Improved Friction and Dynamics Estimation in Legged Robots

By

Laura Schwendeman

Submitted to the
Department of Mechanical Engineering
In Partial Fulfillment of the Requirements for the Degree of

BACHELOR OF SCIENCE IN MECHANICAL ENGINEERING

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2023

Authored by: Laura Schwendeman
                Department of Mechanical Engineering
                May 12, 2023

Certified by: Sangbae Kim
                Professor of Mechanical Engineering,
                Thesis Supervisor

Accepted by: Kenneth Kamrin
                Undergraduate Officer

Improved Friction and Dynamics Estimation in Legged Robots

by

Laura Schwendeman

Submitted to the Department of Mechanical Engineering
on May 12, 2023, in partial fulfillment of the
requirements for the degree of
Bachelor of Science in Mechanical Engineering

Abstract

Reducing the Sim-to-Real gap between robot simulation and robot performance could lead to improved and more efficient robot design through more accurate controls and design testing in simulation and through more accurate state detection for model-based control architectures. This work built upon current research in the field of robot system dynamics by investigating the effect of using single-layer feed-forward neural nets to model non-linear friction forces and other forms of dynamics that are difficult to account for with traditional robot system identification schemes. Applying the single-layer feed-forward neural nets to system identification data from the dynamic MIT Humanoid and MIT Mini Cheetah robots significantly reduced torque prediction errors. The neural net was able to reduce torque errors by modeling both linear and non-linear effects that could not be easily fit by traditional methods. The results of this paper suggest that using the system identification methodology outlined within could lead to more accurate dynamics modeling, which would assist with closing the Sim-to-Real gap through simulated dynamics with more fidelity and a more robust representation of robot dynamics.

Thesis Supervisor: Sangbae Kim
Title: Professor of Mechanical Engineering

Acknowledgments

The author would like to thank Andrew SaLoutos and the rest of the Biomimetics Robots Lab for mentoring her throughout this research and making this thesis possible.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The Sim-to-Real gap between simulated and physical robot performance presents a problem for robot controls engineers and researchers. To control complex robots performing complex tasks, often engineers will first design and test control systems in simulation software packages. This helps roboticists gain an understanding of the efficacy of their control structures through simulated dynamics before applying said control structures to physical hardware that is more costly to test, especially in the event of catastrophic failures. Vetted control schemes are then applied in physical robotic systems and further tuned and optimized. However, since the dynamics coded in simulation are often not entirely reflective of true physical systems, the transition from applying a control scheme generated through simulation to a physical robot can be time-consuming and lead to control that is much less effective than what was predicted in simulation. Especially in popular dynamic control schemes with model-based components, the differences between simulated and physical dynamics, which are integral to the control scheme itself, can lead to vastly different results between simulated and real performances. Such differences can even persist with corrective learning control methods like reinforcement learning that seek to edit control automatically based on performance metrics and errors.

Many research papers from a variety of different fields have developed techniques to make simulated physics more accurately reflect and predict true physical results. With respect to dynamic-legged robots, Lee et al. recently developed a method using a regularized geometric convex optimization approach with a least squares formulation and additional

physical constraints [4]. This approach uses a given robot model and can fit robot inertial parameters while accounting for the pitfalls that traditional robot inertial fitting methods fall into. However, this approach is still linear in nature, and its accuracy still depends on the robustness of the underlying robot system model that describes which inertial and dynamic parameters to fit in the first place. Any un-modeled or non-linear effects will create errors in torque predictions.

This work theorized then that including more detailed, non-linear effects measured during system identification in the control design pipeline would not only help diminish the discrepancies between simulated and physical control performance but also improve the performance of controllers that depend on dynamics modeling to connect input commands to controlled movements and torques. Particularly, this work focuses on internal, non-linear friction-based dynamic effects that often are unaccounted for in model-based control schemes. This work investigated applying neural networks in addition to the regularized convex approach developed by Lee et al. to model the largely friction-based dynamic effects that are difficult to accurately include in simulated robot models built from traditional system identification methods alone [4]. This work included non-linear friction estimation in both the MIT Mini Cheetah and MIT Humanoid robots' system identification pipeline with physical and simulated data sets. Including said non-linear dynamic effects in simulated and physical dynamic state estimation led to reduced torque prediction errors and was able to model more complex dynamics on robots with coupled dynamics and non-linear belt frictions. Such results suggest that including non-linear friction estimation in robot control could lead to improved performance and potentially reduce the need for aggressive tuning and correction when transitioning controllers between simulated and physical robots.

Chapter 2

Background

Determining the dynamic parameters and responses of a system through testing is intrinsic to the understanding and development of many physical systems. System identification finds relevance in fields ranging from neuromuscular biomechanics to industrial robotics to drone control. System identification is particularly important in the development of robotic control systems because many control schemes generate commands based on a simulated physical model of the system in question. Classically, a least squares approach has been used to fit system identification data to robot model parameters.

2.1 Classic Least Squares Approach

Starting in the 1980s, popular methods for estimating robot dynamic parameters included a suite of Least-Squares based regression methods. Researchers like Gautier et al. investigated techniques for applying least-squares and getting inertial parameters from physical robot data with optimized trajectories [2]. Similarly, Swevers et al. used least-squares-based linear techniques with a method for fitting parameters in the presence of noise and with a unique optimized robot trajectory. Swevers et al. were able to closely estimate the inertial parameters of industrial robots with this method [6]. These initial works and others like them were largely designed for and tested on industrial robots. They create a linear fit in terms of robot inertial parameters using excitatory robot dynamics data.

Different traditional papers will formulate a given robot's least squares regression sys-

tem identification using different models of rigid body dynamics. For example, Gautier et al. describe their robot dynamics through Hamiltonians [2]. However, regardless of the fundamental physics equations used to correlate measured kinematic parameters to robot forces, a classic least squares regression in robotics will follow the form of:

$$\tau = A\Phi \tag{2.1}$$

Where τ refers to an n-sized vector of robot torque measurements, Φ refers to an r-sized vector of relevant inertial parameters and linear friction coefficients, and A refers to a n-by-r matrix describing relevant model speeds, velocities, and acceleration measures depending on the choice of physics model. In a robot system's identification problem then, the goal is to fit values to the numbers in Φ that reduce the error between the measured τ values and those predicted by a linear fit of the data. This creates the least squares optimization problem depicted in 2.2.

$$\min_{\Phi} \quad \|A\Phi - \tau\| \tag{2.2}$$

Solving this least sqaures problem for Φ then gives a set of parameters that can be used to estimate and model robot joint torques given input joint velocities, positions, and/or accelerations. Such approaches using dynamics parameters with such a least squares formulation, with additional methods to account for measurement noise, have been successful in simulating the physical dynamics of industrial robots as evinced by Gautier et al., Swevers et al., and other authors [2–4, 6].

2.2   Regularized Convex Least Squares Approach

For cases outside industrial robots, such as in legged robots with more complex mechanisms and environments, traditional least-squares approaches do not efficiently account for the more complicated dynamics, increased degrees of freedom, and the large magnitude ranges of velocities and inertias present between different links in a legged robot [1]. To account for the difficulties in applying traditional methods to legged robots, many re-

searchers have experimented with new approaches for inertial parameter system identification. In particular, Lee et al. formulated and tested a convex geometric regularization approach that made use of an entropic divergence distance metric to regularize and optimize robot inertial parameters within a least squares optimization method [1]. Such an approach was able to model, among other systems, a 3 DoF mini-cheetah robot leg with reduced errors when compared to traditional methods. In Lee et al.'s work, they dynamics of a system were modeled using equation 2.3:

$$\tau = M(q,\Phi)\ddot{q} + b(q,\dot{q},\Phi) = \Gamma(q,\dot{q},\ddot{q})\Phi \tag{2.3}$$

Where τ represents robot joint torques, M describes the mass matrix of the system, q refers to joint position, b describes gravitational and Coriolis forces, and Γ is a regressor matrix that puts equation 2.3 into a factored form similar to equation 2.1. Φ then is a vector of the inertial parameters for each moving rigid body in the system, and Φ includes the components of each body's inertia tensor in addition to its mass and an indicator of each body's center of mass [4].

