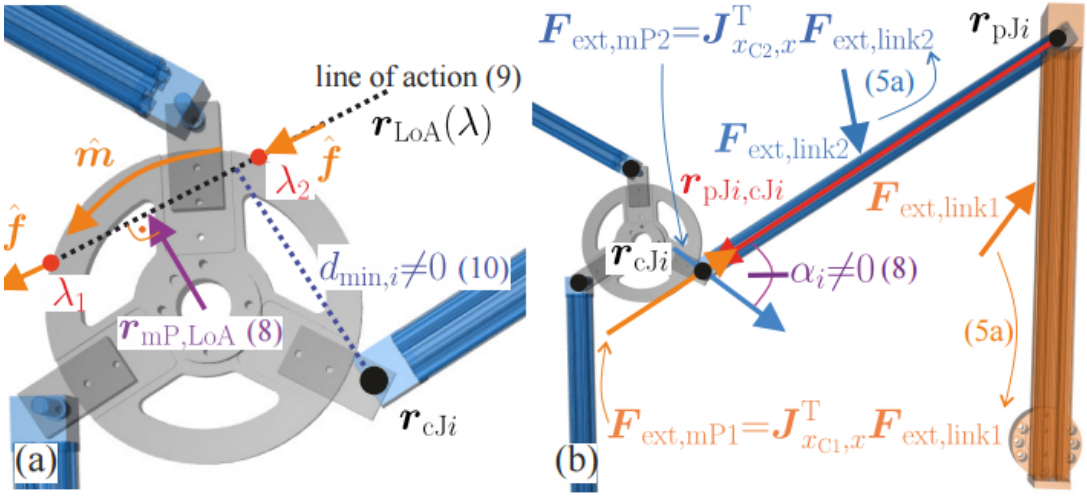


Figure 3.1: Definition of the force line of action and the associated geometric features used for contact-body classification. The figure illustrates the line of action, the minimum distance to the coupling joints, and the angle features. Reproduced from [2].

3.3.3 Distance Features

Let $\boldsymbol{r}_{cJ_i}$ denote the position of the coupling joint of chain i. The perpendicular distance between the coupling joint and the external-force line of action is

$$d_i = \left\| (\boldsymbol{r}_{cJ_i} - \boldsymbol{r}_{\mathrm{mP,LoA}}) \times \hat{\boldsymbol{n}}_f \right\|_2 , \qquad i \in \{1, 2, 3\}. \tag{3.20}$$

The three distance features are collected in

$$\boldsymbol{d} = \begin{bmatrix} d_1 & d_2 & d_3 \end{bmatrix}^{\mathsf{T}}. \tag{3.21}$$

For a force transmitted through chain i, the projected force line passes close to the corresponding coupling joint. Therefore, d_i can approach zero. A large concentration of near-zero distance values is thus physically plausible and does not necessarily indicate an error in the feature calculation.

3.3.4 Angle Features

Let

$$\boldsymbol{a}_{2,i} = \boldsymbol{r}_{cJ_i} - \boldsymbol{r}_{pJ_i} \tag{3.22}$$

denote the vector from the passive joint to the coupling joint of chain i. The geometric feature used in the implementation is the unsigned angle between the external-force direction and this forearm vector:

$$\alpha_i = \operatorname{atan2}\left(\left\|\hat{\boldsymbol{n}}_f \times \boldsymbol{a}_{2,i}\right\|_2, \hat{\boldsymbol{n}}_f^{\mathsf{T}} \boldsymbol{a}_{2,i}\right), \qquad i \in \{1,2,3\}. \tag{3.23}$$

The MATLAB implementation evaluates this expression with `atan2d`. Consequently, the exported angle is expressed in degrees and lies in the interval

$$\alpha_i \in [0^\circ, 180^\circ]. \tag{3.24}$$

Thus, parallel force and forearm directions correspond to $\alpha_i \approx 0^\circ$, antiparallel directions to $\alpha_i \approx 180^\circ$, and perpendicular directions to $\alpha_i \approx 90^\circ$.

The complete angle vector is

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \end{bmatrix}^{\mathsf{T}}. \tag{3.25}$$

Together with the three line-of-action distances, these angles describe how the external-force direction is oriented relative to the individual kinematic chains. The exact numerical safeguards and the corresponding MATLAB implementation are documented in Section 4.8.

The line-of-action, distance, and angle features were originally derived through the kinetostatic analysis presented in [2].

3.4 Final Physical Feature Representation

The final classifier uses twelve physically motivated input features. They are collected in the vector

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{F}_{\text{ext}}^{\mathsf{T}} & \boldsymbol{\tau}_{a,\text{ext}}^{\mathsf{T}} & \boldsymbol{d}^{\mathsf{T}} & \boldsymbol{\alpha}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^{12}. \tag{3.26}$$

Here, $\boldsymbol{F}_{\text{ext}} = [F_x, F_y, M_z]^{\mathsf{T}}$, $\boldsymbol{\tau}_{a,\text{ext}} = [\tau_1, \tau_2, \tau_3]^{\mathsf{T}}$, $\boldsymbol{d} = [d_1, d_2, d_3]^{\mathsf{T}}$, and $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \alpha_3]^{\mathsf{T}}$.

The input vector combines three complementary sources of information:

- the equivalent external wrench (F_x, F_y, M_z);

- the actuator response (τ_1, τ_2, τ_3);
- the geometric relationship between the force line of action and the three kinematic chains, represented by $(d_1, d_2, d_3, \alpha_1, \alpha_2, \alpha_3)$.

Unlike an input representation based directly on the end-effector position, these quantities describe the physical effect of the contact on the robot. The motivation for excluding the end-effector pose and robot scale from the final classifier is discussed in Chapter 4.

3.5 Feedforward Neural Networks

A feedforward neural network maps the input feature vector $\boldsymbol{z}$ to seven output scores, one for each possible contact body. For hidden layer l, the forward computation is

$$\boldsymbol{h}^{(l)} = \sigma\left(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}\right), \tag{3.27}$$

where $\boldsymbol{W}^{(l)}$ and $\boldsymbol{b}^{(l)}$ are trainable weights and biases, and $\sigma(\cdot)$ is a nonlinear activation function. The first hidden-layer input is

$$\boldsymbol{h}^{(0)} = \boldsymbol{z}. \tag{3.28}$$

The final layer generates the logits

$$\boldsymbol{o} = \boldsymbol{W}^{(\mathrm{out})}\boldsymbol{h}^{(L)} + \boldsymbol{b}^{(\mathrm{out})}, \tag{3.29}$$

and the predicted contact class is

$$\hat{y} = \underset{k\in\{0,\ldots,6\}}{\arg\max}\, o_k. \tag{3.30}$$

The specific architecture, number of neurons, activation functions, optimizer, and training procedure used in this thesis are presented in Chapter 4.

3.6 Classification and Evaluation Metrics

The classifier distinguishes seven contact bodies:

$$y \in \{0, 1, 2, 3, 4, 5, 6\}. \tag{3.31}$$

The overall classification accuracy is

$$\text{Accuracy} = \frac{\text{number of correctly classified samples}}{\text{total number of samples}}. \tag{3.32}$$

Because overall accuracy alone may hide differences between classes, the evaluation also considers class-wise precision, recall, F1-score, and confusion matrices. These metrics are reported separately for the simulation validation set, Bench Set 1, and Bench Set 2.

3.7 Domain Randomization

A deterministic simulation represents the robot using a single set of model parameters. Domain randomization instead treats the simulation parameters as random variables. Let

$$\boldsymbol{\theta} \sim p_{\mathrm{DR}}(\boldsymbol{\theta}) \tag{3.33}$$

denote a parameter vector sampled from the selected randomization distribution. Each realization of $\boldsymbol{\theta}$ defines a different but physically plausible simulated robot.

The classifier parameters $\boldsymbol{w}$ are learned by minimizing the expected classification loss over the distribution of randomized simulations:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \mathbb{E}_{\boldsymbol{\theta} \sim p_{\mathrm{DR}}} \left[\mathbb{E}_{(\boldsymbol{z}, y) \sim \mathcal{D}(\boldsymbol{\theta})} \left[\mathcal{L}\big(f_{\boldsymbol{w}}(\boldsymbol{z}), y\big) \right] \right]. \tag{3.34}$$

The objective is not to reproduce the physical robot perfectly with one simulator. Instead, the classifier is exposed to a range of plausible simulation conditions so that it becomes less dependent on errors in any individual model.

In this thesis, the randomization is centred around the modified reference model defined in Section 4.5. This reference centre is distinct from the Base-XML nominal baseline used for comparison. The selected parameters, distributions, and numerical ranges are described in Chapter 4.

CHAPTER 4

Methodology

This chapter describes the controlled simulation, preprocessing, training, and evaluation pipeline used in this thesis. The final experimental design was constructed so that nominal and randomized models are compared using the same contact scenarios, rollout identities, data split, feature definition, and training protocol. The chapter first introduces the overall workflow and then describes the robot and MuJoCo model, the canonical scenario bank, the reference-centred randomization scheme, the observation model, rollout-aware preprocessing, and the controlled ablation studies.

4.1 Overview of the Proposed Framework

The complete workflow consists of six stages:

1. generation of an immutable bank of contact scenarios;
2. validation of the scenarios and construction of one common rollout mask;
3. simulation with either the nominal or a randomized MuJoCo model;
4. construction of the twelve physics-based input features;
5. rollout-aware sampling, splitting, and standardization;
6. FFNN training in simulation followed by evaluation on the two physical benchmark datasets.

Figure 4.1 summarises the workflow. The physical benchmarks enter only after simulation-side preprocessing, training, and checkpoint selection. The exact role of each dataset is stated in one place in Section 4.13.

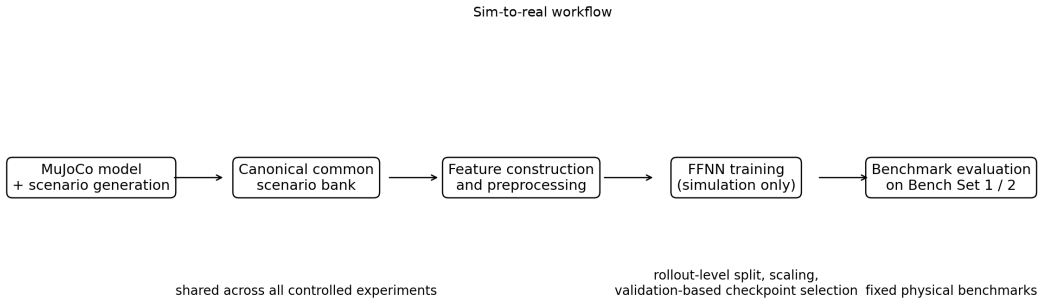


Figure 4.1: Overview of the sim-to-real workflow. Training and model selection are performed on simulation data; the two physical datasets are used afterwards as fixed external benchmarks.

4.2 Robot, Test Bench, and MuJoCo Model

The investigated system is the planar 3-RRR parallel robot introduced in Chapter 3. The classification problem contains seven contact-body classes: the mobile platform and the two links of each of the three kinematic chains. Two independently recorded physical datasets are used as fixed physical benchmarks and are denoted Bench Set 1 and Bench Set 2. Both are processed using the same final feature definition as the simulated data so that the learned classifier can be evaluated without changing its input representation.

