A Preliminary Study of Transfer Learning between Unicycle Robots

Kaizad V. Raimalwala, Bruce A. Francis and **Angela P. Schoellig**
University of Toronto
27 King's College Circle
Toronto, ON M5S 1A1, Canada

Abstract

Methods from machine learning have successfully been used to improve the performance of control systems in cases when accurate models of the system or the environment are not available. These methods require the use of data generated from physical trials. Transfer Learning (TL) allows for this data to come from a different, similar system. The goal of this work is to understand in which cases a simple, alignment-based transfer of data is beneficial. A scalar, linear, time-invariant (LTI) transformation is applied to the output from a source system to align with the output from a target system. In a theoretic study, we have already shown that for linear, single-input, single-output systems, the upper bound of the transformation error depends on the dynamic properties of the source and target system, and is small for systems with similar response times. We now consider two nonlinear, unicycle robots. Based on our previous work, we derive analytic error bounds for the linearized robot models. We then provide simulations of the nonlinear robot models and experiments with a Pioneer 3-AT robot that confirm the theoretical findings. As a result, key characteristics of alignment-based transfer learning observed in our theoretic study prove to be also true for real, nonlinear unicycle robots.

Introduction

If robots are to work effectively in multi-agent scenarios, their ability to share knowledge and learn from each others' experiences is important to develop. If multiple, similar robots have to perform the same task, it is more cost-effective if one robot learns to perform the task and transfers its knowledge to the other robots. In cooperative team-based scenarios, the exchange of a malfunctioning robot with a new one can be made faster if the old robot transfers its learned knowledge.

While there is different information that robots can share, such as maps (Dieter et al. 2006), object models and environments (Tenorth et al. 2012), or controllers (Chowdhary et al. 2013), we focus on the transfer of data for model learning for robot control.

Model learning is beneficial when models that accurately represent the robot's behavior are difficult to derive mathematically, or are unreliable due to parameter uncertainties or unknown external disturbances affecting the robot. The potential use of data from simulations or experiments to improve robot models is a prime motivator for modern research at the intersection of machine learning, control theory and robotics. Various regression techniques have been used to learn dynamics, kinematics and disturbance models using data from physical trials or simulations of a robot (Nguyen-Tuong and Peters 2011; Berkenkamp and Schoellig 2015; Ostafew, Schoellig, and Barfoot 2014; Schoellig, Mueller, and D'Andrea 2012).

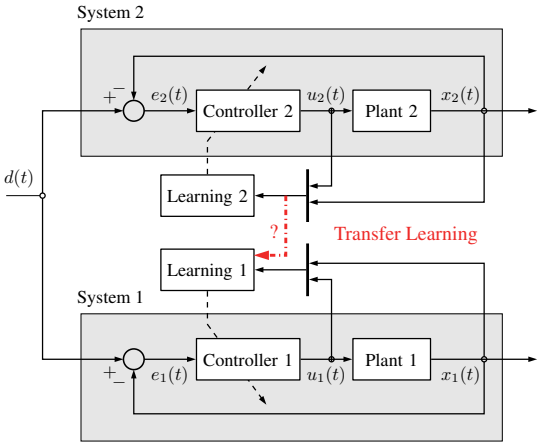


Figure 1: Transfer Learning framework. Systems 1 and 2 learn from input-output data (dashed lines). The concept of Transfer Learning (TL) allows for System 1 to use data from System 2 for its own learning task (red dash-dotted line). In this paper, we study TL in the context of robotics and provide insight to when TL is beneficial.

Transfer Learning (TL) allows for this data to be generated by a second system. In a training phase, both systems generate data, and a transformation that aligns one dataset to the other is learned. Once this mapping is learned, the first system, also called the target system, can use data generated by the second system, the source system, in subsequent model learning (see Fig. 1). This transfer of data may be beneficial to the model learning process if the source system is less costly, difficult, or hazardous to operate than the

target system.

In robotics, TL has often been considered in the context of speeding up a single robot's task learning using knowledge obtained in a previous task performed by the same robot (Konidaris, Scheidwasser, and Barto 2012). TL has, for example, been successfully used in an Iterative Learning Control (ILC) framework to speed up learning of new tasks (Arif, Ishihara, and Inooka 2001; Janssens, Pipeleers, and Swevers 2012; Hamer, Waibel, and D'Andrea 2013). Research for multi-agent robotic systems is relatively sparse (Tuyls and Weiss 2012). Most common applications aim to speed up joint or sequential learning either in an ILC framework by transferring task-dependent disturbance estimates (Schoellig, Alonso-Mora, and D'Andrea 2012), or in a Reinforcement Learning framework by transferring rules and policies of simple systems with discrete states (Taylor and Stone 2009; Lakshmanan and Balaraman 2010; Boutsioukis, Partalas, and Vlahavas 2012). However, TL can also be used in a model learning framework to apply a transformation on input-output data generated by one robot. This transformed data can then be input to a model learning algorithm for a second, similar robot (Bocsi, Csató, and Peters 2013). We are interested in the latter multi-agent learning scenario.