To fit experimental data to equation 2.3, Lee et. al then formalized a least squares minimization problem with an extra regularization factor that penalizes divergence from an initial guess at the system's inertial parameters and the inertial parameter's divergence from physically feasible parameters. The formulation is described in equation 2.4:

$$min_{\Phi} \quad \|\Gamma(q,\dot{q},\ddot{q})\Phi - \tau\|_{C^{-1}}^2 + \gamma \cdot d(\Phi,{}^0\Phi)^2 \tag{2.4}$$

Where C, the error covariance, is based on a non-regularized data fit, γ is a weight that emphasizes the importance of adherence to the initial guesses of inertial parameters ${}^0\Phi$, and $d(\Phi,{}^0\Phi)$ describes a regularized distance metric between ${}^0\Phi$ and the newly optimized Φ. In the Lee et al. paper, a variety of different distance metrics, $d(\Phi,{}^0\Phi)$, were used. This work specifically utilizes an entropic Bregman distance metric which is described by Lee et al. [4]. Using the entropic distance metric reduced the torque prediction error in optimizing Φ when compared to other distance metrics like the Euler distance, and it especially improved torque estimation at lower sample sizes from MIT Mini Cheetah leg

data in addition to other system models [4].

The convex approach developed by Lee et al. promises to be useful for estimating robot inertial parameters in a quick and more robust way for legged robots and more complicated systems. However, while the convex regularization approach succeeds in finding the inertial parameters of models with high degrees of freedom and more complicated coupled dynamics, it does not account for non-linear dynamic effects present in larger legged robots such as the MIT Humanoid.

2.3 Friction Estimation Techniques

This work seeks to include a complimentary module that can work in tandem with an approach like Lee et al.'s convex regularization algorithm to account for the non-linear dynamics and errors that an approach solving for inertial parameters alone cannot easily model. Commonly difficult physical parameters to model include torque-dependent friction and non-linear friction-based effects such as Stribeck friction and belt friction.

Recently Huang et al. developed a method employing a small neural network with backpropagation in conjunction with a hybrid least squares algorithm to estimate robot friction torques unaccounted for by the hybrid least squares algorithm alone. Such a method used a single-layer neural net with 10 neurons and was able to reduce the torque prediction errors in a 7 DoF Fanka Emika Panda and UR5 Robot [3]. This work thus sought to see if similarly adding a small artificial neural network with backpropagation to robot system identification in addition to Lee et al.'s convex regularization approach could improve the torque errors present in the MIT Cheetah and MIT Humanoid robots, which differ from the fixed base Panda and UR5 robots. Additionally, this work also sought to better understand if improving torque predictions on a model level would result in more accurate control and tracking in the MIT Cheetah and MIT Humanoid robots.

Chapter 3

Methods

3.1 Robot Modeling

3.1.1 Rigid Body Dynamics Models

The approach developed by Lee et. al. for estimating robot inertial parameters depends on underlying robot kinematic models to provide constraints and relationships between fit robot inertial parameters. The choice of model can thus greatly affect the results of the optimization. In this work, two different robot systems were investigated to understand the efficacy and effects of employing friction compensation in robot system identification. Initial data was collected with a 3 DoF leg from a MIT Mini Cheetah robot to test the effects of friction compensation on a lower DoF system with smaller inertias and lower torque errors due to friction. Data was also collected with the 5 DoF left leg of the MIT Humanoid robot to further test the effects of friction compensation in a robot with more complicated coupled dynamics, larger inertias, and harder to model effects from belt and torque dependant frictions.

Rigid Body Dynamics (RBD) models, shown in Figure 3-1, were made for both robot systems in order to describe the robot kinematics and initial prior parameters used for inertial parameter optimization. For the MIT Mini Cheetah, the same RBD model that was used in the Lee et al. paper was employed. This model includes the inertias of three rigid links and 3 rotors whose dynamics are coupled between the second and third link. For

the MIT Humanoid Robot, three different models were investigated in order to determine which model could best reduce torque prediction errors and to analyze the differences between performing system identification with simulated and physical data. One model, the Hardware model, is a representation of the humanoid robot's kinematics that can best describe the geometries and coupled dynamics of lower leg links. The Hardware Model for the MIT Humanoid includes belt ratios, rotor inertias, and a representation of the coupled dynamics between the lower leg that arise due to off-joint actuation through belt drives. The other models used in this work, the Isaac Gym model and the Robot Software model represent simplified versions of the MIT Humanoid that are used for simulating control strategies and dynamics in different software systems. The Robot Software model includes rotor inertias and dynamics but does not include the coupled dynamics between links or the effects of belt gear ratios. Each rotor and link pair are modeled as directly driven joints. The Issac Gym model then further simplifies the MIT Humanoid to just its rigid links and ignores the effects of separate rotor inertias.

For the MIT Humanoid, all three models were investigated to determine the minimum level of complexity required to accurately describe generated robot motor torques. The results of fitting inertial parameters to the different MIT Humanoid models with different motor torque datasets are shown in section 4.2.1.

3.1.2   Inertial Parameter Regression

The approach described by Lee et. al generates bounding ellipsoids based on physical robot parameters in order to constrain the set of optimized inertial parameters to ones that are more physically realistic. In this work, bounding ellipsoids were generated in order to encompass the range of inertias within the constraints of the robot's geometric bounds taken from CAD software. These bounds are graphically depicted in Figure 3-1 for each RBD model used in this work.

Additionally, the approach described by Lee et. al. includes a regularization term that weights the optimizations divergence from the inertial parameters first fed into the parameter optimization algorithm. In this work, the initial parameters (referred to as priors) were

16

(a)

(b)

(c)

(d) .

Figure 3-1: Graphical depictions of the RBD models and their bounding ellipsoids used to represent and fit the intertias of a MIT Mini Cheetah leg and a MIT Humanoid leg. Rotor inertias, if present, are shown in translucent black. Colored ellipses represent bounding ellipses for robot links. Dark black lines denote the relative coordinate frames of each link. (a) RBD model of the MIT Mini Cheetah with 3 DoF and 3 rotors in their true locations in CAD. (b) Isaac Gym RBD model of the MIT Humanoid with 5 DoF and no rotors. (c) Robot Software RBD model of the MIT Humanoid with 5 DoF and 5 rotors placed at each joint (d) Hardware RBD model of the MIT Humanoid with 5 DoF and 5 rotors in their true locations.

taken from the estimated link weights and dimensions taken from CAD representations of the robots studied. The regularization parameter γ was set to 0.001 for all optimizations in this work.

In this work, nearly the same approach and optimization formulation developed by Lee et. al. and described in section 2.2 was used to fit inertial parameters to robot system identification data. The only considerable difference was the exclusion of the friction terms from the Γ function described in equation 2.3. As shown in figure 4-3, the effect of including or excluding the friction terms in the Γ function have a relatively small effect on the optimal parameters found through Lee et. al's optimization approach.