MuJoCo is used to simulate the controlled response of the same robot to externally applied planar contact forces. The simulation time step is $1\,\mathrm{ms}$. Before every rollout, the MuJoCo state is fully reset so that state, contact, force, or actuator history from one scenario cannot influence the next. The final scenario definition uses a fixed robot scale of 1.0. The contact force is applied at the centre of mass of the selected contact body, and no independently sampled physical moment is directly applied. A non-zero equivalent platform moment can still arise naturally from the force and its lever arm according to Equation (3.14).

The physical test bench and the corresponding MuJoCo representation are shown in Figure 4.2.

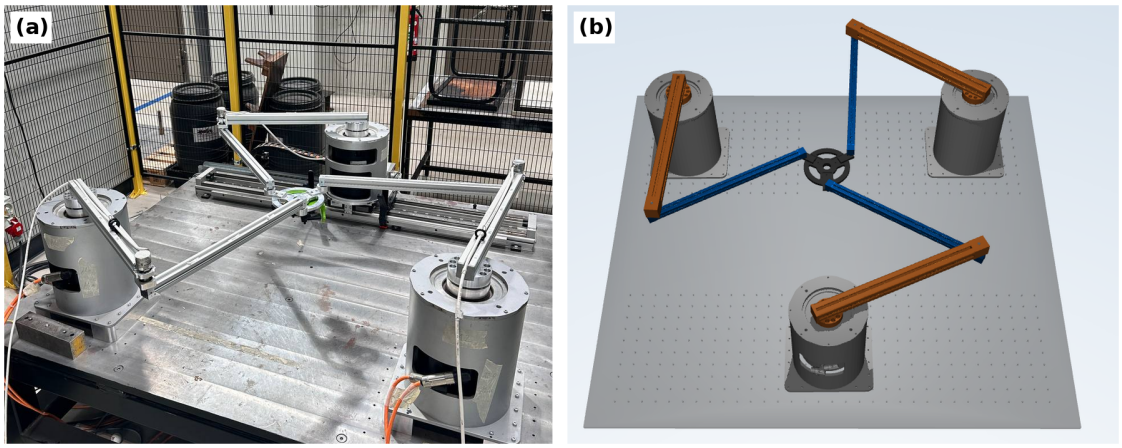


Figure 4.2: Physical planar 3-RRR test bench and corresponding MuJoCo model. Both domains represent the same seven contact-body classes used by the classifier.

4.3 Development of the Final Simulation and Feature Pipeline

The final controlled pipeline was reached through a sequence of method-development investigations. High validation accuracy in simulation did not guarantee transfer to the physical test bench, so the development focused on removing simulator-specific cues, preserving physical consistency between exported features, and making comparisons between datasets reproducible. Not every intermediate version was preserved as a fully paired ablation; the choices in Table 4.1 are therefore reported as methodological development decisions, while quantitative claims are restricted to the controlled experiments in Chapter 5.

Table 4.1: Main development decisions leading to the final controlled pipeline. These choices are fixed before the controlled comparisons reported in the Results chapter.

Aspect	Investigated issue	Final controlled choice
Robot scale	Scale randomization was tested during development but did not provide a consistent improvement in physical-benchmark accuracy.	Scale fixed to 1.0 and excluded from the classifier input.
End-effector pose	Including the end-effector pose as a classifier input did not provide a consistent improvement in physical-benchmark accuracy.	Pose removed from the classifier input; retained only where required internally for geometric calculations.

Continued on next page

Table 4.1 – continued from previous page

Aspect	Investigated issue	Final controlled choice
External-force quantity	Alternative observer-based and ideal-force formulations were tested during development without improving the physical-benchmark result over the final mapped-wrench formulation.	Force-dependent simulation features are constructed from the mapped external wrench and then passed through the defined observation model.
External moment	Adding an independently sampled M_z did not improve the physical-benchmark accuracy during development.	No independent physical M_z is applied; the equivalent moment is generated by the force and its lever arm.
Simulator state	Residual state or contact history from a previous rollout can contaminate the next scenario.	MuJoCo is explicitly reset before every rollout.
Data splitting	Row-wise splitting can place samples from the same rollout in both training and validation sets.	A fixed 90/10 split is performed at rollout level.
Scenario comparison	Independently generated scenarios can confound the effect of the tested randomization.	All controlled experiments reuse one canonical scenario bank and one common validity mask.

Two choices are especially important for the physical interpretation of the final features. First, a non-zero M_z appears only as the moment induced by a planar force and its lever arm. Second, force, torque, distance, and angle features are not perturbed independently. They are reconstructed from one common feature wrench, which preserves the mechanical relationship between them. The exact observation model is described in Section 4.7.

4.4 Canonical Scenario Bank and Contact Generation

A single immutable scenario bank was created before the nominal and randomized experiments. The bank contains 30 blocks. Each block contains 20 robot poses, each pose contains 20 force cases, and every force case is evaluated for all seven contact bodies. Consequently, the original bank contained

$$30 \times 20 \times 20 \times 7 = 84{,}000 \tag{4.1}$$

planned rollouts.

During validation, MuJoCo detected unintended contact already in the initial state for 25 complete pose cases, before the intended external force was applied. Excluding these complete poses removed

$$25 \times 20 \times 7 = 3{,}500 \tag{4.2}$$

rollouts. One additional isolated rollout was excluded because of a singular case. The resulting common mask therefore contains

$$84{,}000 - 3{,}500 - 1 = 80{,}499 \tag{4.3}$$

valid rollouts. The same 80,499 rollout identities are used in every controlled experiment. The final grouped physical ablation also retained the complete 80,499-rollout intersection, so no additional scenarios were removed for that comparison. This pairing prevents changes in the sampled scenarios from being confused with the effects of model randomization, observation noise, or actuator latency.

The construction of the common bank and the exclusions applied during validation are summarised in Figure 4.3.

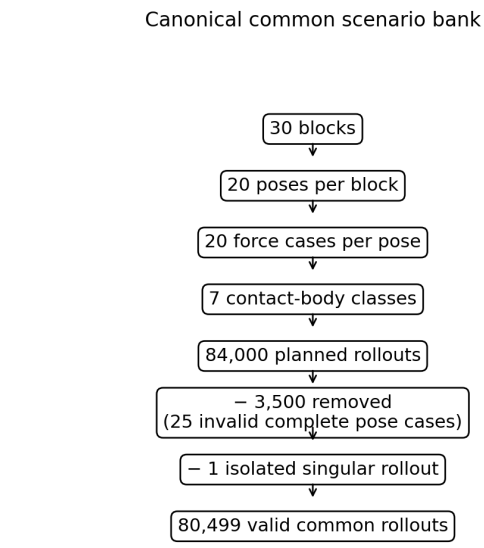


Figure 4.3: Construction and validation of the canonical scenario bank. Removing invalid initial configurations and one singular rollout leaves 80,499 common rollout identities that are reused in every controlled experiment.

Each rollout is identified by the scenario-level fields below. Here, PKM denotes the dataset identifier for the parallel kinematic machine; in this study it refers to the investigated planar 3-RRR robot.

$$(\texttt{RandomizationID}, \texttt{PKM}, \texttt{Configuration}, \texttt{CollidedBody}, \texttt{Force}) . \tag{4.4}$$

The external planar force is applied at the exact centre of mass of the selected body. The physical force passed to MuJoCo remains clean; observation uncertainty is introduced only when the exported feature representation is constructed. A complete MuJoCo reset is performed before every rollout.

4.5 Reference XML and Domain Randomization Strategy

The randomized experiments are centred on a modified reference MuJoCo XML model file, which contains the robot model and its simulation parameters. The original unmodified model file is referred to as the Base-XML model. The modified centre values are engineering-informed reference values and should not be interpreted as experimentally calibrated physical parameters. The effective physical randomization considered in the final analysis varies active- and passive-joint friction and damping.

The Base-XML values and the modified reference centre are compared in Table 4.2. The initial design also includes the parameter `solref1`; its interpretation and diagnostic result are given once in Section 4.12.3.

Table 4.2: Base-XML values and modified-reference centre values for the parameters retained in the initial study.

Parameter	Base XML	Modified centre
Active-joint friction loss	$0\,\mathrm{N\,m}$	$0.05\,\mathrm{N\,m}$
Passive-joint friction loss	$0\,\mathrm{N\,m}$	$0.001\,\mathrm{N\,m}$
Active-joint damping	$1.0\,\mathrm{N\,m\,s/rad}$	$1.0\,\mathrm{N\,m\,s/rad}$
Passive-joint damping	$0\,\mathrm{N\,m\,s/rad}$	$0.01\,\mathrm{N\,m\,s/rad}$
`solref1`	0.1	0.1

The fixed-centre experiment therefore differs from the nominal Base-XML baseline even though no parameter variation is applied. Throughout this thesis, *reference-centred randomization* means randomization around this modified reference centre; it does not mean that the distribution is centred on the Base-XML nominal baseline model.

Let p_c denote the modified-XML centre of parameter p, Δp the half-width defined for $\lambda = 1$, and $u \in [0, 1]$ the paired normalized random draw. The randomized parameter is defined as

$$p(\lambda, u) = p_c + \lambda \Delta p(2u - 1). \tag{4.5}$$

Negative friction or damping values are clipped at zero. The four initial levels are the reference centre ($\lambda = 0$, no parameter variation), Narrow ($\lambda = 0.5$), Medium ($\lambda = 1$), and Broad ($\lambda = 2$). The parameter intervals produced by these levels are listed in Table 4.3.

Table 4.3: Parameter ranges used in the initial coupled randomization-strength sweep.

Parameter	Fixed, $\lambda = 0$	Narrow, $\lambda = 0.5$	Medium, $\lambda = 1$	Broad, $\lambda = 2$
Active friction	0.05	[0.0375, 0.0625]	[0.025, 0.075]	[0, 0.1]
Passive friction	0.001	[0.00075, 0.00125]	[0.0005, 0.0015]	[0, 0.002]
Active damping	1.0	[0.9, 1.1]	[0.8, 1.2]	[0.6, 1.4]
Passive damping	0.01	[0.0075, 0.0125]	[0.005, 0.015]	[0, 0.02]
`solref1`	0.1	[0.0875, 0.1125]	[0.075, 0.125]	[0.05, 0.15]

The normalized draws are paired by block across all levels. Increasing λ therefore changes the magnitude of the same underlying draw rather than replacing it with a different random experiment.

4.6 Observation Noise and Actuator Latency

In the initial λ-sweep, observation-noise magnitude and maximum actuator latency were scaled together with the parameter randomization width. The settings are summarised in Table 4.4.

Table 4.4: Observation-noise and latency settings in the initial coupled λ-sweep. Both effects increase with λ, and all Gaussian noise is truncated at $\pm 3\sigma$.