In several applications beyond robotics, Manifold Alignment has been used to find an optimal transformation to align datasets (Ham, Lee, and Saul 2005; Wang and Mahadevan 2009). In (Wang and Mahadevan 2008), this technique is demonstrated for two simple state-space models that each need to learn a task using Reinforcement Learning. In (Bocsi, Csató, and Peters 2013), a similar transformation technique is used on data from one robotic arm to speed up learning of a robot model for a second robotic arm.

While these works have shown that TL is feasible for some examples, they do not address the question of when the data transfer works well and when it fails. In previous work (Raimalwala, Francis, and Schoellig 2015), we introduced a framework to conduct an analytic study of how the quality of a linear transfer depends on the properties of the source and target systems that are both first-order, linear, time-invariant (LTI), and single-input, single-output (SISO).

In this paper, we consider nonlinear, unicycle robots. We apply our previous theoretic results to the linearized unicycle models to analyze when the transfer may work well and when it may not. This paper then makes a novel contribution by corroborating our theoretical findings with data from simulation of nonlinear unicycle models and from a real-world experiment with an indoor robot, a Pioneer 3-AT, which displays nonlinear behavior.

We begin by providing a background on model learning and TL for control systems, followed by the methodology used for our theoretical analysis. These sections summarize results from our previous paper (Raimalwala, Francis, and Schoellig 2015), while subsequent sections on unicycle robots present new results. We introduce the linearized unicycle model and apply our theoretic tools to these models. The theoretic results are then supported by results from nonlinear simulations and from an experiment with a Pioneer 3-AT robot.

Background

Model Learning

Dynamics and kinematics models govern a robot's behavior. While analytic models can be derived from first principles, they often do not capture the real-world dynamics accurately (Berkenkamp and Schoellig 2015; Ostafew, Schoellig, and Barfoot 2014). Supervised model learning presents a solution by employing a regression tool to find a map from input data to labelled observations. Given a sequence of input-output data with input $\mathbf{x}[i] \in \mathbb{R}^n$ and output $\mathbf{y}[i] \in \mathbb{R}^p$, where $i \in \{0, 1, 2, ..., N\}$ and N is the number of samples obtained, the problem of model learning is to find a map $\mathcal{M} : \mathbf{f}(\mathbf{x}) \rightarrow \mathbf{y}$ such that some measure of the magnitude of the error sequence,

$$\boldsymbol{\epsilon}[i] = \mathbf{y}[i] - \mathbf{f}(\mathbf{x}[i]), \quad i \in \{0, 1, 2, ..., N\}, \tag{1}$$

is minimized. For example, learning an inverse dynamics model for a robot arm can be formulated as finding a map $\mathcal{M} : \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \rightarrow \boldsymbol{\tau}$, where $\mathbf{q} \in \mathbb{R}^p$ is a vector of joint angles for all p joints of the arm, $\boldsymbol{\tau} \in \mathbb{R}^p$ is a vector of applied torques to each joint and $\mathbf{x} \in \mathbb{R}^{3p}$ (Nguyen-Tuong, Seeger, and Peters 2009).

Transfer Learning

When two robots (or control systems, in general), S_1 and S_2, execute a task, data is generated by each system. Data from S_2 can then undergo a transformation to align with data from S_1. The problem is akin to model learning in that a map needs to be found. In (Bocsi, Csató, and Peters 2013) and (Wang and Mahadevan 2008), the authors model this map as a time-invariant, linear transformation for each data sample. In this work, we make the same assumption.

Let vectors $\mathbf{x}_1[i]$ and $\mathbf{x}_2[i]$ be sampled data from S_1 and S_2. We thus define the problem of TL as finding a matrix $\mathbf{A}$ such that the vector 2-norm of

$$\boldsymbol{\epsilon}[i] = \mathbf{x}_1[i] - \mathbf{A}\mathbf{x}_2[i] \tag{2}$$

is minimized for all times $i \in \{0, 1, 2, ..., N\}$. The vector $\mathbf{x}$ can consist of system states, control inputs, or other variables that are relevant for a specific model learning algorithm. For the inverse dynamics model learning example in (Bocsi, Csató, and Peters 2013), the vector $\mathbf{x}$ is defined as $\mathbf{x} = [\mathbf{q}^T, \dot{\mathbf{q}}^T, \ddot{\mathbf{q}}^T, \boldsymbol{\tau}^T]^T$. Once such a matrix is learned from one pair of datasets, additional training data for learning the model of S_1 can be obtained by transforming subsequent data collected from S_2 using $\mathbf{A}$.