3.2   Neural Network Training and Dynamics Identification

Once inertial parameters were fit using Lee et al.'s regularized optimization approach, a single-layer feed-forward neural net with 16 neurons was trained on the resulting error between the predicted torque from optimization and the true, measured torque. The neural nets for each robot in each data set were given a vector with the recorded joint angles, velocities, and torques. The number of inputs to each net, n, was equal to the number of degrees of freedom for each robot times three, where the joint angles, joint velocities, and torques for each DoF's controlling motor constituted the input. The neural nets then outputted m friction torques where m is equal to the number of DoF in the system. Each neural net was trained on 50% of the data with 20% reserved for validation and 30% reserved for testing. The neural net was implemented using Matlab's learning toolbox, using the default parameters coded for a feedforward neural network shown in table 3.1.

Table 3.1: Neural Net Parameters

| Neural Net Parameter	Value
Training Algorithm	Levenberg-Marquardt
Performance Metric	Mean Squared Error
Max Number of Epochs	1000
Gradient Threshold	10^{-7}
μ Threshold	10^{10}

3.3 System Identification Data Collection

System identification was performed with both simulated and physical excitatory data on both a MIT Mini Cheetah leg and the left leg of the MIT Humanoid. Swept simulated data and oscillatory fixed speed data were collected from the MIT Biomimetics Robotics Lab's Robot Software simulation package. Robot Software used the same Robot Software MIT Humanoid and MIT Mini Cheetah models as this work to generate oscillatory robot paths with added noise on all 3 motors of the Mini Cheetah and all 5 motors of the MIT Humanoid. The measured motor torques resulting from swept simulations for the MIT Humanoid and constant frequency simulations for the Mini Cheetah were recorded and are shown in Figure 3-2.

Physical excitatory data was also collected for both the MIT Humanoid and MIT Cheetah robots (see Figure 3-4). Both robots were commanded to generate motor torques following swept sine paths at each motor. The recorded motor torques for each data set and robot platform are shown in Figure 3-3.
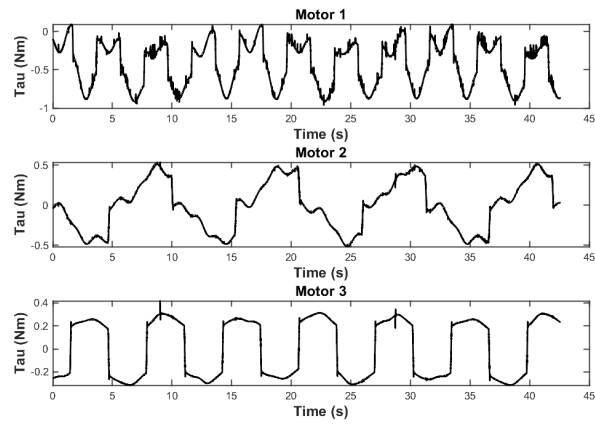
(a)
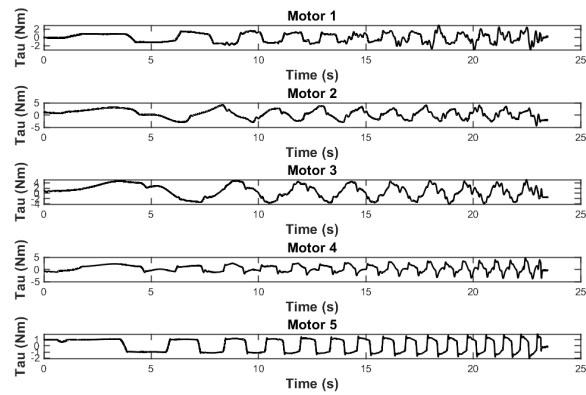
(b)

Figure 3-2: (a) Measured simulated torque data from commanding the MIT Mini Cheetah to follow a constant frequency sinusoid. (b) Simulated motor torques for the MIT Humanoid robot when commanding the robot joints to follow swept oscillations.
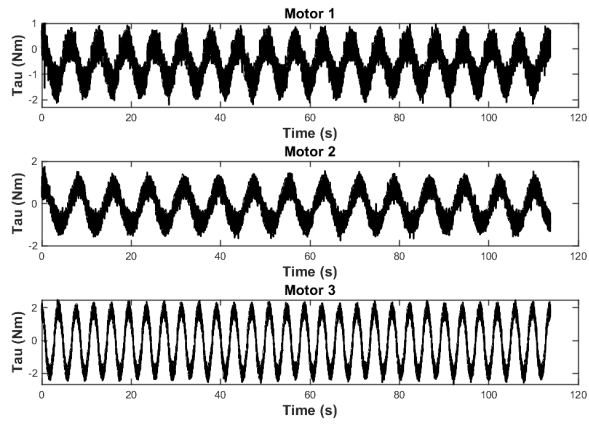
(a)
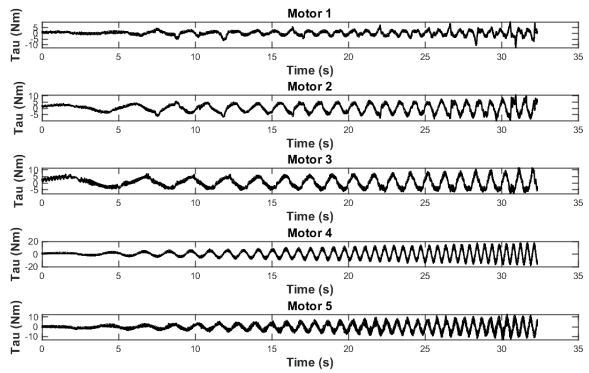
(b)

Figure 3-3: (a) Measured physical torque data from commanding the MIT Mini Cheetah to follow a constant Frequency Sinusoid. (b) Physical motor torques for the MIT Humanoid robot when commanding the robot joints to follow swept oscillations.

(a)   (b)

Figure 3-4: The two robot systems used in this work. (a) the MIT Mini Cheetah. (b) a rendering of the full MIT Humanoid. For this work, the left leg of the MIT humanoid was modeled and tested.

Chapter 4

Results

4.1 MIT Mini Cheetah Leg Results

4.1.1 Simple Nets Improve Torque Predictions

To first test the efficacy of a neural net in estimating robot friction torques, this work investigated the improvement of torque estimation for a 3 DoF MIT Mini Cheetah Leg. As a first pass, simulated MIT Mini Cheetah Leg oscillating position, velocity, and acceleration data were taken and frictionless torque time series were generated for each motor using known simulated Mini Cheetah inertial parameters. Friction torques using a non-linear model of Stribeck friction described by Marton and Lantos were then added to the torque time series data along with Coulomb friction and viscous joint friction torques [5]. As shown in Figure 4-1, the single-layer neural net described in section 3.2 was able to reduce prediction errors by nearly perfectly matching the simplified added friction torques.

The neural net framework was then tested on filtered physical Mini Cheetah leg dynamics data. Figure 4-2 shows similarly that applying a neural net for friction compensation was able to greatly reduce the torque prediction errors of the Mini Cheetah model.

4.2 MIT Humanoid Leg Results

4.2.1 Fitting Inertial Parameters

After verifying that neural nets could predict the friction torques present a 3 DoF MIT Mini Cheetah Leg at varying speeds, the MIT Humanoid was analyzed in a similar manner, in more depth, and across different potential models.

Figure 4-3 shows the entropic Bregman Distance between the prior parameters fed into optimizations with or without linear friction and Coulomb friction terms included in the cost function. As shown in the figure, the difference between a given model's inertial parameter fitting when the linear dynamic and Coulomb friction terms were or were not included was very small compared to the distance between the optimized parameters and the initial parameters used at the beginning of optimization. The small regularized distances between optimizations with and without linear friction terms generally held true across all models and data sets. The presence or absence of the friction term in the regularized convex optimization did not significantly affect the results of final torque predictions, and so all following results and investigations into inertial parameter fitting did not include dynamic and Coulomb friction terms in equation 2.3 when optimizing inertial parameters on a given data set.