Quantity	$\lambda = 0$	$\lambda = 0.5$	$\lambda = 1$	$\lambda = 2$
$\sigma_x = \sigma_y$	0	$0.25\,\mathrm{mm}$	$0.5\,\mathrm{mm}$	$1.0\,\mathrm{mm}$
σ_φ	0	0.05°	0.1°	0.2°
$\sigma_{F_x} = \sigma_{F_y}$	0	$0.125\,\mathrm{N}$	$0.25\,\mathrm{N}$	$0.5\,\mathrm{N}$
σ_{M_z}	0	$0.005\,\mathrm{N\,m}$	$0.01\,\mathrm{N\,m}$	$0.02\,\mathrm{N\,m}$
Maximum latency	$0\,\mathrm{ms}$	$1\,\mathrm{ms}$	$2\,\mathrm{ms}$	$4\,\mathrm{ms}$

The initial broad case therefore combines the $\lambda = 2$ parameter ranges, observation noise, and integer latency values between 0 and 4 simulation steps. It must not be interpreted as a single-factor physical-range experiment. The later controlled studies separate these effects.

The frozen observation-noise model used in the final selected condition has

$$\begin{aligned} \sigma_x = \sigma_y &= 1\,\mathrm{mm}, \\ \sigma_\varphi &= 0.2^\circ, \\ \sigma_{F_x} = \sigma_{F_y} &= 0.5\,\mathrm{N}, \\ \sigma_{M_z} &= 0.02\,\mathrm{N\,m}, \end{aligned} \tag{4.6}$$

with Gaussian noise truncated at $\pm 3\sigma$. The frozen noise seed is 20260804. The final selected model uses actuator latency fixed at 0 ms; this is an experimental setting and should not be interpreted as evidence that the physical robot has no timing delay.

4.7 Consistent Noisy-Feature Definition

The physical external force applied to MuJoCo is kept clean:

$$\boldsymbol{F}_{\text{applied}} = \boldsymbol{F}_{\text{clean}}. \tag{4.7}$$

Observation noise is added only after the physical or mapped platform wrench has been obtained:

$$\boldsymbol{F}_{\text{feature}} = \boldsymbol{F}_{\text{ext,clean}} + \boldsymbol{\varepsilon}_F. \tag{4.8}$$

The same noisy wrench is then used consistently to construct all physically dependent features:

$$\boldsymbol{\tau}_{\text{feature}} = \boldsymbol{J}^{\mathsf{T}} \boldsymbol{F}_{\text{feature}}, \tag{4.9}$$

while the distance and angle features are also recomputed from $\boldsymbol{F}_{\text{feature}}$. No independent joint-torque noise is added. Noisy platform pose is used internally for the geometric feature calculation, but the platform pose itself is not included in the final twelve-dimensional classifier input.

This construction preserves the physical dependence between the external-wrench, torque, distance, and angle features. The observation noise on M_z represents uncertainty in the mapped third wrench component; it is not a directly applied physical torque.

4.8 Numerical Implementation of the Geometric Features

The MATLAB implementation uses the routine `CalcGeomFeatures_cluster` for the distance and angle features. It first extracts the planar force component

$$\boldsymbol{f}_{xy} = \begin{bmatrix} F_x & F_y \end{bmatrix}^{\mathsf{T}} \tag{4.10}$$

and constructs a force-direction vector. With $\varepsilon_f = 10^{-3}$ in the numerical force units, the code uses

$$\boldsymbol{u}_f = \begin{cases} \dfrac{\boldsymbol{f}_{xy}}{\|\boldsymbol{f}_{xy}\|_2}, & \|\boldsymbol{f}_{xy}\|_2 \geq \varepsilon_f, \\ \dfrac{\boldsymbol{f}_{xy}}{\|\boldsymbol{f}_{xy}\|_2 + \varepsilon_f}, & \|\boldsymbol{f}_{xy}\|_2 < \varepsilon_f. \end{cases} \tag{4.11}$$

The small-force branch prevents a numerical division by zero. For the ordinary non-zero contact-force cases, $\boldsymbol{u}_f$ is therefore the unit direction of the planar force.

For the wrench $\boldsymbol{F}_{\text{feature}} = [F_x, F_y, M_z]^{\mathsf{T}}$, the implementation forms

$$\boldsymbol{S}(\boldsymbol{f}) = \begin{bmatrix} 0 & 0 & F_y \\ 0 & 0 & -F_x \\ -F_y & F_x & 0 \end{bmatrix} \tag{4.12}$$

and obtains the minimum-norm lever arm from

$$\boldsymbol{r}_{\min} = \left[\boldsymbol{S}^{\mathsf{T}}(\boldsymbol{f})\right]^{\dagger} \begin{bmatrix} 0 & 0 & M_z \end{bmatrix}^{\mathsf{T}}. \tag{4.13}$$

The point used to represent the line of action is then shifted by the current platform position:

$$\boldsymbol{r}_{\text{LoA},0} = \begin{bmatrix} x_E & y_E & 0 \end{bmatrix}^{\mathsf{T}} + \boldsymbol{r}_{\min}. \tag{4.14}$$

For each chain, the forward kinematics provide the coupling-joint position $\boldsymbol{r}_{cJ_i}$ and the second-link vector $\boldsymbol{a}_{2,i}$, which points from the passive joint toward the coupling joint. Defining $\boldsymbol{u}_{f,3D} = [\boldsymbol{u}_f^{\mathsf{T}}, 0]^{\mathsf{T}}$, the implemented distance is

$$d_i = \left\| (\boldsymbol{r}_{cJ_i} - \boldsymbol{r}_{\text{LoA},0}) \times \boldsymbol{u}_{f,3D} \right\|_2 . \tag{4.15}$$

When the force direction is numerically undefined ($\|\boldsymbol{u}_{f,3D}\|_2 \leq 10^{-5}$), the implementation assigns the sentinel value 100 to the distance. This branch is a numerical safeguard rather than a physical distance model.

The angle feature is computed directly with MATLAB's `atan2d` function:

$$\alpha_i = \operatorname{atan2d}\left(\|\boldsymbol{u}_{f,3D} \times \boldsymbol{a}_{2,i}\|_2, \boldsymbol{u}_{f,3D}^{\mathsf{T}} \boldsymbol{a}_{2,i} \right). \tag{4.16}$$

Because the cross-product magnitude is non-negative, α_i is an unsigned angle in the interval $[0^\circ, 180^\circ]$. This confirms the convention introduced in Section 3.3.4: 0° denotes parallel directions, 180° antiparallel directions, and 90° perpendicular directions. The three distances and three angles exported by this implementation are the `distance_chain_1-3` and `angle_chain_1-3` features used by the classifier.

4.9 Final Feature Set

The final feature vector contains twelve quantities:

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{F}_{\text{ext}}^{\mathsf{T}} & \boldsymbol{\tau}_{a,\text{ext}}^{\mathsf{T}} & \boldsymbol{d}^{\mathsf{T}} & \boldsymbol{\alpha}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^{12}. \tag{4.17}$$

Here, $\boldsymbol{F}_{\text{ext}} = [F_x, F_y, M_z]^{\mathsf{T}}$, $\boldsymbol{\tau}_{a,\text{ext}} = [\tau_1, \tau_2, \tau_3]^{\mathsf{T}}$, $\boldsymbol{d} = [d_1, d_2, d_3]^{\mathsf{T}}$, and $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \alpha_3]^{\mathsf{T}}$.

Robot scale and end-effector pose are excluded from the final classifier input. The pose may still be used internally when the geometric features are evaluated.

4.10 Rollout Sampling and Dataset Splitting

A maximum of ten uniformly distributed time samples is retained from each rollout. Training and validation are separated at rollout level using the fixed scenario identity rather than by splitting individual rows. Samples from one physical scenario therefore cannot appear in both subsets.

Table 4.5 gives the fixed 90/10 split used in every controlled experiment.

Table 4.5: Controlled rollout-level split used in all experiments. Samples from one rollout cannot appear in both training and validation subsets.

Subset	Number of rollouts
Training	72,449
Validation	8,050
Total	80,499

The standard scaler is fitted on the simulation training subset only. The fitted transformation is then reused unchanged for simulation validation and for both physical benchmark datasets.

4.11 Neural Network Architecture and Training

All controlled models use the same feedforward architecture,

$$12 \to 128 \to 64 \to 32 \to 7. \tag{4.18}$$

The training settings in Table 4.6 are held fixed so that the comparison between simulation conditions is not confounded by changes in the learning procedure.

Table 4.6: Preprocessing and training settings held fixed across the controlled experiments so that changes in physical performance are not caused by changes in the learning procedure.

Setting	Value
Random seed	42
Device	CPU
CPU threads	8
PyTorch version	2.0.0.post200
DataLoader workers	0
Deterministic algorithms	Enabled
Maximum samples per rollout	10
Train/validation split	90/10 at rollout level
Batch size	64
Learning rate	0.001
Maximum epochs	100
Early-stopping patience	10
Checkpoint criterion	Minimum simulation-validation loss

4.12 Controlled Experimental Sequence

The controlled experiments were carried out as a sequence rather than as one single parameter sweep. This distinction is important because different factors were intentionally frozen at different stages.

The first stage was the coupled randomization-strength sweep introduced in Sections 4.5 and 4.6. At the three non-zero range settings $\lambda = 0.5$, 1, and 2, the friction and damping ranges, observation-noise magnitude, and maximum latency were widened together. Average physical-benchmark performance increased across these three randomized settings. This made the broader setting worth pursuing, but because several uncertainty sources changed simultaneously, the trend could not be attributed to the physical range alone. The next stages therefore separated observation noise and latency before testing still broader physical ranges.

4.12.1 Noise–Latency Ablation

After the coupled sweep, a controlled noise–latency ablation was performed at the broad $\lambda = 2$ friction-and-damping ranges. The four conditions are defined in Table 4.7.

Table 4.7: Definitions of the controlled noise–latency ablation experiments. The design separates observation noise and latency while keeping the $\lambda = 2$ friction-and-damping ranges fixed.

Condition	Observation noise	Actuator latency
Friction+damping, clean	No	0 ms
Friction+damping + noise	Yes	0 ms
Friction+damping + latency	No	[0, 2, 4, 6, 8, 10] ms
Friction+damping + noise + latency	Yes	[0, 2, 4, 6, 8, 10] ms

For latency-enabled experiments, the six latency values are assigned to the 30 blocks such that each value is used by exactly five blocks. The noise-enabled conditions use the frozen observation model in Equation (4.6). The ablation showed which observation/control setting should be held fixed before extending the physical randomization range.

4.12.2 Controlled Physical-Range Extension

The initial coupled sweep showed a recovery in physical performance as the setting was widened from $\lambda = 0.5$ through $\lambda = 1$ to $\lambda = 2$. Because observation noise and latency also changed in that stage, this trend could not be interpreted as a physical-only lambda curve. It nevertheless motivated a direct test of whether the physical ranges should be widened further.

A separate controlled comparison was therefore performed at $\lambda = 2, 3, 4$. Across these three datasets, the same 80,499 rollout identities, rollout-level train/validation split, normalized parameter draws, observation-noise magnitudes and paired noise stream, network seed, preprocessing, and training protocol were reused. The selected observation noise was held fixed and actuator latency was fixed at 0 ms. The intended changed factor was the width of the friction and damping randomization ranges. The results of this experiment determine the range used in the final grouped physical ablation.

