2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Data-Driven Dynamic Input Transfer for Learning Control in Multi-Agent Systems with Heterogeneous Unknown Dynamics

Dustin Lehmann, Philipp Drebinger, Thomas Seel and Jörg Raisch

Abstract— **Learning input signals that make a dynamic system respond with a desired output is often data intensive and time consuming. It is therefore natural to ask whether, in a heterogeneous multi-agent scenario, an input signal learned by one agent can be suitably adapted and transferred to make the other agents respond with the same desired output, despite exhibiting different dynamics. In this paper, we propose a novel method to achieve this by employing a dynamic input transfer map. The method does not require any a-priori knowledge of the individual agents' dynamics. Instead, a small amount of experimental data from the source and target systems are used to estimate the transfer map. We evaluate the proposed method and compare it to existing approaches using static input transfer maps by investigating two example scenarios: (i) a simulation scenario for muscle dynamics, (ii) an experimental setting with a group of two-wheeled inverted pendulum robots and a sim-to-real motion learning problem.**

I. INTRODUCTION

In many systems and control applications, a common task is to find a feedforward input signal that makes the output of a dynamical system track a desired reference signal. Methods that solve these problems, such as iterative learning control (ILC), e.g. [1] or reinforcement learning, e.g. [2], usually achieve this by improving a feed-forward input signal over repeated trials using a learning policy based on the corresponding output signal. In multi-agent systems (MAS), the exchange of these signals between individual agents is a crucial element to transfer the experience gained on a *source* system to one or multiple *target* systems. This transfer is a key component in methods such as the one proposed in [3], in which a recurring transfer is used by a MAS to find a collective solution for a motion task or in the methods proposed in [4]–[7], in which it is used to transfer already learned output signals from a source to a target system. The latter is used if one or multiple agents aim to perform the same task to which a solution is already available for a source agent [5] or to use the source agent's experience to accelerate the learning on the target systems [7]. This is especially beneficial if learning on the source system is easier, faster or safer [8]. A typical example of such a case is the so-called *sim-to-real transfer*, in which a certain task is learned by a digital clone of the target system in simulation and the learned input signal is then transferred to the target to perform the task [9], [10] (see Fig. 1).

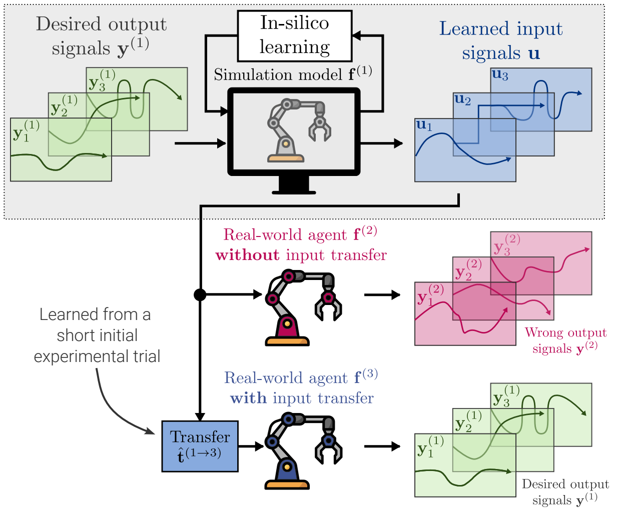
Desired output signals $\mathbf{y}^{(1)}$
In-silico learning
Simulation model $\mathbf{f}^{(1)}$
Learned input signals $\mathbf{u}$
$\mathbf{y}_3^{(1)}$
$\mathbf{y}_2^{(1)}$
$\mathbf{y}_1^{(1)}$
$\mathbf{u}_3$
$\mathbf{u}_2$
$\mathbf{u}_1$
Real-world agent $\mathbf{f}^{(2)}$
without input transfer
$\mathbf{y}_3^{(2)}$
$\mathbf{y}_2^{(2)}$
$\mathbf{y}_1^{(2)}$
Wrong output signals $\mathbf{y}^{(2)}$
Learned from a short initial experimental trial
Real-world agent $\mathbf{f}^{(3)}$
with input transfer
Transfer $\hat{\mathbf{t}}^{(1\to3)}$
$\mathbf{y}_3^{(1)}$
$\mathbf{y}_2^{(1)}$
$\mathbf{y}_1^{(1)}$
Desired output signals $\mathbf{y}^{(1)}$

Fig. 1: Sim-to-real scenario with and without the use of an input transfer map. A simulation model $\mathbf{f}^{(1)}$ is used to learn input signals $\mathbf{u}$ for a set of given, desired output signals $\mathbf{y}^{(1)}$. If the learned input signals $\mathbf{u}$ are applied to real-world agent 2 with different dynamics $\mathbf{f}^{(2)}$, the resulting outputs $\mathbf{y}^{(2)}$ differ from the desired $\mathbf{y}^{(1)}$. However, using an input transfer map $\mathbf{t}^{(1\to3)}$ for agent 3 with dynamics $\mathbf{f}^{(3)}$, the desired outputs $\mathbf{y}^{(1)}$ from the source system can be reproduced on agent 3.