Figure 4-4 then compares the entropic distances of the inertial parameters fit using the three different robot models described in 3.1.1 with both simulated and physical MIT Humanoid measured torque data. As shown in Figure 4-4, the smallest distance measures are first between the distance calculated between the same inertial parameter sets running down the diagonal of the matrix. The next smallest distances then fall between the prior parameters of the Isaac Gym model and the other two MIT Humanoid models investigated in the study. This is largely an artifact of the reduced inertial bodies described in the Isaac Gym model of the MIT Humanoid because the Isaac Gym model does not have any rotors. The inertial distances calculated by the entropic Bregman distance metric for inertial parameters involving the Isaac Gym model then only compare the distances between inertias of the robot links. The reported values discount the distances between the non-existent rotors of the Isaac Gym model and the rotors of the Hardware and Robot Software models.

24

It is somewhat difficult then to make conclusions and comparisons between the Isaac Gym model and the other two MIT Humanoid models based purely on the comparisons of the distances between inertial parameters. This work thus focused on the fit parameters of the Issac Gym model in isolation from the parameters of the Robot Software and Hardware models.

Figure 4-5 highlights the distances between the priors of the Issac Gym Model and newly generated parameters from both simulated and physical data sets. Smaller distances were observed between the Isaac Gym priors and the parameters optimized from simulation data compared to those optimized using hardware torques. Interestingly, the Isaac Gym model optimized with software data had parameters that were closer to the Isaac Gym priors than the parameters that were fit with hardware data. This generally goes against the trend shown in Figure 4-6 where the largest distances exist between the priors and the optimized parameters, and parameters optimized on either data set have the smallest distances outside those between a data set and itself.

4.2.2 Neural Network Friction Estimation

As shown in Figure 4-7, applying a low-layer neural net after fitting inertial parameters can greatly improve the accuracy of torque predictions of a model on a given data set just as it did for the MIT Cheetah Leg. Tables 4.1, 4.2, and 4.3 show that for all data sets and all Humanoid models, applying a neural net to estimate friction torques reduces the root mean square errors between a model's predicted motor torques and the true measured torque. This holds true across model fits across simulated and physical MIT Humanoid data sets, suggesting that friction-based and unmodeled forces can account for a large amount of the error between measured and model-predicted torque values.

Tables 4.1, 4.2, and 4.3 also show that, on average, using the MIT Humanoid Hardware model with a neural net for friction estimation correlated with the lowest torque prediction errors on the Hardware data set compared to all other models. Comparatively, the Isaac Gym and Robot Software models with a compensatory neural net were best able to reduce the prediction errors on the simulated data sets. Additionally, the Isaac Gym and Robot

Software models both were able to reduce errors on simulated data sets to a similar extent on average, and they both had similar errors, within 15% of each other on average, on the hardware data set even with a compensatory neural net. This suggests that the coupled inertias and belt ratios modeled in the Hardware Model but not the other models may be appreciable in generating motor torques and can lead to torque prediction errors that are more easily fit by a neural net on noisy, physical MIT Humanoid torque measurements.

Tables 4.1, 4.2, and 4.3 showed that adding friction and modeling error compensation with a neural net could reduce torque prediction errors. Figures 4-8 and 4-9 qualitatively show potential mechanisms as to how the neural net's predicted friction torques differ from the method of fitting linear friction torques described by Lee et. al. One mechanism may just be a stronger fit to channel noise. However, in Figure 4-8, the neural net friction torques differ from linear ones, especially at low speeds, where the neural net fits some overshooting friction profiles that bring to mind Stribeck friction transitions and Coulomb friction forces in the sharp transition between motor torque direction changes. These characteristic effects can be seen similarly in the friction torque comparisons for motors 1, 2, and 3 in Figure 4-9, and similar, although slightly less noticeable, Coulomb friction spikes present at lower speeds on motor 5, which drives the ankle joint which has the smallest inertia out of all the links in the Humanoid Leg.

| Label	Motor 1	Motor 2	Motor 3	Motor 4	Motor 5
Simulated, no Friction	1.0095	1.0869	1.2457	1.3573	1.0622
Simulated, with Friction	0.10247	0.096516	0.049276	0.0999	0.028167
Hardware, no Friction	1.124	0.90553	1.0317	0.85934	1.1184
Hardware, with Friction	0.12809	0.15685	0.063481	0.13078	0.01765

Table 4.1: Root Mean Square Error (RSME) between the measured and predicted torques for each MIT Humanoid Motor when using the Hardware model for inertial parameter optimization on hardware and software data sets

| Label	Motor 1	Motor 2	Motor 3	Motor 4	Motor 5
Simulated, no Friction	0.95828	1.0381	1.1508	1.2139	1.0637
Simulated, with Friction	0.0768	0.078842	0.057406	0.059127	0.017483
Hardware, no Friction	1.8635	2.8796	3.1244	5.7569	3.4991
Hardware, with Friction	0.1388	0.17251	0.11898	0.15015	0.096106

Table 4.2: Root Mean Square Error (RSME) between the measured and predicted torques for each MIT Humanoid Motor when using the Robot Software model for inertial parameter optimization on hardware and software data sets

Label	Motor 1	Motor 2	Motor 3	Motor 4	Motor 5
Simulated, no Friction	0.95928	1.0383	1.0869	1.2349	1.0656
Simulated, with Friction	0.068952	0.075697	0.070939	0.054919	0.012441
Hardware, no Friction	1.7592	1.6082	1.1388	1.2741	1.0844
Hardware, with Friction	0.15327	0.15017	0.16238	0.12248	0.02019

Table 4.3: Root Mean Square Error (RSME) between the measured and predicted torques for each MIT Humanoid Motor when using the Isaac Gym model for inertial parameter optimization on hardware and software data sets
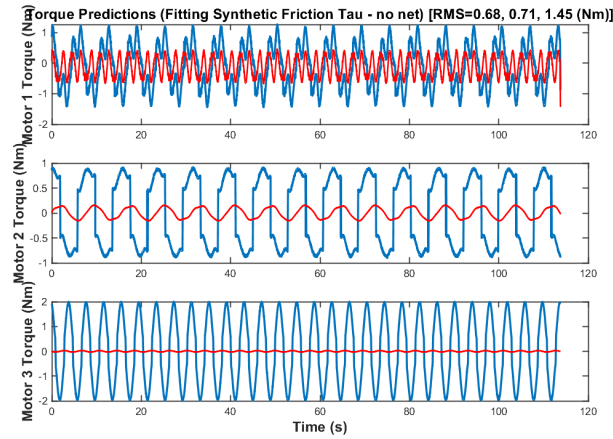
(a)
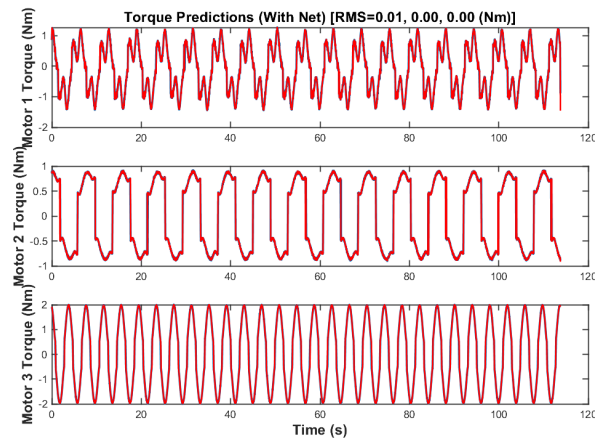
(b)