4.12.3 Final Friction–Damping Ablation

The final grouped physical study separates two binary factors at the selected randomization range:

- active- and passive-joint friction randomization;
- active- and passive-joint damping randomization.

This produces four interpretable conditions: no friction or damping randomization, damping only, friction only, and friction plus damping. Observation noise is ON for all four conditions using the frozen model in Equation (4.6), actuator latency is fixed at 0 ms, and all four conditions use the complete 80,499-rollout common intersection. The same training seed, split, scaler rule, and checkpoint criterion are retained.

As an additional diagnostic check, `solref1` was toggled while each friction–damping setting was otherwise kept unchanged. In MuJoCo, `solref1` is the first component of the positive `solref` contact-constraint setting and controls the solver reference time constant; it is therefore a numerical contact-solver parameter rather than mechanical stiffness [23]. Changing it produced no observable change in the downstream training or benchmark results under the retained force-injection protocol. It is therefore not included as a factor in the final physical-ablation tables or figures.

4.13 Evaluation Procedure

The evaluation protocol is fixed across all controlled experiments and is summarised in Table 4.8. This table defines the role of each dataset and is the reference point for the remainder of the thesis.

Table 4.8: Evaluation protocol. Physical data are used only for external benchmarking after simulation-side model selection.

Data	Role	Use in the pipeline
Simulation training	Model fitting	Fits the standard scaler and updates the FFNN parameters.
Simulation validation	Model selection	Selects the checkpoint by minimum validation loss; no gradient updates are performed on this subset.
Bench Set 1 and Bench Set 2	Fixed external benchmarks	Evaluated only after checkpoint selection. They are not used for gradient-based training, scaler fitting, or feature alignment.

The same two physical benchmarks are reused across the sequence of experiments. Their results therefore inform which simulation setting is investigated next, even though they do not enter the learning procedure. The study is consequently described as simulation-only training with repeated external benchmark evaluation, not as a strict one-shot hold-out experiment.

Reported metrics are overall accuracy, macro-averaged F1-score, class-wise precision, recall and F1-score, and row-normalized confusion matrices. Confusion-matrix rows denote true classes and columns denote predicted classes.

CHAPTER 5

Results

This chapter reports the measured performance of the completed controlled experiments. The order follows the experimental sequence: the initial coupled randomization sweep, the noise–latency separation, the controlled $\lambda = 2, 3, 4$ physical-range extension, and the final friction–damping ablation. Interpretation is developed in Chapter 6. All models use the common scenario bank, rollout mask, data split, preprocessing procedure, network architecture, and deterministic training seed described in Chapter 4.

5.1 Headline Comparison

The main physical result is summarised in Table 5.1 and Figure 5.1. The final selected condition combines friction and damping randomization at $\lambda = 2$ with the frozen observation-noise model and actuator latency fixed at 0 ms.

Table 5.1: Headline comparison of the principal controlled models on the two physical benchmarks. The friction + damping + noise condition gives the highest accuracy on both datasets and on average.

Model	Bench Set 1	Bench Set 2	Average
Nominal baseline	63.51%	70.87%	67.19%
Friction+damping, clean	61.38%	71.01%	66.19%
Friction+damping + noise	**66.29%**	**76.99%**	**71.64%**
Friction+damping + noise + latency	58.75%	66.53%	62.64%

Relative to the Base-XML nominal baseline, the final selected model improves average physical accuracy by 4.45 percentage points and average macro-F1 by 4.33 percentage points.

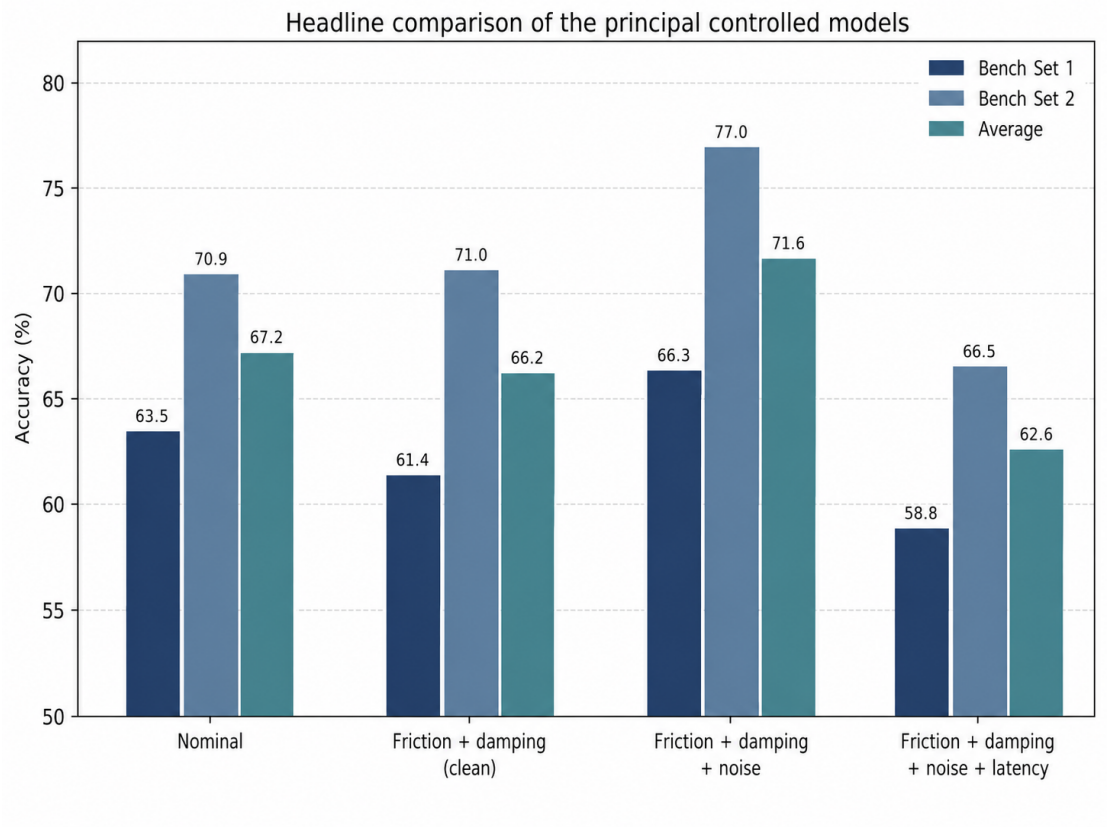


Figure 5.1: Headline accuracy comparison for the principal controlled models. Observation noise provides the clearest gain for the friction–damping model, while the tested noise–latency combination falls below the nominal baseline.

5.2 Initial Coupled Randomization-Strength Sweep

The first experimental stage varied the overall randomization setting from the fixed modified reference through $\lambda = 0.5$, 1, and 2. The physical parameter ranges, observation-noise magnitude, and maximum latency all scaled with λ, so this experiment is a coupled sweep rather than a physical-only range study. The physical accuracies are listed in Table 5.2.

Table 5.2: Physical accuracy in the initial coupled randomization-strength sweep. Physical parameter ranges, observation noise, and latency all scale with λ.

Experiment	Bench Set 1	Bench Set 2	Average
Nominal baseline	63.5105%	70.8664%	67.1884%
Fixed reference, $\lambda = 0$	60.7988%	66.7614%	63.7801%
$\lambda = 0.5$	59.2317%	66.7182%	62.9749%
$\lambda = 1$	59.6718%	70.7483%	65.2100%
$\lambda = 2$	64.3371%	70.1124%	67.2248%

Among the randomized settings, the average physical accuracy rises from 62.97% at $\lambda = 0.5$ to 65.21% at $\lambda = 1$ and 67.22% at $\lambda = 2$. The $\lambda = 2$ average is essentially tied with the nominal baseline, exceeding it by only 0.04 percentage points. Because noise and latency change together with the physical ranges, the increasing trend from $\lambda = 0.5$ to $\lambda = 2$ cannot establish that physical randomization alone is responsible. It did, however, motivate separating the uncertainty sources and later testing whether the physical range should be widened beyond $\lambda = 2$.

5.3 Noise–Latency Ablation

The next stage separates observation noise and the tested latency model while holding the broad $\lambda = 2$ friction-and-damping ranges fixed. Table 5.3 reports the main metrics.

Table 5.3: Controlled noise–latency ablation. Observation noise improves both physical benchmarks; the tested latency model provides little benefit alone and removes the gain when combined with noise.

Condition	B1 acc.	B2 acc.	Avg. acc.	Avg. macro F1
Friction+damping, clean	61.3769	71.0069	66.1919	64.1992
Friction+damping + noise	**66.2931**	**76.9909**	**71.6420**	**70.3340**
Friction+damping + latency	61.9602	70.8296	66.3949	64.3671
Friction+damping + noise + latency	58.7502	66.5280	62.6391	61.5460

All values are percentages. Relative to the clean friction–damping condition, observation noise increases average accuracy by approximately 5.45 percentage points and average macro-F1 by approximately 6.13 percentage points. Latency alone changes average accuracy by only about +0.20 percentage points. When latency is combined with noise, average accuracy falls by about 9.00 percentage points relative to the noise-only model. The selected observation setting is therefore the frozen noise model with actuator latency fixed at 0 ms. This setting is carried forward to the controlled physical-range extension.

5.4 Controlled Physical-Range Extension

The recovery seen from $\lambda = 0.5$ to $\lambda = 2$ in the initial coupled sweep raised a natural question: would an even wider physical range continue to improve transfer? To answer this without the earlier confound, observation noise was now frozen at the selected level and randomized latency was fixed at 0 ms. Only the friction-and-damping randomization width was extended to $\lambda = 3$ and $\lambda = 4$, with $\lambda = 2$ retained as the controlled reference.

The results are given in Table 5.4.

Table 5.4: Controlled physical-range extension with frozen observation noise and actuator latency fixed at 0 ms. Among the tested ranges, $\lambda = 2$ gives the strongest transfer on both physical benchmarks.

Physical λ	B1 acc.	B2 acc.	Avg. acc.	Avg. macro F1
2	**66.2931**	**76.9909**	**71.6420**	**70.3340**
3	64.8343	69.8675	67.3509	65.8652
4	65.7284	74.8234	70.2759	68.8858

Relative to $\lambda = 2$, average accuracy decreases by 4.29 percentage points at $\lambda = 3$ and by 1.37 percentage points at $\lambda = 4$. The same ranking is obtained from average macro-F1 and from each individual physical benchmark. The response is non-monotonic because $\lambda = 4$ partially recovers the loss at $\lambda = 3$, but neither wider setting exceeds $\lambda = 2$. The experiments therefore support selecting $\lambda = 2$ as the working range among those tested; they do not establish a mathematical or global optimum. This selected range is used for the final friction–damping ablation.

5.5 Final Friction–Damping Ablation and Selected Model

The final physical study asks which of the two effective randomized parameter groups is responsible for the transfer improvement. Observation noise is fixed ON for every condition, actuator latency is fixed at 0 ms, and any active physical factor uses the selected $\lambda = 2$ range. Table 5.5 and Figure 5.2 show the result.