To find an optimal transformation $\mathbf{A}$ that aligns the discrete datasets, *a priori* models of each system need not be known, as the transformation only depends on data collected from physical trials of the two systems. The disadvantage of this data-alignment technique is that it is difficult to make predictions on the quality of the transformation. Furthermore, there usually is no guarantee on the performance of a given transformation on subsequent data.

Work in (Bocsi, Csató, and Peters 2013) shows that for two simulated robot arms, the data alignment worked well and sped up model learning. However, it is not obvious that the same approach works in other applications. Our work is

motivated by an interest to further explore the properties of a time-invariant, linear transformation for control system data, and determine when TL is most beneficial and when it fails.

We therefore consider two first-order, LTI, SISO systems, S_1 and S_2, in our analytic framework. We study TL in continuous time to facilitate our analysis. In this framework, $x_1(t), x_2(t) \in \mathbb{R}$ are the scalar system states, which are driven by the reference signal $d(t)$, and $\mathbf{A} = \alpha$ is a scalar that must map $x_2(t)$ to $x_1(t)$ (see Fig. 2). We find the transformation α^* that optimally aligns $x_2(t)$ to $x_1(t)$ for all $t \in [0, \infty)$ in an H_∞-norm sense.

This minimization provides the least upper bound on the 2-norm of the transformation error signal and we use this error bound to show how the quality of TL varies depending on the poles of S_1 and S_2, or in the case of our linearized unicycle models, depending on their controller gains.

Methodology

In this section, we introduce a framework for analyzing TL for simple, LTI, SISO systems: we define an H_∞-norm minimization problem where a scalar α is used to optimally align $x_2(t)$, the output of the source system, to $x_1(t)$, the output of the target system.

The transfer function of the target and source system are given by

$$G_1(s) = \frac{k_1}{s + a_1} \quad \text{and} \quad G_2(s) = \frac{k_2}{s + a_2}, \tag{3}$$

respectively, where $-a_1$ and $-a_2$ are the poles, and k_1 and k_2 are the gains of G_1 and G_2 (see Fig. 2). These transfer functions can represent path-following robots with closed-loop poles $-a_1$ and $-a_2$ and DC gains k_1/a_1 and k_2/a_2. For example, we later show that a unicycle's linearized kinematics model under proportional feedback control with gain k can be represented by the transfer function $k(s + k)^{-1}$.

The quantity of interest in the TL problem is the error of the alignment of $x_2(t)$ to $x_1(t)$ and is the output of the transfer system,

$$e_A(t) = x_1(t) - \alpha x_2(t), \tag{4}$$

where α is a time-invariant scalar. The transfer function from $d(t)$ to $e_A(t)$ is

$$G_A(s) = \frac{k_1}{s + a_1} - \alpha \frac{k_2}{s + a_2}. \tag{5}$$

To assure that $G_A(s)$ is asymptotically stable, a_1 and a_2 are assumed to be positive. Furthermore, k_2 is assumed to be non-zero to avoid the degenerate case where $G_A = G_1$.

Design Criterion. The signal 2-norm is chosen as a measure for the signal $e_A(t)$ and is denoted by $\| \cdot \|_2$. This measure can be determined for a specific reference signal $d(t) \in \mathcal{L}_2[0, \infty)$, where $\mathcal{L}_2[0, \infty)$ denotes the set of all signals that have finite energy on an infinite-time interval $[0, \infty)$. However, the H_∞-norm of G_A provides the least upper bound on $\|e_A\|_2$ for all $d(t) \in \mathcal{D}$, as shown in (Doyle, Francis, and Tannenbaum 2013); that is,

$$\|G_A\|_\infty = \sup\{\|e_A\|_2 : d(t) \in \mathcal{D}\}, \tag{6}$$

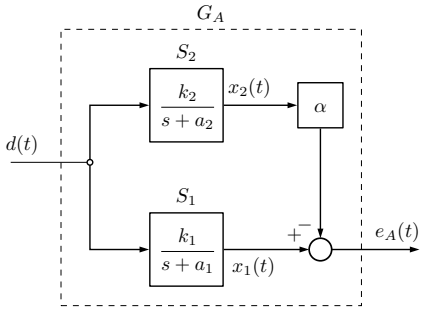


Figure 2: For data transfer, $x_2(t)$ is multiplied by a scalar α to match $x_1(t)$. While $x_1(t)$ and $x_2(t)$ are outputs of subsystems S_1 and S_2, the output of the overall system is $e_A(t)$.