Figure 4-1: (a) A comparison of synthetic motor torques (blue) and those predicted after only fitting inertial parameters through convex optimization (red). The data shown was generated by using a set of known inertial parameters, joint velocities, and accelerations for the Mini Cheetah robot to create a time series sum of inertial torques, Coulumb frictions, Viscous frictions, and Stribeck frictions. The Root Mean Square error for the fit on motors 1, 2, and 3 of the Mini Cheetah was 0.68, 0.71, and 1.45 respectively before friction compensation. (b) A comparison of synthetic motor torques (blue) and those predicted after training a neural net on the generated friction torques (red). The predicted torques visibly nearly perfectly match the measured ones with 0.00 RMS error on all motors besides motor 1.
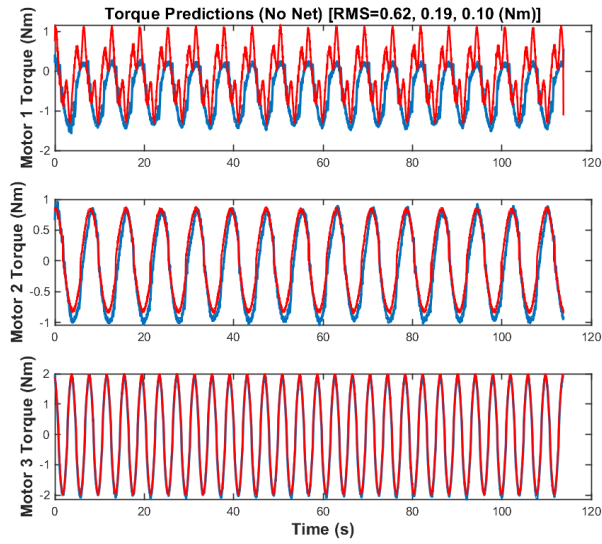
(a)
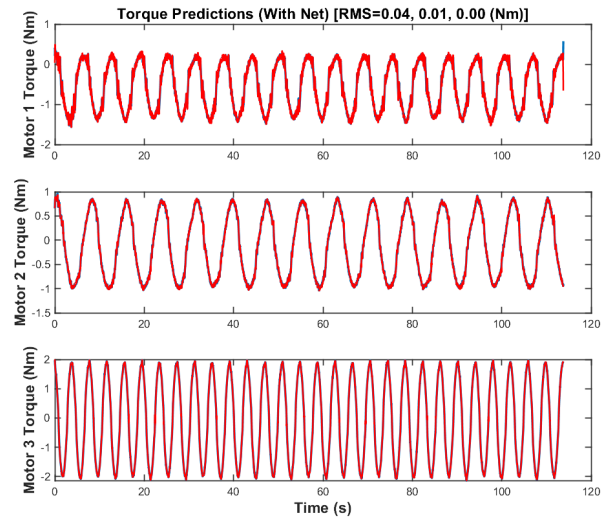
(b)

Figure 4-2: (a) A comparison of filtered measured physical Mini Cheetah motor torques (blue) and those predicted after only fitting inertial parameters through convex optimization (red). The Root Mean Square error for the fit on motors 1, 2, and 3 of the Mini Cheetah was 0.62, 0.19, and 0.10 respectively before friction compensation. (b) A comparison of synthetic motor torques (blue) and those predicted after training a neural net on the error between the measured torques and inertial torques (red). The predicted torques visibly nearly perfectly match the measured ones with RMSE values of 0.04, 0.01, and 0.0 for motors 1, 2, and 3 respectively

Log Dist Between Inertial Distributions
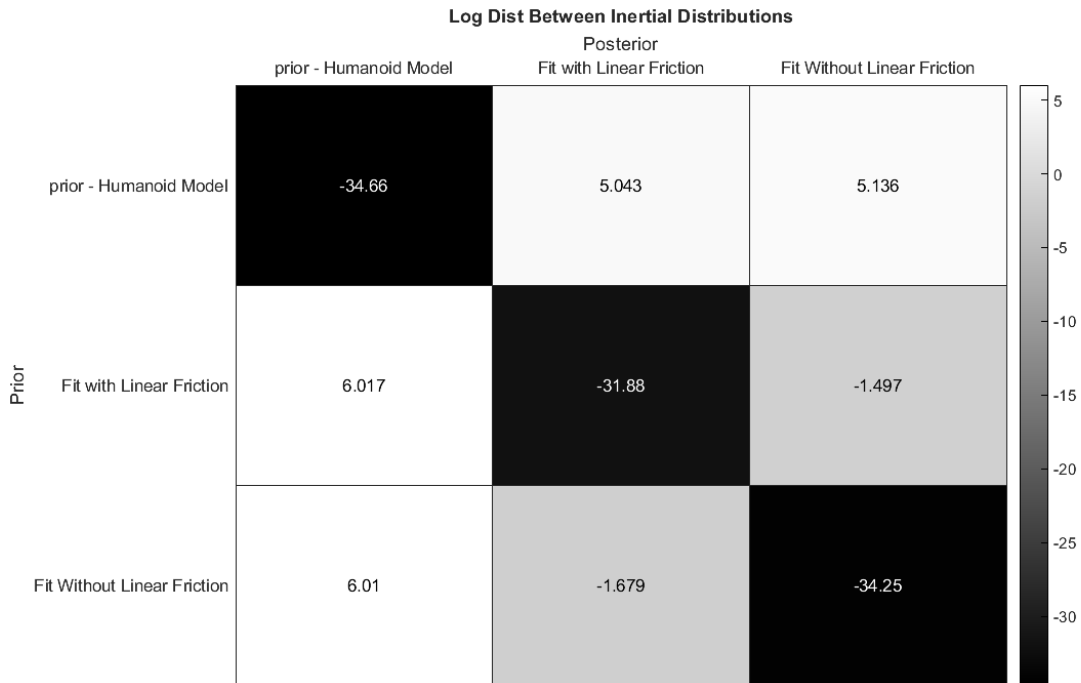
Figure 4-3: A matrix representation of the calculated Bregman distances between the prior inertial parameters and those fit from optimization using the Hardware MIT Humanoid rigid body model with MIT Humanoid Hardware data. Inertial Parameters from optimizations with dynamic and Coulomb friction terms and without any friction terms are compared. Bregman Distances are reported on a log scale.