Table 5.5: Final friction–damping ablation at $\lambda = 2$ with observation noise fixed ON and actuator latency fixed at 0 ms. Neither factor improves the physical average alone, while their combination gives the strongest result.

Condition	Friction DR	Damping DR	B1 acc.	B2 acc.	Avg. acc.
Reference centre + noise	OFF	OFF	61.1004	72.0586	66.5795
Damping + noise	OFF	ON	59.6147	72.2638	65.9393
Friction + noise	ON	OFF	58.0045	67.9578	62.9812
Friction+damping + noise	**ON**	**ON**	**66.2931**	**76.9909**	**71.6420**

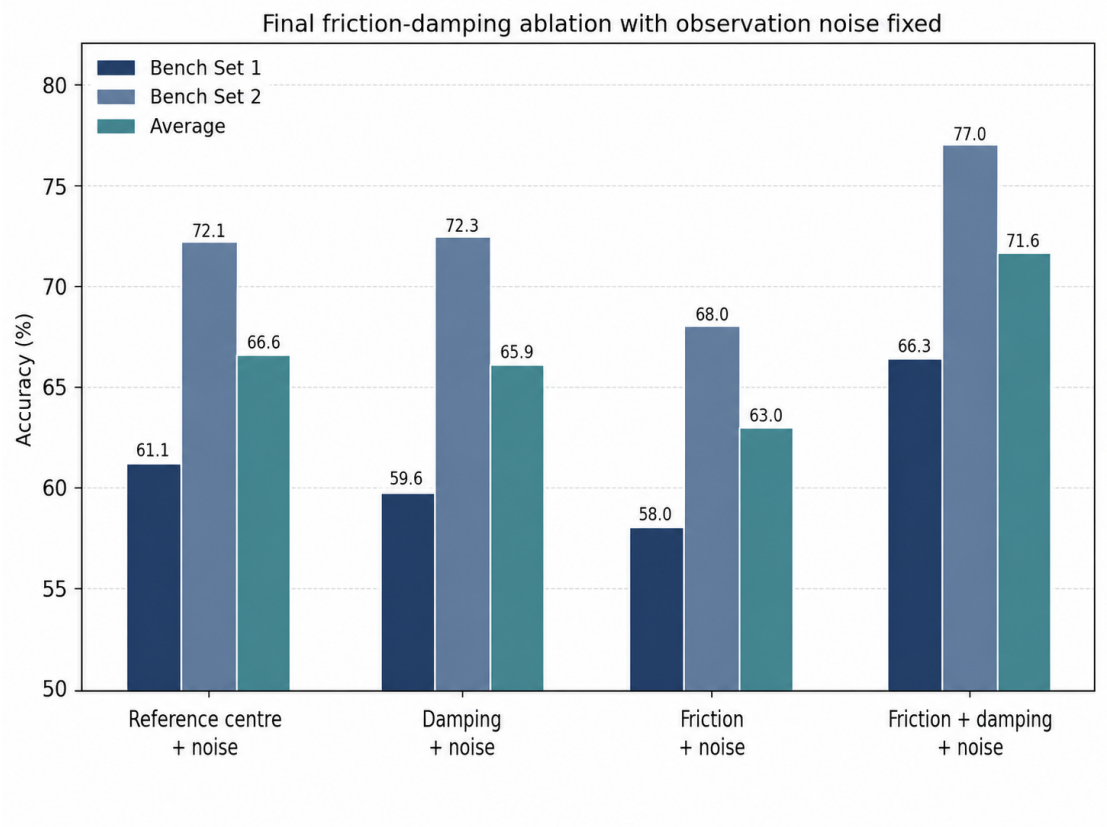


Figure 5.2: Final friction–damping ablation with observation noise fixed ON. Randomizing friction or damping alone does not improve the physical benchmark average, whereas their joint randomization produces the strongest result on both datasets.

The separate effects make the interaction clear. Relative to the reference centre with noise, damping alone changes average accuracy by about -0.64 percentage points and friction alone by about -3.60 percentage points. In contrast, adding damping when friction is already randomized increases the average by 8.66 percentage points, while adding friction when damping is already randomized increases it by 5.70 percentage points. The difference-of-differences is +9.30 percentage points, corresponding to a balanced friction–damping interaction effect of +4.65 percentage points in average accuracy and +5.06 percentage points in average macro-F1. Under the tested ranges and observation model, the benefit therefore comes from the combination rather than either factor in isolation.

The final selected model is therefore the $\lambda = 2$ friction + damping + noise condition with actuator latency fixed at 0 ms. Its average physical accuracy is 71.64% and its average macro-F1 is 70.33%. Relative to the friction/damping-off condition with the same observation noise, the largest average class-wise F1 improvements occur for C2L1 (+9.91 percentage points), the mobile platform (+8.11), C2L2 (+7.61), and C1L1 (+6.12). C3L1 is essentially unchanged on average (-0.29 percentage points).

Bench Set 1 remains the harder recording, and C2L1 is the clearest residual class-level

failure mode. Its F1-score improves from 16.71% to 32.80% and its recall from 9.15% to 19.69%, but the row-normalized confusion matrix in Figure C.1 shows that approximately 73% of true C2L1 samples are still classified as C2L2, while only about 20% are classified correctly. The dominant remaining error on Bench Set 1 is therefore a confusion between the first and second links of chain 2, particularly C2L1 being assigned to C2L2. Complete class-wise results and confusion matrices are provided in Appendix C. A post-hoc analysis of this benchmark-specific failure mode is discussed in Section 6.1.

CHAPTER 6

Discussion

The completed experiments show that the classification problem can be fitted very accurately in simulation but remains substantially harder on the physical robot. They also show that transfer is not improved simply by increasing every source of simulated variability. The sequence of experiments makes it possible to distinguish the roles of observation uncertainty, randomized latency, physical-range width, and the interaction between friction and damping.

6.1 Interpretation of the Sim-to-Real Results

The Base-XML nominal model reaches 98.16% simulation-validation accuracy but only 67.19% average physical-benchmark accuracy. The final friction + damping + noise model reaches a lower simulation-validation accuracy of 96.84% while achieving 71.64% on the physical benchmarks. The same pattern appears inside the final ablation: the friction/damping-off condition with observation noise reaches 97.30% simulation validation and 66.58% average physical accuracy, whereas the friction + damping condition reaches 96.84% simulation validation and 71.64% physically. Higher source-domain validation accuracy is therefore not a reliable proxy for better sim-to-real transfer once all models already fit the simulated classification task well.

The physical comparisons are more informative. Friction and damping randomization without observation noise does not improve the average over the Base-XML nominal model. Adding the consistent observation-noise layer, however, improves both independently recorded physical datasets. This suggests that an important part of the remaining mismatch lies in the quantities presented to the classifier rather than only in the deterministic mechanical model. Bench Set 1 remains consistently harder than Bench Set 2, which also shows that the physical target distribution changes across recordings.

The clearest remaining class-level problem is C2L1 on Bench Set 1: approximately 73% of its samples are classified as C2L2 and only 19.69% are classified correctly. A short post-hoc feature-space check found no evidence of a global C2L1/C2L2 label swap; the Bench Set 1 C2L1 centroid remained closer to simulated C2L1 than to simulated C2L2. The

misclassified samples mainly differed in their force-line geometry, while the same class was classified correctly in 95.89% of Bench Set 2 samples. This points to a benchmark-specific local shift rather than a mapping error, although the post-hoc analysis does not establish a unique physical cause.

6.2 Influence of Feature Selection and Consistency

The final representation is built from quantities that have a direct counterpart in both domains: the external wrench, actuator torques, and force-line distance and angle features. Direct pose and scale cues are excluded from the classifier input, while pose remains available internally for geometric calculations. These choices reduce the opportunity for the network to exploit correlations tied only to the simulated scenario construction, although the development decisions were not all repeated as fully paired ablations.

A more consequential choice is that force-dependent features are constructed from one common feature wrench. Once observation noise is added, actuator torques, line-of-action distances, and angles are recomputed from that same perturbed wrench. The classifier therefore sees a noisy but mechanically coherent observation instead of independently corrupted channels. The positive noise ablation on both physical datasets supports the relevance of this consistent observation layer to the sim-to-real gap.

The treatment of M_z follows the same principle. The simulator does not apply an independent physical moment in the final scenarios. Instead, the equivalent moment arises from the planar force and its lever arm. This keeps force, contact location, and moment coupled in the same way as the point-force scenario definition.

6.3 Influence of Domain Randomization

The initial λ-sweep suggested that broader settings were worth testing: the average benchmark accuracy increased from $\lambda = 0.5$ to $\lambda = 1$ and then to $\lambda = 2$. Because observation noise and latency were also changing in that sweep, this was only a reason to continue the study, not evidence that a wider physical range caused the improvement.

Noise and latency were therefore separated first. Observation noise was the clear positive factor, while the tested latency model did not improve the noise-enabled result. With observation noise fixed and actuator latency fixed at 0 ms, the physical range was then tested at $\lambda = 2, 3, 4$. $\lambda = 3$ and $\lambda = 4$ both remained below $\lambda = 2$, so $\lambda = 2$ was retained as the best range among those tested, not as a universal optimum.

The final friction–damping ablation gives the clearest physical result. Friction randomization alone and damping randomization alone both fail to improve the physical benchmark average, whereas combining them raises the average to 71.64%. The useful effect therefore comes from their interaction under the tested setup rather than from either parameter group by itself.

The tested latency randomization should be interpreted just as narrowly. It changes little without observation noise and reduces performance when added to the noise-enabled model. This result applies to the latency representation used here; it does not show that timing mismatch is absent from the physical robot.

6.4 Comparison with Previous Work

Giourgas studied the same seven-class problem on the same research platform [5]. The useful comparison separates the conventional FFNN baseline from the transfer method. His reported FFNN sim-to-real mean accuracy is 61.60%, while the DANN sim-to-real mean is 70.20%. The present study uses different datasets and a different source-domain construction, so the values in Table 6.1 are contextual rather than a controlled head-to-head benchmark.

Table 6.1: Contextual comparison with the two closest sim-to-real results reported by Giourgas [5]. The protocols and averaging procedures are not identical.

Comparison	Giourgas	Present thesis	Difference
FFNN sim-to-real baseline	FFNN: 61.60% mean accuracy	Nominal baseline: 67.19% average bench accuracy	+5.59 pp
Main transfer approach	DANN: 70.20% mean accuracy	Friction+damping+noise model: 71.64% average bench accuracy	+1.44 pp

The baseline comparison indicates that the revised simulation and preprocessing pipeline already improves on the earlier FFNN sim-to-real result. The transfer-method comparison is much closer: the final simulation-only FFNN lies in the same numerical range as the target-informed DANN. Since Giourgas uses physical target samples during adversarial alignment and the experimental protocols differ, the 1.44-point difference is not evidence that one transfer strategy is statistically superior to the other.