where the H_∞-norm of G_A is given by

$$\|G_A\|_\infty = \sup_\omega |G_A(j\omega, \alpha)|, \quad \omega \in \mathbb{R}. \tag{7}$$

Problem Definition. The transfer problem is formulated as minimizing $\|G_A\|_\infty$ with respect to α:

$$\alpha^* := \arg\min_\alpha \|G_A\|_\infty. \tag{8}$$

The H_∞-norm is useful in analyzing the properties of TL for a large set of reference signals. Assuming that any signal in $\mathcal{D}$ is a potential reference signal, the optimal transformation α^* represents the best possible transformation that would be obtained when observing the system for an infinite amount of time under all possible reference inputs $d(t) \in \mathcal{D}$. Consequently, as long as the reference signal belongs to the set $\mathcal{D}$, the H_∞-norm evaluated at α^*, denoted by γ_A^*, provides the worst possible transformation error we could get.

In previous work (Raimalwala, Francis, and Schoellig 2015), we derived an analytic solution for α^* and found that the minimized lower bound takes the form

$$\gamma_A^* = \frac{k_1}{a_1} h(a_1, a_2), \tag{9}$$

where $h(a_1, a_2)$ is a rather complex function of a_1 and a_2. The ratio $k_1 a_1^{-1}$ is known as the DC gain of S_1 and indicates the steady-state output for a unit step input. This result can be used to study TL for linearized unicycle models with proportional feedback control.

Unicycle Model

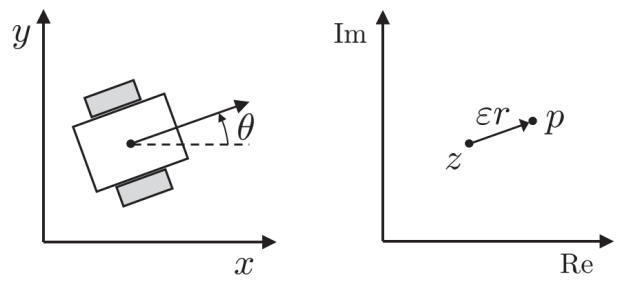
The 2D unicycle, depicted in Fig. 3a, has the pose $\mathbf{x} = [x, y, \theta]^T$ and its motion in an inertial frame can be modelled by the nonlinear kinematic equations,

$$\dot{x}(t) = v(t) \cos\theta(t), \tag{10}$$

$$\dot{y}(t) = v(t) \sin\theta(t), \tag{11}$$

$$\dot{\theta}(t) = \omega(t), \tag{12}$$

where $v(t)$ and $\omega(t)$ are the translational and rotational speed, respectively, and are considered to be inputs to the robot. To analyze TL for the unicycle in the H_∞ framework,



(a) (b)

Figure 3: On the left is a depiction of a unicycle robot in the 2D plane. On the right is the position of the robot, z, represented in the complex plane with the point p being ahead by distance ε in the direction of the robot's heading.

the model must first be linearized. To begin, we represent the unicycle pose in the complex plane,

$$z(t) = x(t) + jy(t), \tag{13}$$

$$r(t) = \mathrm{e}^{j\theta(t)}. \tag{14}$$

In this representation, the unicycle model is

$$\dot{z}(t) = r(t)v(t), \tag{15}$$

$$\dot{r}(t) = jr(t)\omega(t). \tag{16}$$

We then linearize the unicycle about a point $p(t) := z(t) + \varepsilon r(t)$ that is a distance $\varepsilon > 0$ ahead of the unicycle in the direction $r(t)$ (see Fig. 3b and (Yun and Yamamoto 1992)). We can decompose the point ahead into its real and complex parts: $p(t) = p_x(t) + jp_y(t)$ with $p_x(t), p_y(t) \in \mathbb{R}$. We define an artificial control input

$$u(t) = u_x(t) + ju_y(t), \quad u_x(t), u_y(t) \in \mathbb{R},$$

such that

$$\dot{p}(t) = u(t) \quad \Leftrightarrow \quad \dot{p}_x(t) = u_x(t), \quad \dot{p}_y(t) = u_y(t), \tag{17}$$

and the linearized dynamics are represented by a simple integrator in each direction.

The original inputs can be obtained from $u(t)$ using $\dot{p}(t) = r(t)v(t) + \varepsilon jr(t)\omega(t) = u(t)$ and separating the real and imaginary components (Yun and Yamamoto 1992):

$$v(t) = u_x(t)\cos\theta(t) + u_y(t)\sin\theta(t), \tag{18}$$

$$\omega(t) = \frac{1}{\varepsilon}\left(u_y(t)\cos\theta(t) - u_x(t)\sin\theta(t)\right). \tag{19}$$

While we design a linear feedback controller for the linearized unicycle robot based on (17), it is important to note that the nonlinear equations (18)–(19) will be used to control the nonlinear unicycle model in simulation and the Pioneer robot in our indoor experiment. In simulation and experiment, we additionally assume that $v(t)$ and $\omega(t)$ are constrained.