Log Dist Between Inertial Distributions

Posterior

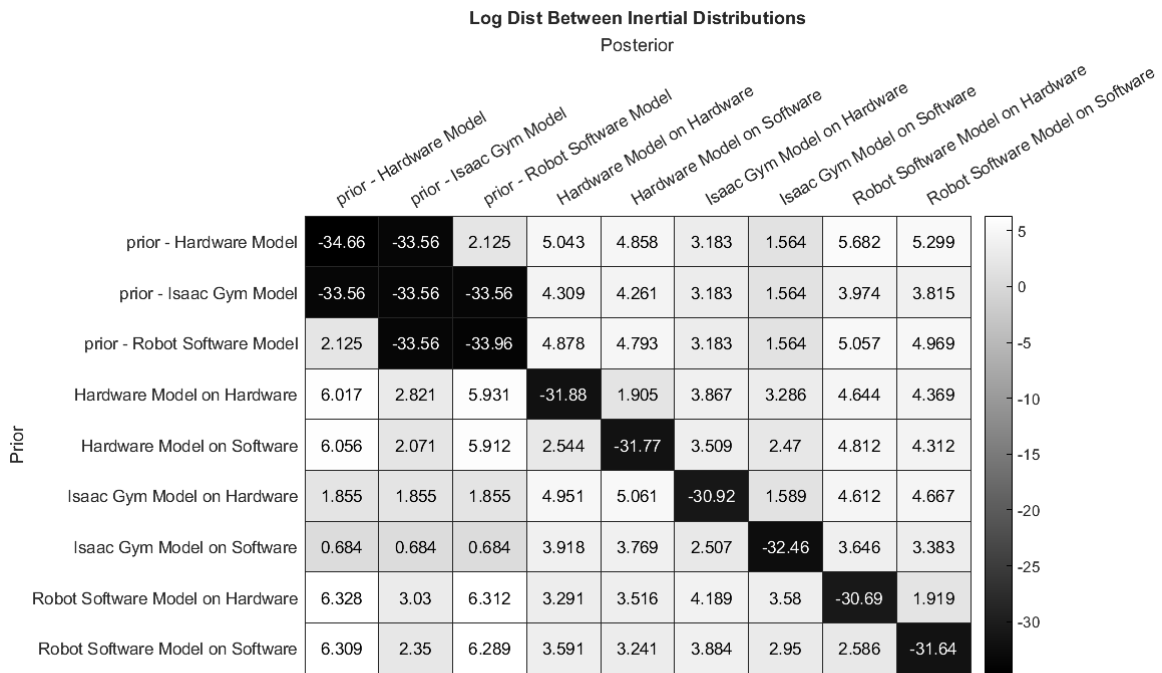| Prior	prior - Hardware Model	prior - Isaac Gym Model	prior - Robot Software Model	Hardware Model on Hardware	Hardware Model on Software	Isaac Gym Model on Hardware	Isaac Gym Model on Software	Robot Software Model on Hardware	Robot Software Model on Software
prior - Hardware Model	-34.66	-33.56	2.125	5.043	4.858	3.183	1.564	5.682	5.299
prior - Isaac Gym Model	-33.56	-33.56	-33.56	4.309	4.261	3.183	1.564	3.974	3.815
prior - Robot Software Model	2.125	-33.56	-33.96	4.878	4.793	3.183	1.564	5.057	4.969
Hardware Model on Hardware	6.017	2.821	5.931	-31.88	1.905	3.867	3.286	4.644	4.369
Hardware Model on Software	6.056	2.071	5.912	2.544	-31.77	3.509	2.47	4.812	4.312
Isaac Gym Model on Hardware	1.855	1.855	1.855	4.951	5.061	-30.92	1.589	4.612	4.667
Isaac Gym Model on Software	0.684	0.684	0.684	3.918	3.769	2.507	-32.46	3.646	3.383
Robot Software Model on Hardware	6.328	3.03	6.312	3.291	3.516	4.189	3.58	-30.69	1.919
Robot Software Model on Software	6.309	2.35	6.289	3.591	3.241	3.884	2.95	2.586	-31.64

Figure 4-4: A matrix representation of the calculated Bregman distances between the prior inertial parameters and those fit from optimization using all MIT Humanoid Models and simulated and hardware data sets. Bregman Distances are reported on a log scale.
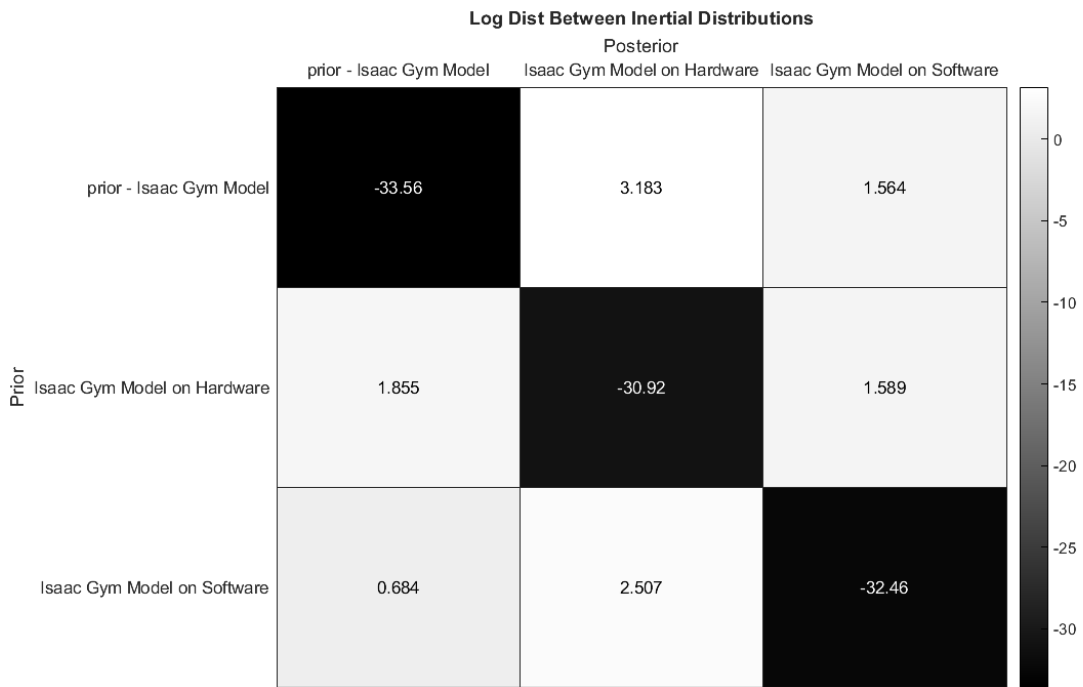
Figure 4-5: A matrix representation of the calculated Bregman distances between the prior inertial parameters and those fit from optimization using the Isaac Gym MIT Humanoid Model and simulated and hardware data sets. Bregman Distances are reported on a log scale.

Log Dist Between Inertial Distributions

Posterior

Prior

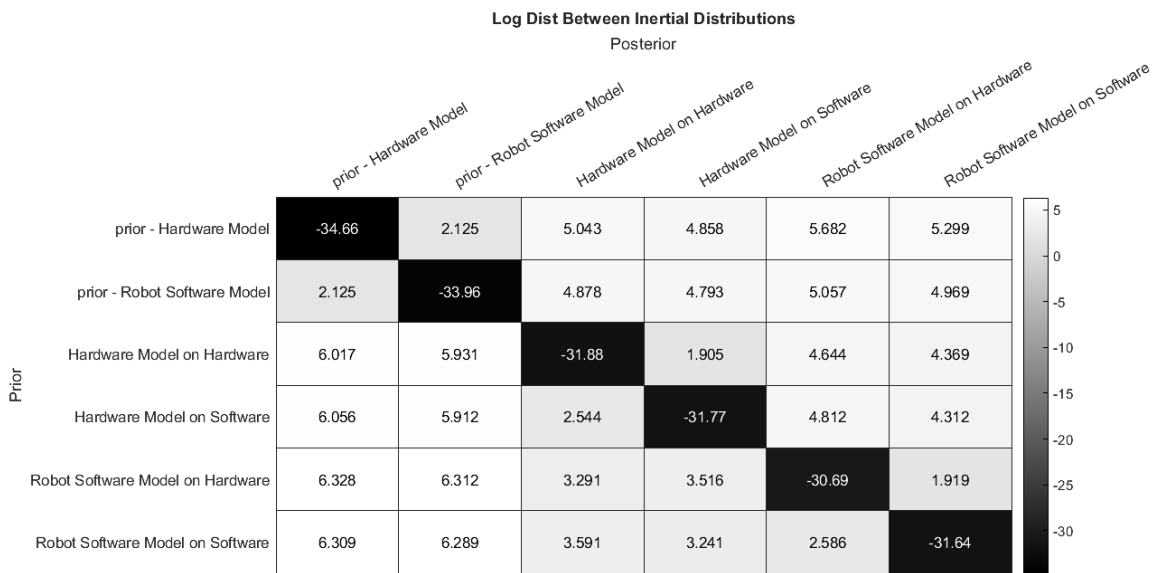|	prior - Hardware Model	prior - Robot Software Model	Hardware Model on Hardware	Hardware Model on Software	Robot Software Model on Hardware	Robot Software Model on Software
prior - Hardware Model	-34.66	2.125	5.043	4.858	5.682	5.299
prior - Robot Software Model	2.125	-33.96	4.878	4.793	5.057	4.969
Hardware Model on Hardware	6.017	5.931	-31.88	1.905	4.644	4.369
Hardware Model on Software	6.056	5.912	2.544	-31.77	4.812	4.312
Robot Software Model on Hardware	6.328	6.312	3.291	3.516	-30.69	1.919
Robot Software Model on Software	6.309	6.289	3.591	3.241	2.586	-31.64