Mohammad et al. provide a useful physical-to-physical reference for the same seven-body classification task [2]. Their FNN was trained on physical measurements from one robot

configuration and tested on physical measurements from two different configurations, reaching 83.98% accuracy. Because both training and testing use real-robot data, this is the most favourable case among these comparisons. The present final model is trained only in simulation: it averages 71.64% across the two physical benchmarks and reaches 76.99% on Bench Set 2, about 7 percentage points below the reported 83.98%. The Bench Set 2 result is therefore relatively close to the physical-to-physical reference despite using no physical training data. The protocols are different, so this remains a contextual rather than a head-to-head comparison. Table 6.2 and Figure 6.1 summarize the comparison.

Table 6.2: Comparison of the present simulation-trained models with the physical-to-physical FNN result reported by Mohammad et al. [2].

Model / source	Training domain	Physical evaluation	Accuracy
Nominal baseline, present thesis	Base-XML simulation only	Average of Bench Set 1 and Bench Set 2	67.19%
Final randomized model, present thesis	$\lambda = 2$ friction + damping + observation-noise simulation	Average of Bench Set 1 and Bench Set 2	**71.64%**
Mohammad et al. FNN [2]	Physical measurements in one robot configuration	Physical measurements in two different robot configurations	83.98%

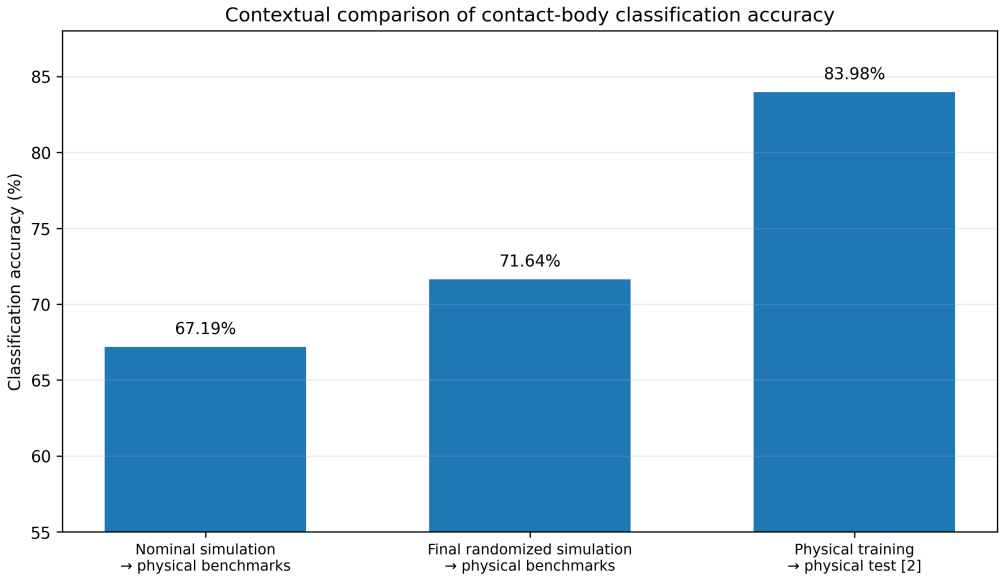


Figure 6.1: Classification-accuracy comparison with the physical-to-physical FNN result of Mohammad et al. [2]. The present-thesis bars use the average of Bench Set 1 and Bench Set 2.

6.5 Limitations

The conclusions are subject to five main limitations.

- **Simplified contact and observation models.** Simulation applies the force at the body centre of mass, and the observation model uses truncated Gaussian perturbations rather than the full dynamics of the physical force observer.
- **Engineering-defined parameter ranges.** The friction and damping ranges are selected engineering ranges rather than distributions identified directly from measurements on the physical robot.
- **Mechanical compliance is outside the tested scope.** Mechanical stiffness or compliance was not explicitly modeled or randomized.
- **Single training seed.** A fixed seed improves pairing between experiments but does not quantify variability due to network initialization or minibatch order. Small differences should therefore not be interpreted as statistical significance.
- **Repeated use of the same physical benchmarks.** Bench Set 1 and Bench Set 2 do not enter model fitting or checkpoint selection, but their repeated inspection influences which simulation experiment is performed next. They are not untouched one-shot holdouts.

6.6 Future Work

Several extensions could make the sim-to-real evaluation more complete. First, the simulation could include a wider variety of contact scenarios. In the present study, the external force is applied at the centre of mass of the selected body; future work could vary the contact location along the links and mobile platform and study how this affects the induced moment, force-line geometry, and classification performance. The scenario bank could also be expanded with additional robot poses, force directions, and force magnitudes.

Mechanical stiffness and compliance were not explicitly randomized in this thesis. A future model could include physically meaningful stiffness or compliance parameters and randomize them together with friction and damping, using ranges derived from measurements where possible. This would provide a more realistic study of mechanical uncertainty than varying numerical contact-solver parameters.

The observation model could also be improved using measurements from the physical force observer, including bias, filtering effects, and configuration-dependent errors in-

stead of relying only on Gaussian noise. Similarly, actuator timing could be represented with a more realistic delay model if measured timing data become available.

Finally, the selected experiments should be repeated with multiple training seeds to quantify the variability of the neural-network training. A larger and independent physical benchmark dataset, ideally collected under different robot configurations and contact conditions, would also provide a stronger final assessment of sim-to-real generalization.

CHAPTER 7

Conclusion

This thesis investigated sim-to-real contact-body classification on a planar 3-RRR parallel robot using simulation-only network training. Rather than changing the classifier architecture, the work focused on the source pipeline: a canonical paired contact-scenario bank, rollout-aware preprocessing, reference-centred friction and damping randomization, mechanically consistent feature construction, and an observation model applied to the quantities seen by the classifier. The experimental sequence first explored coupled randomization strength, then separated observation noise and actuator latency, extended the physical randomization range to $\lambda = 3$ and $\lambda = 4$, and finally isolated friction and damping.

7.1 Summary and Main Findings

The Base-XML nominal model reaches 63.51% accuracy on Bench Set 1 and 70.87% on Bench Set 2, corresponding to an average of 67.19%. The final selected model combines active/passive friction and damping randomization at $\lambda = 2$ with the frozen consistent observation-noise model and actuator latency fixed at 0 ms. It reaches 66.29% and 76.99% on the two physical benchmarks, increasing average accuracy to 71.64% and average macro-F1 to 70.33%.

The completed experiments explain how this model was reached. In the initial coupled sweep, average physical performance recovered as the setting was widened from $\lambda = 0.5$ to $\lambda = 1$ and $\lambda = 2$. Because noise and latency changed at the same time, that result motivated rather than proved the value of wider physical randomization. The subsequent noise–latency ablation identified observation noise as the clearest positive transfer factor and showed that the tested latency representation removed the gain when combined with noise. With observation noise fixed and actuator latency fixed at 0 ms, the controlled $\lambda = 2, 3, 4$ extension showed that widening beyond $\lambda = 2$ did not improve transfer.

The final friction–damping ablation gives the most specific physical result. Friction randomization alone and damping randomization alone both fail to improve the physical benchmark average relative to the friction/damping-off condition with the same observa-

tion noise. Their joint randomization, however, produces the strongest result, revealing a strong positive interaction under the tested protocol. The benefit therefore does not come from either physical factor independently, but from their combination.

7.2 Answers to the Research Questions

RQ1: Can a contact-body classifier trained exclusively on randomized simulation data generalise to the physical robot? Yes. The final selected model transfers to both physical benchmark datasets without using physical samples for network training, scaler fitting, feature alignment, or checkpoint selection. A substantial gap relative to simulation validation remains, but the final physical accuracy exceeds the nominal simulation-only baseline on both recordings.

RQ2: Which features and simulation-design choices are most important for transfer? The final pipeline relies on external-wrench, actuator-torque, force-line distance, and force-line angle features whose mechanical relationships are preserved through a common observation model. Among the controlled uncertainty studies, observation noise provides the clearest positive transfer effect. The final physical ablation further shows that friction and damping must be considered jointly under the tested ranges because neither factor improves the physical average when randomized alone.

RQ3: To what extent do reference-centred physical randomization and the tested observation model improve robustness over nominal simulation training? Physical randomization without observation noise does not exceed the Base-XML nominal average in the completed broad clean comparison. When the selected $\lambda = 2$ friction and damping ranges are combined with the consistent observation-noise model, average physical accuracy increases from 67.19% to 71.64%. Wider controlled ranges at $\lambda = 3$ and $\lambda = 4$ do not improve this result.

RQ4: Is the behaviour consistent across the two physical benchmark datasets? Partly. The direction of the main improvement is consistent: the selected model outperforms the nominal baseline on both datasets, and the benefit of observation noise is visible in both recordings. However, absolute performance is not fully comparable. The selected model reaches 66.29% on Bench Set 1 and 76.99% on Bench Set 2, and the difference is particularly pronounced for C2L1. The post-hoc analysis indicates that this residual C2L1 failure is associated with a benchmark-specific region of the force-line feature

distribution rather than with evidence of a global C2L1/C2L2 class-mapping mismatch. The results therefore indicate a benchmark-specific component of the remaining domain shift despite a consistent overall improvement trend.

7.3 Outlook

Future work should move from engineering-defined uncertainty toward measured uncertainty. The most important extensions are explicit mechanical compliance/stiffness modelling with physically justified ranges, an observation model derived from real observer residuals, multiple training seeds, and an additional physical dataset reserved as a stricter final holdout. These steps would clarify how much of the remaining sim-to-real gap is caused by unmodelled mechanical compliance, observation dynamics, optimization variability, or benchmark-specific conditions.

The main conclusion is not that more randomization is inherently better. For this contact-classification problem, the strongest transfer was obtained when the simulated observations were made more representative and the relevant physical uncertainty sources were combined coherently. Under the tested protocol, that combination is $\lambda = 2$ friction and damping randomization together with the consistent observation-noise model.

APPENDIX A

Additional Dataset Information

A.1 Canonical Class-Label Order

The same class convention is used in every simulation and physical evaluation. Rows of all confusion matrices correspond to true classes, whereas columns correspond to predicted classes. The numerical label mapping is given in Table A.1.

Table A.1: Canonical class-label order used throughout the thesis.

Label	Contact body
0	Mobile platform
1	C1L1
2	C1L2
3	C2L1
4	C2L2
5	C3L1
6	C3L2

A.2 Scenario-Bank Summary

The final scenario-bank counts and fixed contact-generation choices are summarised in Table A.2.

Table A.2: Canonical common scenario bank used for all controlled experiments.

Quantity	Value
Blocks	30
Poses per block	20
Force cases per pose	20
Contact-body classes	7
Planned rollouts	84,000
Excluded complete poses	25
Rollouts excluded through complete poses	3,500
Additional isolated excluded rollouts	1
Final valid common rollouts	80,499
Final physical-ablation intersection	80,499
Additional physical-ablation exclusions	0
Contact position	Exact body COM
Directly applied physical M_z	0
Robot scale	1.0

A.3 Controlled Experiment Summary

The controlled conditions referred to throughout the Results chapter are summarised in Table A.3. Descriptive names are used instead of internal data-generation codes.

Table A.3: Descriptive summary of the controlled experiments used in the thesis.