By controlling the position of a point ahead of the robot, the rotation (the nonlinearity of the system) is removed, and

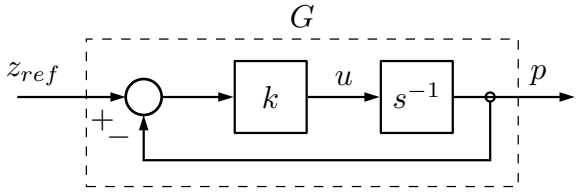


Figure 4: Control block diagram for the linearized unicycle with proportional feedback control.

the system behaves like a kinematic point. To track a reference signal $z_{ref}(t)$ in the complex plane, a simple proportional controller with gain k can be devised,

$$u(t) = k\left(z_{ref}(t) - p(t)\right). \tag{20}$$

The resulting closed-loop system is depicted in Fig. 4 and summarized by the transfer function

$$G = \frac{k}{s+k}, \tag{21}$$

which is exactly the form of the system used in the earlier TL analysis, where here the DC gain is equal to one. For two of these linearized unicycle models, each with gain k_1 and k_2, the resulting minimized bound on the 2-norm of the transformation error can be expressed as

$$\gamma_A^* = h(k_1, k_2). \tag{22}$$

Theoretic Results

We began by presenting a method to analyze how the error of data alignment varies with respect to the system parameters, and then applied this result to linearized unicycle models, which was shown to be a special case of the general result. We can now visualize γ_A^* from (22) in a contour plot with the unicycle controller gains k_1 and k_2 on the x- and y-axes (see Fig. 5). The base-10 logarithm of the data is shown to illustrate the variation more clearly. A white line is drawn through the diagonal as $\gamma_A^* = 0$ when $k_1 = k_2$. Of note are the following observations:

Key Observation 1. The error is lower if k_1 and k_2 lie close together, that is, if the two systems are dynamically similar.

Key Observation 2. If k_1 and k_2 are greater, they can be further apart for the minimized transfer error bound to stay on the same contour line. For example, consider the two black contour lines for $\gamma_A^* = 0.1$ in Fig. 5. If $k_1 = 10$, then k_2 must be approximately between 7.8 and 13.1 for γ_A^* to stay under 0.1. However, if $k_1 = 12$, then the range of allowable k_2 increases by around 19% to be approximately between 9.4 and 15.7.

Key Observation 3. In contrasting the top-left and bottom-right corners of Fig. 5, we see that it is slightly more preferable to have $k_2 > k_1$; that is, it is preferable for the source system to have a faster response than the target system.

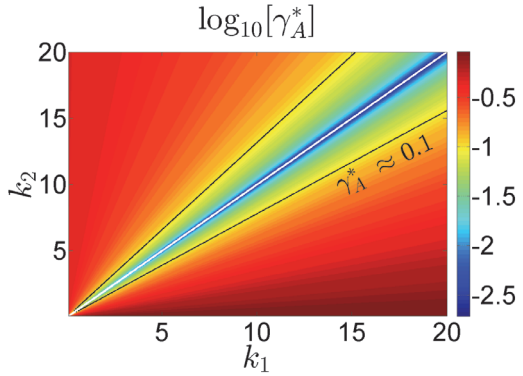


Figure 5: A contour plot of $\log_{10}[\gamma_A^*]$ vs. k_1 and k_2 is shown. The parameters k_1 and k_2 reflect the dynamics of unicycle 1 and 2. Low values of γ_A^* represent low transfer error.

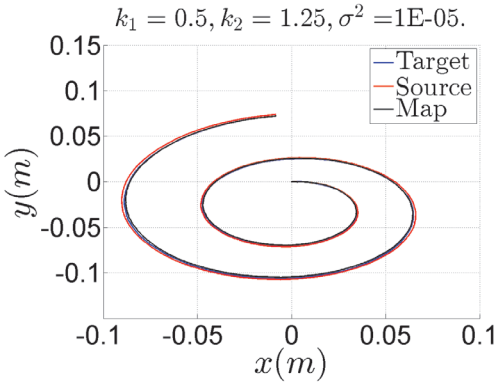


Figure 6: The two unicycle robots move in the outward spiral. The TL objective is to align the red trajectory to the blue through $\bar{\alpha}$ (black line).

Unicycle Simulation

In this section, we demonstrate TL for datasets generated by tasking two linearized unicycle robots to follow a specific reference signal in simulation. Their models and controllers are given by (17) and (20), respectively.