Figure 4-6: A matrix representation of the calculated Bregman distances between the prior inertial parameters and those fit from optimization using the Robot Software and Hardware MIT Humanoid Models and simulated and hardware data sets. Bregman Distances are reported on a log scale.
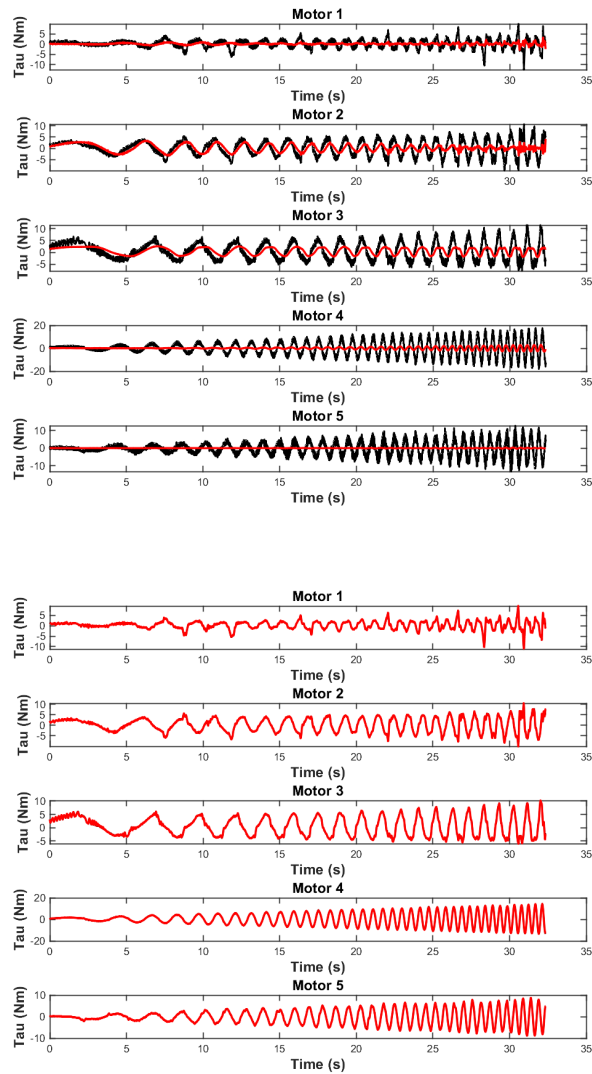
Figure 4-7: (Top) A comparison of measured physical motor torques (black) and those predicted after only fitting inertial parameters through convex optimization (red). The data shown is a representative example of the measured and predicted torques from using the MIT Humanoid Hardware Model on hardware data. (Bottom) A comparison of measured physical motor torques (black) and those predicted after fitting inertial parameters through convex optimization and training a neural net (red). The data shown is a representative example of the measured and predicted torques from using the MIT Humanoid Hardware Model on hardware data with a neural network for friction estimation. The predicted torques visibly nearly perfectly match the measured ones.
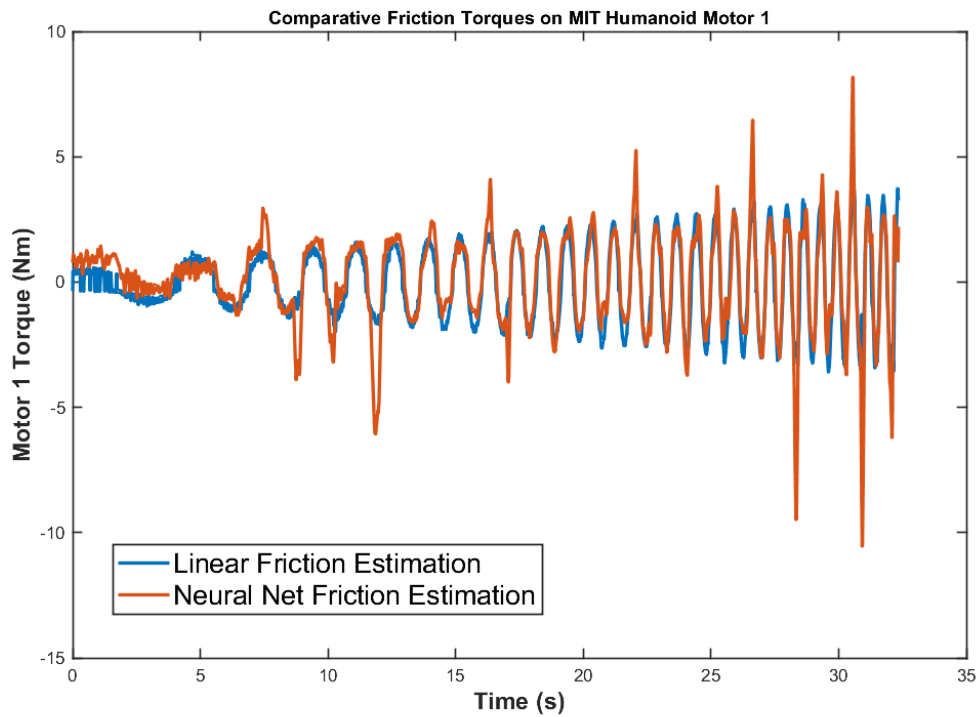
34