Condition	Description
Nominal baseline	Base-XML simulation baseline.
Fixed reference	Modified reference XML with $\lambda = 0$ in the initial coupled sweep.
Coupled $\lambda = 0.5$	Initial coupled sweep; physical ranges, noise, and latency scale together.
Coupled $\lambda = 1$	Initial coupled sweep at $\lambda = 1$.
Coupled $\lambda = 2$	Initial coupled sweep at $\lambda = 2$.
Friction+damping, clean	$\lambda = 2$ friction and damping randomization without observation noise and with actuator latency fixed at 0 ms.
Friction+damping + noise	$\lambda = 2$ friction and damping randomization with frozen observation noise and actuator latency fixed at 0 ms.
Friction+damping + latency	$\lambda = 2$ friction and damping randomization with randomized latency and no observation noise.
Friction+damping + noise + latency	$\lambda = 2$ friction and damping randomization with both observation noise and randomized latency.
Controlled $\lambda = 3$	Physical-range extension with frozen observation noise and actuator latency fixed at 0 ms.
Controlled $\lambda = 4$	Physical-range extension with frozen observation noise and actuator latency fixed at 0 ms.
Reference centre + noise	Final grouped ablation with friction and damping randomization OFF.
Damping + noise	Final grouped ablation with damping randomization only.
Friction + noise	Final grouped ablation with friction randomization only.
Friction+damping + noise	Final selected grouped-ablation condition.

APPENDIX B

Neural Network Hyperparameters

B.1 Final Input Representation

The final classifier uses the following twelve input features:

$$[F_x,\, F_y,\, M_z,\, \tau_1,\, \tau_2,\, \tau_3,\, d_1,\, d_2,\, d_3,\, \alpha_1,\, \alpha_2,\, \alpha_3]^{\mathsf{T}}\,. \tag{B.1}$$

The end-effector pose is not supplied directly to the classifier, although it remains available internally for geometric feature construction.

B.2 Network Architecture

All controlled experiments use the same feedforward network:

$$12 \to 128 \to 64 \to 32 \to 7. \tag{B.2}$$

The layer sizes, activation functions, loss, and optimizer are listed in Table B.1.

Table B.1: Neural-network architecture.

Component	Specification
Input dimension	12
Hidden layer 1	128 neurons, ReLU
Hidden layer 2	64 neurons, ReLU
Hidden layer 3	32 neurons, ReLU
Output layer	7 logits
Loss function	Cross-entropy loss
Optimizer	Adam

B.3 Training and Preprocessing Settings

For reproducibility, the fixed training and preprocessing settings are listed in Table B.2.

Table B.2: Controlled preprocessing and training settings.

Setting	Value
Random seed	42
Device	CPU
CPU threads	8
PyTorch version	2.0.0.post200
DataLoader workers	0
Deterministic algorithms	Enabled
Maximum samples per rollout	10
Train/validation split	90/10 at rollout level
Batch size	64
Learning rate	0.001
Maximum epochs	100
Early-stopping patience	10
Checkpoint criterion	Minimum simulation-validation loss
Standardization	Fitted on simulation training data only

APPENDIX C

Additional Evaluation Results

This appendix collects detailed metrics that support the main comparisons in Chapter 5. Keeping them here avoids interrupting the main Results chapter with class-by-class and experiment-by-experiment detail.

C.1 Full Aggregate Metric Tables

Tables C.1 to C.4 provide the complete aggregate metrics for the initial coupled sweep, noise–latency ablation, controlled physical-range extension, and final friction–damping study.

Table C.1: Complete metrics for the initial coupled randomization-strength sweep.

Experiment	Sim. acc.	Sim. macro F1	B1 acc.	B1 macro F1	B2 acc.	B2 macro F1	Avg. bench acc.
Nominal baseline	98.1633	98.1583	63.5105	61.4239	70.8664	70.5851	67.1884
Fixed reference	98.1024	98.0991	60.7988	56.5909	66.7614	65.9379	63.7801
$\lambda = 0.5$	98.0663	98.0620	59.2317	55.4329	66.7182	66.2386	62.9749
$\lambda = 1$	97.7345	97.7303	59.6718	55.5427	70.7483	71.2423	65.2100
$\lambda = 2$	97.3580	97.3549	64.3371	61.1521	70.1124	70.9952	67.2248

Table C.2: Complete metrics for the controlled noise–latency ablation at the $\lambda = 2$ friction-and-damping ranges.

Condition	Sim. macro F1	B1 acc.	B1 macro F1	B2 acc.	B2 macro F1	Avg. acc.	Avg. macro F1
Friction+damping, clean	98.1484	61.3769	57.8807	71.0069	70.5176	66.1919	64.1992
Friction+damping + noise	96.8295	66.2931	63.9282	76.9909	76.7398	71.6420	70.3340
Friction+damping + latency	98.4046	61.9602	58.8158	70.8296	69.9184	66.3949	64.3671
Friction+damping + noise + latency	97.1689	58.7502	55.4771	66.5280	67.6150	62.6391	61.5460

Table C.3: Complete aggregate metrics for the controlled physical-range extension. Observation noise is frozen and actuator latency is fixed at 0 ms.

λ	B1 acc.	B1 macro F1	B2 acc.	B2 macro F1	Avg. acc.	Avg. macro F1
2	66.2931	63.9282	76.9909	76.7398	71.6420	70.3340
3	64.8343	60.8926	69.8675	70.8378	67.3509	65.8652
4	65.7284	62.1763	74.8234	75.5952	70.2759	68.8858

Table C.4: Complete aggregate metrics for the final friction–damping ablation. Observation noise is fixed ON and actuator latency is fixed at 0 ms for every condition.

Condition	B1 acc.	B2 acc.	Avg. acc.	Avg. macro F1
Reference centre + noise	61.1004	72.0586	66.5795	65.0329
Damping + noise	59.6147	72.2638	65.9393	64.1280
Friction + noise	58.0045	67.9578	62.9812	61.1259
Friction+damping + noise	66.2931	76.9909	71.6420	70.3340

C.2 Class-Wise Effect of the Final Friction–Damping Condition

Table C.5 compares the friction/damping-off condition with the final friction + damping condition. Observation noise is fixed in both cases. The average delta is the mean of the Bench Set 1 and Bench Set 2 F1 changes for each class.

Table C.5: Class-wise F1 changes from the friction/damping-off condition to the final friction + damping condition on the two physical benchmarks.

Class	B1 F1 (F/D off)	B1 F1 (friction + damping)	B2 F1 (F/D off)	B2 F1 (friction + damping)	Avg. delta
Mobile platform	41.14	47.10	61.17	71.42	+8.105
C1L1	70.87	82.11	85.01	86.01	+6.120
C1L2	49.38	53.02	51.87	55.07	+3.420
C2L1	16.71	32.80	93.61	97.33	+9.905
C2L2	52.50	58.78	57.45	66.39	+7.610
C3L1	96.23	95.54	90.60	90.72	-0.285
C3L2	77.20	78.14	66.71	70.23	+2.230

C.3 Confusion Matrices

The first figure compares the Base-XML nominal baseline with the final selected friction + damping + noise model. The remaining figures document the other controlled noise–latency conditions. All plots use the same row-normalized scale from 0 to 1, with rows denoting true classes and columns denoting predicted classes.

Figures C.1 to C.4 show the complete set.

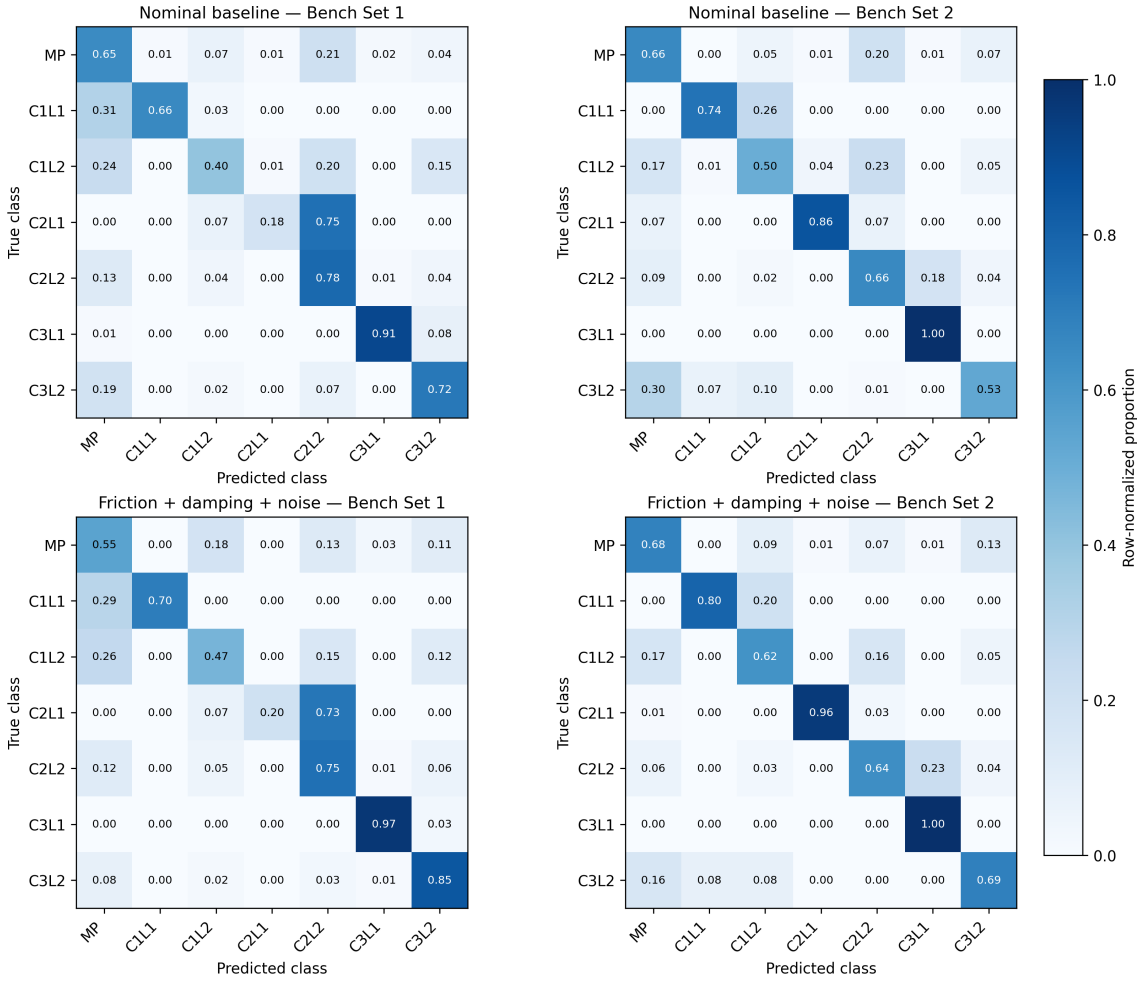


Figure C.1: Row-normalized confusion matrices for the Base-XML nominal baseline and the final selected friction + damping + noise model on both physical benchmarks. A common 0–1 scale is used in all panels.