For the simulation, the model and controller are discretized, resulting in the discrete-time equations

$$p[i+1] = p[i] + \Delta t\, u[i] + n_{p,x}[i] + j n_{p,y}[i] \quad \text{with } n_{p,x}[i], n_{p,y}[i] \sim \mathcal{N}(0, \sigma^2), \tag{23}$$

$$u[i] = k(z_{ref}[i] - p[i]), \tag{24}$$

where i and Δt are the discrete-time index and time step, respectively. To match the experiment later, Δt is chosen to be 0.005. To the x- and y-dimensions, we separately add noise $n_{p,x}$ and $n_{p,y}$, which are Gaussian distributed with zero mean and variance σ^2. The robots are tasked to follow an outward spiral trajectory,

$$x_{ref}(t) = \eta_3 \sin(\eta_1 t) \mathrm{e}^{\eta_2 t}, \tag{25}$$

$$y_{ref}(t) = \eta_3 \big(\cos(\eta_1 t) \mathrm{e}^{\eta_2 t} - 1 \big) \tag{26}$$

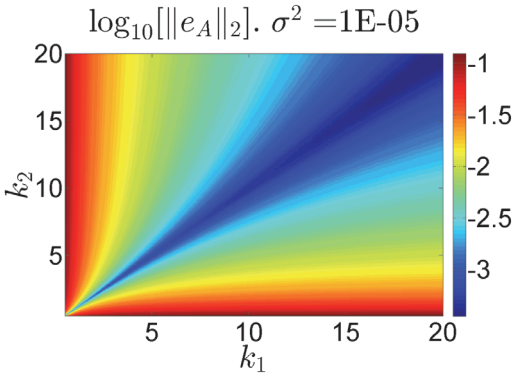


Figure 7: A contour plot of $\log_{10}[\|e_A\|_2]$ vs. k_1 and k_2 for $\sigma^2 = 1 \times 10^{-5}$.

with parameters $\eta_1 = 0.1$, $\eta_2 = 0.01$, and $\eta_3 = 0.03$ (see Fig. 6). The simulation is run for $4\pi/\eta_1 \approx 126$ seconds. Data is collected for $p_1[i]$, $p_2[i]$, and $z_{ref}[i]$, where $i \in \{1, 2, \ldots, N\}$. Let $\mathbf{P}_1$ denote an $(N \times 1)$ matrix of samples of $p_1[i]$,

$$\mathbf{P}_1 = [p_1[1] \quad \cdots \quad p_1[i] \quad \cdots \quad p_1[N]]^T. \tag{27}$$

Similarly, $\mathbf{P}_2$ is constructed from samples of $p_2[i]$. Let $\mathbf{M}_1 = [\mathrm{Re}(\mathbf{P}_1^T), \mathrm{Im}(\mathbf{P}_1^T)]^T$ and likewise for $\mathbf{M}_2$. Then, by ordinary least-squares regression, an optimal $\bar{\alpha}$ is found such that

$$\bar{\alpha} = \arg\min_{\alpha} \mathbf{E}_A^T \mathbf{E}_A, \tag{28}$$

where $\mathbf{E}_A := \mathbf{M}_1 - \alpha \mathbf{M}_2$. The solution is given by

$$\bar{\alpha} = \mathbf{M}_2^+ \mathbf{M}_1, \tag{29}$$

where $\mathbf{M}_2^+ = [\mathbf{M}_2^T \mathbf{M}_2]^{-1} \mathbf{M}_2^T$. The mapped trajectory in the complex plane is then

$$\hat{\mathbf{P}} = \bar{\alpha} \mathbf{P}_2. \tag{30}$$

An estimate of the 2-norm of the error signal $e_A[i] = p_1[i] - \bar{\alpha} p_2[i]$ is then computed by trapezoidal integration,

$$\|e_A\|_2 = \sqrt{\frac{\Delta t}{2} \sum_{i=2}^{N} \left(|e_A[i]|^2 + |e_A[i-1]|^2 \right)}. \tag{31}$$

Fig. 6 shows the trajectories of the target and source systems and the alignment for $k_1 = 0.5$ and $k_2 = 1.25$, with noise variance $\sigma^2 = 1 \times 10^{-5}$. The error 2-norm is 0.079, slightly lower than the 2-norm of the difference between the source and target trajectories at 0.081.

We now see how the 2-norm of the transformation error varies for simulations that use different combinations of k_1 and k_2 in Fig 7. We see that $\|e_A\|_2$ varies in a way that is similar to γ_A^*. The three key observations made in the previous section from the analysis for a large set of reference signals also hold for this specific outward-spiral reference signal. However, we note that $\|e_A\|_2$ is only slightly asymmetrical with respect to a_1 and a_2, in contrast to the more asymmetric variation of γ_A^*.