Figure 4-8: The estimated friction torques fit to MIT Humanoid hardware using the linear friction compensation approach employed by Lee et al. (blue) and using a neural net fitting the residual after only fitting inertial parameters using the convex optimization approach developed by Lee et al. (red). In both cases, the humanoid hardware model was used to describe the geometry constraints when optimizing inertial parameters.
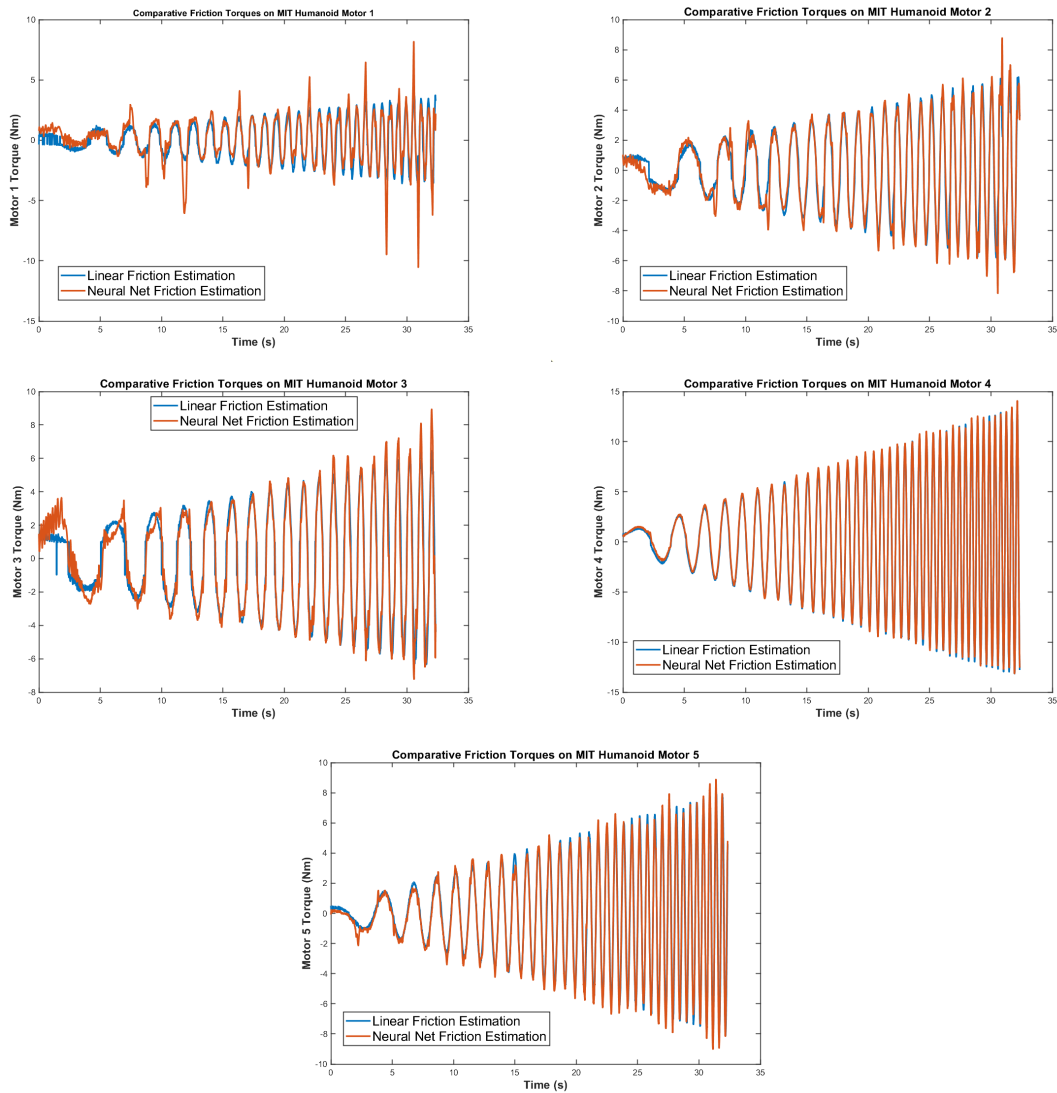
Figure 4-9: The estimated friction torques fit to MIT Humanoid hardware torque data using the linear friction compensation approach employed by Lee et al. (blue) and using a neural net fitting the residual after only fitting inertial parameters using the convex optimization approach developed by Lee et al. (red). In both cases, the humanoid hardware model was used to describe the geometry constraints when optimizing inertial parameters.

Chapter 5

Discussion

This body of work suggests that employing low-layer neural nets to estimate friction torques can result in reduced errors between the predicted torques output by a model and the true torques measured on a robot. This work was able to combine Lee et al.'s algorithm for finding the optimal parameters for a given model with an additional torque prediction measure that can compensate for any discrepancies between the torques predicted by a model and those measured. Tables 4.1, 4.3, and 4.2 showed that employing a neural net to predict unmodeled dynamics could greatly improve prediction errors from a variety of models fit to the same simulated and physical measurements. Part of these improvements run the risk of overfitting noise, but from figures like Figure 4-8, and the fact that the fit friction torques closely followed Lee et al.'s linear friction torques with additional characteristic features, this work suggests that simple neural networks can model effects that are difficult to model with linear methods and traditional system identification techniques alone. This work suggests then that using neural nets could serve as a useful tool for modeling and estimating non-linear torques in complicated systems with high DoF and non-linear dynamics. Using trained friction models fit to hardware data in simulation software could lead to more accurate robot modeling in simulation, further leading to control architectures that are better prepared for transitions to physical robots. Additionally, employing non-linear friction models in the state estimation of physical robots could lead to more accurate tracking in force feedback and impedance-based control schemes that depend on estimations of output forces for generating trajectory and force corrections.

Future work then would involve investigating if employing a neural net architecture in simulation and physical hardware when calculating joint torques could measurably improve robot performance in tracking tasks and tasks involving dynamic interactions. Furthermore, the current models of friction could be used to model non-linear dynamics in simulation, closing the sim-to-real gap and better-informing roboticists of their robot's performance outside of simulation. This would confer the benefits discussed in the introduction, leading to less time and cost in readjusting robot designs and control schemes when transitioning to physical systems and a better understanding of performance early on in the robot development cycle.

The author of this work particularly sees the benefit of employing the neural network friction system identification technique described in this paper on increasingly complex robotic systems. In the current study, the benefits of employing friction compensation with a neural net on the MIT Humanoid were clear and likely much more beneficial because the MIT Humanoid has larger inertias and coupled dynamics that are difficult to account for with traditional methods. Further studies could investigate the robustness of the current work's friction torque estimation procedure against increasingly more complicated systems, such as the entire MIT Humanoid robot, with arms and both legs included. By nature, a neural network is model blind and easily scaled to any type of system. As long as the system inertias can be formulated into equation 2.3, then the approach used in this work could potentially scale to systems with high degrees of freedom with relatively complex dynamics. Testing this work's friction compensation and system identification approach could reveal an easily translatable system for more accurate system identification in increasingly more complex robots that are hard to model by traditional means, giving roboticists and engineers the ability to more quickly design robots that can potentially handle increasingly more complex tasks.

Chapter 6

Conclusion

This work sought to investigate the effect of including non-linear friction compensation in the system identification of the MIT Cheetah and MIT Humanoid robotic systems in conjunction with the convex optimization algorithm developed by Lee et. al. The work found that employing non-linear friction compensation through a relatively small trained neural network was able to reduce the torque prediction errors of a variety of different robot models in both simulated and physically measured datasets. Models that more accurately reflected the robot used to collect data in a given dataset, such as the Hardware Model used on the MIT Humanoid hardware torque and positions data, generally tended to experience greater reductions in torque prediction errors compared to simpler models that did not include all of the dynamics present in the measured model. However, even in simplified models, friction compensation still significantly decreased robot torque prediction errors by including effects that are difficult to model with traditional linear system identification systems.

Overall, the conducted research of this work promoted the efficacy of using neural nets to include non-linear and unmodeled effects in the system identification of dynamic robotic systems. Such work adds to the body of knowledge seeking to improve robot system identification and may be able to improve the control and design of more complex robotic systems.

Bibliography

[1] Vincent Bonnet, Philippe Fraisse, André Crosnier, Maxime Gautier, Alejandro González, and Gentiane Venture. Optimal exciting dance for identifying inertial parameters of an anthropomorphic structure. 32(4):823–836. Conference Name: IEEE Transactions on Robotics.

[2] M. Gautier and W. Khalil. Exciting trajectories for the identification of base inertial parameters of robots. In *[1991] Proceedings of the 30th IEEE Conference on Decision and Control*, pages 494–499 vol.1.

[3] Yanjiang Huang, Jianhong Ke, Xianmin Zhang, and Jun Ota. Dynamic parameter identification of serial robots using a hybrid approach. 39(2):1607–1621. Conference Name: IEEE Transactions on Robotics.

[4] Taeyoon Lee, Patrick M. Wensing, and Frank C. Park. Geometric robot dynamic identification: A convex programming approach. 36(2):348–365. Conference Name: IEEE Transactions on Robotics.

[5] Lrinc Marton and Bla Lantos. Modeling, identification, and compensation of stick-slip friction. 54(1):511–521. Conference Name: IEEE Transactions on Industrial Electronics.

[6] J. Swevers, C. Ganseman, D.B. Tukel, J. de Schutter, and H. Van Brussel. Optimal robot excitation and identification. 13(5):730–740. Conference Name: IEEE Transactions on Robotics and Automation.