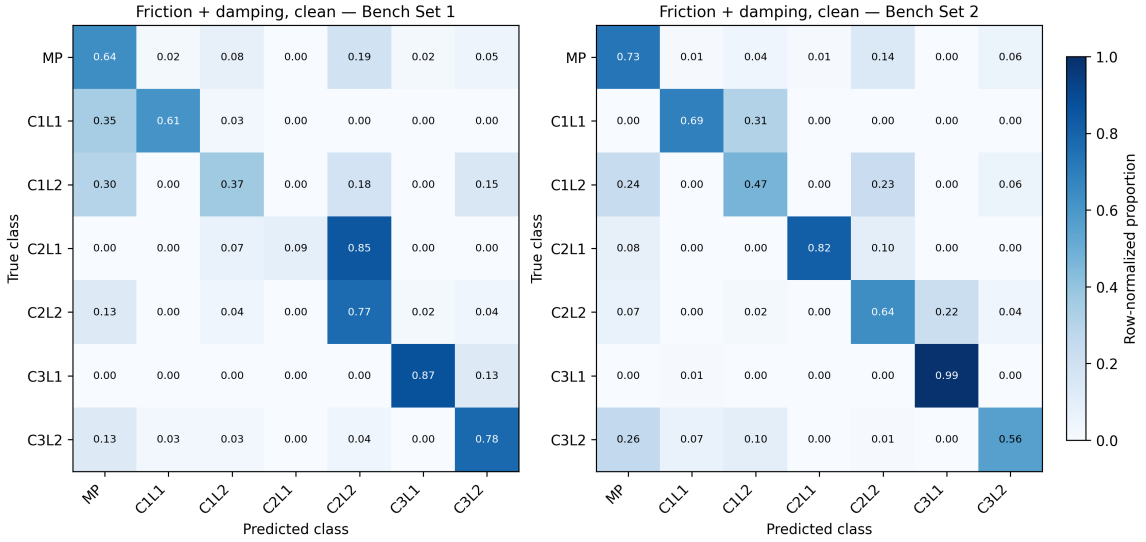


Figure C.2: Row-normalized confusion matrices for the clean friction + damping condition on Bench Set 1 and Bench Set 2. The class-wise performance remains uneven without observation noise.

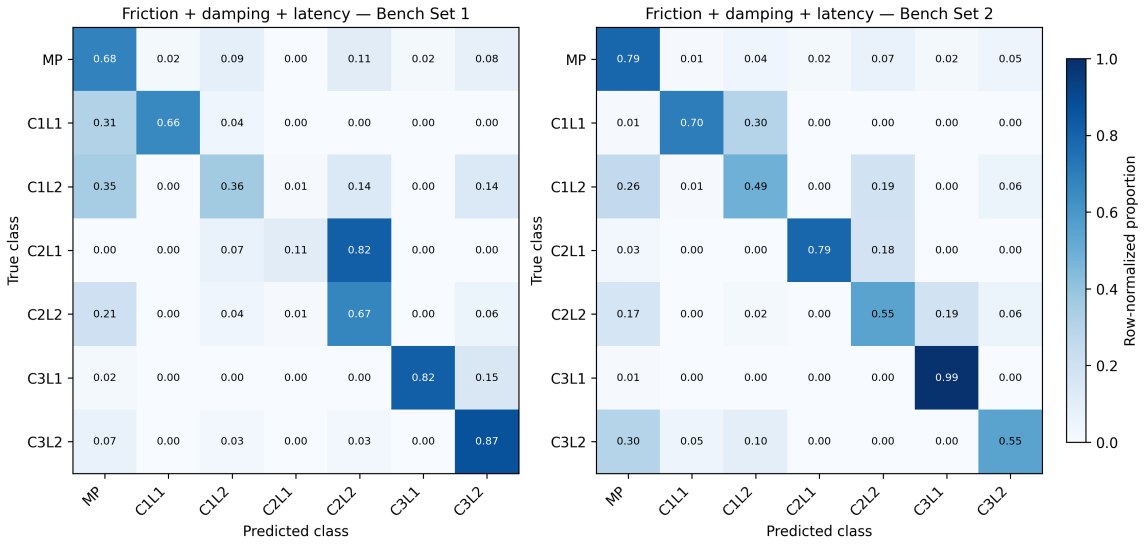


Figure C.3: Row-normalized confusion matrices for the friction + damping + latency condition on Bench Set 1 and Bench Set 2. The tested latency changes some class-level errors but does not materially improve the average physical accuracy.

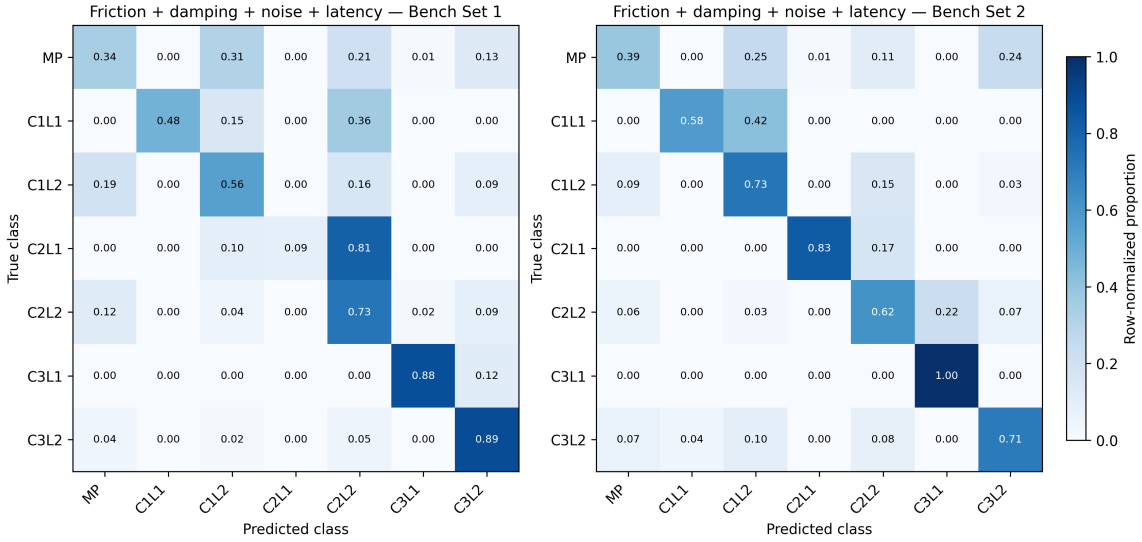


Figure C.4: Row-normalized confusion matrices for the friction + damping + noise + latency condition on Bench Set 1 and Bench Set 2. The combined noise–latency setting produces broader errors than the selected noise-only model, consistent with its lower average accuracy.

References

[1] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot Collisions: A Survey on Detection, Isolation, and Identification," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.

[2] A. Mohammad, M. Schappler, and T. Ortmaier, "Collision Isolation and Identification Using Proprioceptive Sensing for Parallel Robots to Enable Human–Robot Collaboration," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5910–5917, 2023, doi: 10.1109/IROS55552.2023.10342345.

[3] X. Chen, J. Hu, C. Jin, L. Li, and L. Wang, "Understanding Domain Randomization for Sim-to-Real Transfer," in *International Conference on Learning Representations (ICLR)*, 2022.

[4] W. Zhao, J. Peña Queralta, and T. Westerlund, "Sim-to-Real Transfer in Deep Reinforcement Learning for Robotics: A Survey," *arXiv preprint arXiv:2009.13303*, 2021.

[5] A. Giourgas, "Investigation of the Usage of Transfer Learning in Contact-Body Classification for Human–Robot Collaboration," Master's thesis, Leibniz University Hannover, 2024.

[6] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30, 2017, doi: 10.1109/IROS.2017.8202133.

[7] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-Real Transfer of Robotic Control with Dynamics Randomization," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3803–3810, 2018, doi: 10.1109/ICRA.2018.8460528.

[8] D. Popov, A. Klimchik, and N. Mavridis, "Collision Detection, Localization & Classification for Industrial Robots with Joint Torque Sensors," in *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 838–843, 2017.

[9] L. Manuelli and R. Tedrake, "Localizing External Contact Using Proprioceptive Sensors: The Contact Particle Filter," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5062–5069, 2016.

[10] A. Zwiener, C. Geckeler, and A. Zell, "Contact Point Localization for Articulated Manipulators with Proprioceptive Sensors and Machine Learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 323–329, 2018.

[11] N. Bouraoui, "Application of Time Series Classification in Human-Robot Collaboration with Parallel Robots," Master's thesis, Leibniz University Hannover, 2025.

[12] N. Jaquier, M. C. Welle, A. Gams, K. Yao, B. Fichera, A. Billard, A. Ude, T. Asfour, and D. Kragic, "Transfer Learning in Robotics: An Upcoming Breakthrough? A Review of Promises and Challenges," *The International Journal of Robotics Research*, vol. 44, no. 3, pp. 465–485, 2025, doi: 10.1177/02783649241273565.

[13] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-Adversarial Training of Neural Networks," *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1–35, 2016.

[14] F. Muratore, C. Eilers, M. Gienger, and J. Peters, "Data-Efficient Domain Randomization with Bayesian Optimization," *arXiv preprint arXiv:2003.02471*, 2021.

[15] A. Bauer, D. Wollherr, and M. Buss, "Human–Robot Collaboration: A Survey," *International Journal of Humanoid Robotics*, vol. 5, no. 1, pp. 47–66, 2008.

[16] D. Kulić and E. A. Croft, "Real-Time Safety for Human–Robot Interaction," *Robotics and Autonomous Systems*, vol. 54, no. 1, pp. 1–12, 2006, doi: 10.1016/j.robot.2005.10.005.

[17] A. Mohammad, M. Schappler, and T. Ortmaier, "Towards Human–Robot Collaboration with Parallel Robots by Kinetostatic Analysis, Impedance Control and Contact Detection," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 12092–12098, 2023, doi: 10.1109/ICRA48891.2023.10161217.

[18] A. Mohammad, H. Muscheid, M. Schappler, and T. Seel, "Quantifying Uncertainties of Contact Classifications in a Human-Robot Collaboration with Parallel Robots," in *Human-Friendly Robotics 2023*, 2023.

[19] H. A. Pierson and M. S. Gashler, "Deep Learning in Robotics: A Review of Recent Research," *Advanced Robotics*, vol. 31, no. 16, pp. 821–835, 2017, doi: 10.1080/01691864.2017.1365009.

[20] D. Nguyen-Tuong and J. Peters, "Using Model Knowledge for Learning Inverse Dynamics," in *2010 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2677–2682, 2010.

[21] R. Calandra, S. Ivaldi, M. P. Deisenroth, E. Rueckert, and J. Peters, "Learning Inverse Dynamics Models with Contacts," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3186–3191, 2015, doi: 10.1109/ICRA.2015.7139726.

[22] E. Aljalbout, J. Xing, A. Romero, I. Akinola, C. R. Garrett, E. Heiden, A. Gupta, T. Hermans, Y. Narang, D. Fox, D. Scaramuzza, and F. Ramos, "The Reality Gap in Robotics: Challenges, Solutions, and Best Practices," preprint to appear in *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 9, 2026.

[23] Google DeepMind, "MuJoCo Documentation: Solver Parameters," online documentation, accessed Aug. 11, 2026.