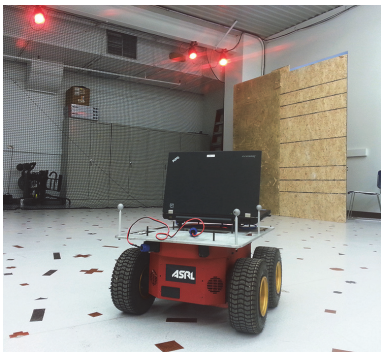


Figure 8: The Pioneer 3-AT robot in the motion-capture-enabled lab at the University of Toronto Institute for Aerospace Studies.

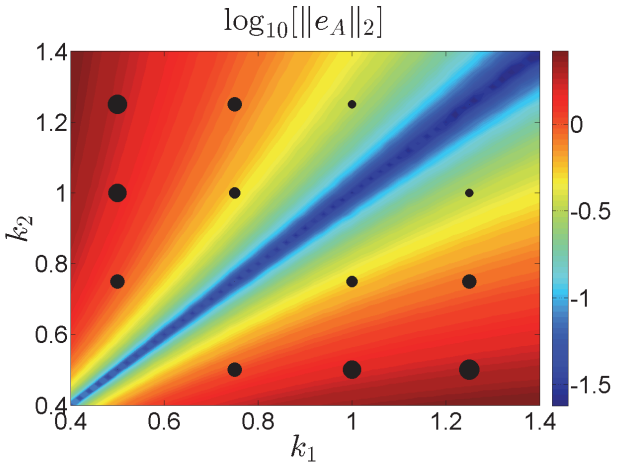


Figure 9: In the background is a contour plot of $\|e_A\|_2$ vs. k_1 and k_2 for the simulation of the nonlinear unicycle system moving on a spiral path. In the foreground are scaled black circles that denote $\|e_A\|_2$ obtained from experiments.

Experimental Results

In this section, we obtain results similar to those in the previous section, but for data gathered in physical experiments with the Pioneer 3-AT, a small four-wheel drive, skid-steer robot (see Fig. 8).

The Pioneer robot is roughly $27\,\mathrm{cm}$ long (distance between front and rear wheel centres) and $50\,\mathrm{cm}$ wide. It can drive at a maximum speed of $0.7\,\mathrm{m\,s^{-1}}$ and turn at a maximum rate of $2.44\,\mathrm{rad\,s^{-1}}$. However, we constrain these in our experiment to $v_{lim} = 0.5\,\mathrm{m\,s^{-1}}$ and $\omega_{lim} = 1.5\,\mathrm{rad\,s^{-1}}$. Considering these parameters, we choose $\eta_1 = 0.1$, $\eta_2 = .01$, $\eta_3 = .7$, and $\varepsilon = 0.1$ for an appropriate path to track.

For data acquisition, we use a motion capture system that uses active, infra-red cameras to provide the robot pose in an inertial frame at $200\,\mathrm{Hz}$. The Pioneer is tasked to follow the path a few times, each time with a different controller gain.

We perform experiments for various combinations of k_1 and k_2. The error values are presented as black circles in Fig. 9. The values for $\|e_A\|_2$ are used to scale the size of the circles. In the background is a contour plot of $\|e_A\|_2$ computed from simulating the full nonlinear unicycle systems with the same controllers and speed constraints used in the experiment.

We see that the three key observations made from the H_∞ analysis hold for the simulation of the nonlinear systems as well as for the experiments with the Pioneer robot. Indeed, the third observation, that it is slightly preferable to have $k_2 > k_1$, can be demonstrated by noting that for $\{k_1 = 1.25, k_2 = 0.5\}$, $||e_A||_2 = 1.56$, while for $\{k_1 = 0.5, k_2 = 1.25\}$, $||e_A||_2 = 1.52$.

Conclusions

In this paper, we first conducted an analytic study on the quality of alignment-based transfer learning (TL) for two linearized unicycle robot models with proportional feedback control. This is a specific application of our more general study in previous work, where we considered the data transfer between two linear, time-invariant (LTI), single-input, single-output (SISO) systems.

For the linearized unicycle model, we showed that the transfer quality depends on the controller gains of both robots. In particular, the transfer quality increases if the two systems have similar control gains and if they have larger gains.

The two linearized unicycle models, each with a different controller gain, were tasked to follow a spiral trajectory in simulation. An estimate of the transfer quality was computed for several combinations of the two system gains. It was shown that the variation of transfer quality was consistent with the analytic results before. This analysis was repeated for simulations with the nonlinear unicycle models and controllers, and velocity constraints. Lastly, we showed that these results also hold for data obtained from indoor experiments with a Pioneer robot. As a result, we proved that our analytic results based on linear models are consistent with nonlinear simulations and real experiments.

References

Arif, M.; Ishihara, T.; and Inooka, H. 2001. Incorporation of experience in iterative learning controllers using locally weighted learning. *Automatica* 37(6):881–888.

Berkenkamp, F., and Schoellig, A. P. 2015. Safe and robust learning control with Gaussian processes. In *Proc. of the European Control Conference (ECC)*, 2501–2506.

Bocsi, B.; Csató, L.; and Peters, J. 2013. Alignment-based transfer learning for robot models. In *Proc. of the 2013 International Joint Conference on Neural Networks (IJCNN)*, 1–7.

Boutsioukis, G.; Partalas, I.; and Vlahavas, I. 2012. Transfer learning in multi-agent reinforcement learning domains. In *Recent Advances in Reinforcement Learning*. Springer. 249–260.

Chowdhary, G.; Wu, T.; Cutler, M.; and How, J. P. 2013. Rapid transfer of controllers between UAVs using learning-based adaptive control. In *Proc. of the IEEE International*

Conference on Robotics and Automation (ICRA), 5409–5416.

Dieter, F.; Ko, J.; Konolige, K.; Limketkai, B.; Schulz, D.; and Stewart, B. 2006. Distributed multi-robot exploration and mapping. *Proceedings of the IEEE* 94:1325–1339.

Doyle, J. C.; Francis, B. A.; and Tannenbaum, A. 2013. *Feedback control theory*. Courier Corporation.

Ham, J.; Lee, D.; and Saul, L. 2005. Semisupervised alignment of manifolds. In *Proc. of the Annual Conference on Uncertainty in Artificial Intelligence*, volume 10, 120–127.

Hamer, M.; Waibel, M.; and D'Andrea, R. 2013. Knowledge transfer for high-performance quadrocopter maneuvers. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1714–1719.

Janssens, P.; Pipeleers, G.; and Swevers, J. 2012. Initialization of ILC based on a previously learned trajectory. In *Proc. of the American Control Conference (ACC)*, 610–614.

Konidaris, G.; Scheidwasser, I.; and Barto, A. G. 2012. Transfer in reinforcement learning via shared features. *The Journal of Machine Learning Research* 13:1333–1371.

Lakshmanan, B., and Balaraman, R. 2010. Transfer learning across heterogeneous robots with action sequence mapping. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3251–3256.

Nguyen-Tuong, D., and Peters, J. 2011. Model learning for robot control: a survey. *Cognitive processing* 12(4):319–340.

Nguyen-Tuong, D.; Seeger, M.; and Peters, J. 2009. Model learning with local Gaussian process regression. *Advanced Robotics* 23(15):2015–2034.

Ostafew, C. J.; Schoellig, A. P.; and Barfoot, T. D. 2014. Learning-based nonlinear model predictive control to improve vision-based mobile robot path-tracking in challenging outdoor environments. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, 4029–4036.

Raimalwala, K. V.; Francis, B. A.; and Schoellig, A. P. 2015. An upper bound on the error of alignment-based transfer learning between two linear, time-invariant, scalar systems. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Schoellig, A. P.; Alonso-Mora, J.; and D'Andrea, R. 2012. Limited benefit of joint estimation in multi-agent iterative learning. *Asian Journal of Control* 14(3):613–623.

Schoellig, A. P.; Mueller, F. L.; and D'Andrea, R. 2012. Optimization-based iterative learning for precise quadrocopter trajectory tracking. *Autonomous Robots* 33(1-2):103–127.

Taylor, M. E., and Stone, P. 2009. Transfer learning for reinforcement learning domains: A survey. *The Journal of Machine Learning Research* 10:1633–1685.

Tenorth, M.; Perzylo, A. C.; Lafrenz, R.; and Beetz, M. 2012. The RoboEarth language: Representing and exchanging knowledge about actions, objects, and environments. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, 1284–1289.

Tuyls, K., and Weiss, G. 2012. Multiagent learning: Basics, challenges, and prospects. *AI Magazine* 33(3):41.

Wang, C., and Mahadevan, S. 2008. Manifold alignment using procrustes analysis. In *Proc. of the 25th International Conference on Machine Learning*, 1120–1127.

Wang, C., and Mahadevan, S. 2009. A general framework for manifold alignment. In *AAAI Fall Symposium on Manifold Learning and Its Applications*, 53–58.

Yun, X., and Yamamoto, Y. 1992. On feedback linearization of mobile robots. Technical Report. Department of Computer Science and Information Science, University of Pennsylvania.