If the agents' dynamics are similar, a *direct transfer* without transformation of signals between agents can be used, e.g. [11]. In [12], [13] the authors propose methods to determine if the agents' dynamics are sufficiently similar for direct transfer to be useful, based on known system dynamics. However, if the source and target agents' dynamics are dissimilar, a direct transfer of input or output signals might not be feasible and could lead to unwanted behaviour on the target system [13]. In [14], [15] the authors propose a method for feedback-controlled systems which circumvents the problem of dissimilar dynamics for signal transfer. They propose the use of an adaptive controller to force similar dynamics on source and target agents. This allows the direct transfer of signals, once the adaptive controllers have converged. However, as demonstrated in [3], multi-agent systems can greatly benefit from heterogeneous dynamics, e.g. for cooperative learning.

To use individual dynamics in dissimilar multi-agent systems, one can use a *transfer map*, which transforms the signals before transferring them between agents with dissimilar dynamics. In [4], [5], [16] the authors propose the use of a *scalar map* to enable the transfer of signals between dynamic systems. However, this is only shown to work for linear systems with dynamics that only differ in

D. Lehmann, P. Drebinger and J. Raisch are with the Control Systems Group, Technische Universität Berlin, Germany and Science of Intelligence, Research Cluster of Excellence, Technische Universität Berlin, Germany. dustin.lehmann@tu-berlin.de and raisch@control.tu-berlin.de

T. Seel is with the Intelligent Sensorimotor Systems Lab, Department AIBE, FAU Erlangen-Nürnberg, Germany thomas.seel@fau.de

979-8-3503-0123-6/23/$31.00 ©2023 IEEE

gain or for low-frequency signals. For dissimilar systems and high-frequency signals, a *dynamic map*, is needed to achieve similar behaviour on source and target systems. In [17] and [7] the authors propose different, promising methods for using a dynamic map for signal transfer. These methods are shown to work for dissimilar systems, however, they require precise a-priori model knowledge of the linear system dynamics of all agents, which can come either from precise modelling or extensive system identification procedures with large amounts of data, both of which can be infeasible or even impossible in certain applications.

In the present manuscript, we present a consistent notational framework for signal transfer in MAS and derive a connection between input transfer and output transfer, two different strategies found in the existing literature. Based on that relationship we propose a novel method for input transfer between agents with dissimilar dynamics using a dynamic map. The method does not require any a-priori individual model knowledge, it instead requires a small amount of experimental data from the source and target system. We evaluate the method in simulation for a commonly used simulation model of muscle dynamics and compare it to selected literature methods. Furthermore, we evaluate the method in an experimental setting with a group of two-wheeled inverted pendulum robots (TWIPR), which feature non-linear, high-order, non-minimum phase dynamics and a complex sim-to-real motion learning problem.

II. Notation

We employ the following notation: $\mathbb{R}$ denotes the set of real numbers, $\mathbb{N}$ the set of positive integers, $\mathbb{N}_0$ the set of non-negative integers and $\mathbb{Z}$ the set of integers. Scalar values and scalar functions are denoted by lower-case regular letters, e.g. $a \in \mathbb{R}$ or $f : \mathbb{R} \to \mathbb{R}$. Vector values and vector functions are denoted by lower-case bold letters, e.g. $\mathbf{x} \in \mathbb{R}^3$ or $\mathbf{f} : \mathbb{R}^3 \to \mathbb{R}^2$. Matrices are denoted by upper-case bold letters, e.g. $\mathbf{P} \in \mathbb{R}^{3\times 3}$.

III. METHOD

A. Dynamic System Description

Consider a group of $A \in \mathbb{N}$ agents $i \in \{1, \dots, A\}$. We assume that the i-th agent's behavior can be described by a single-input single-output (SISO) state model

$$\mathbf{x}_{k+1}^{(i)} = \mathbf{g}^{(i)}(\mathbf{x}_k^{(i)}, u_k^{(i)}) \tag{1}$$

$$y_k^{(i)} = h^{(i)}(\mathbf{x}_k^{(i)}, u_k^{(i)}) \tag{2}$$

with $k \in [0, \dots, N-1]$ being the sample time index and $N \in \mathbb{N}$ the number of samples. The functions $\mathbf{g}^{(i)} : \mathbb{R}^{n_i} \times \mathbb{R} \to \mathbb{R}^{n_i}$ and $h^{(i)} : \mathbb{R}^{n_i} \times \mathbb{R} \to \mathbb{R}$ with $n_i \in \mathbb{N}$ are time-invariant and describe the state and output dynamics of agent i, respectively.

Assumption 1 (Stability): We assume the dynamics given in (1) *to be input-to-state stable [18].*

Assumption 2 (Initial Conditions): We assume the initial conditions of all agents to be identical, i.e. $\mathbf{x}_0^{(i)} = \mathbf{x}_0 = \mathbf{0}, i \in \{1, \dots, A\}$. *Furthermore, we assume that* $(\mathbf{x}_0, 0)$ *is a stationary point with zero output for all agents, i.e.* $y_k^{(i)} = 0 \;\forall i, k$ *if* $u_k^{(i)} = 0 \;\forall i, k$.

We denote the relative degree of the i-th agent's dynamics by $m^{(i)} \in \mathbb{N}_0$, i.e., the effect of the input signal on the output signal is delayed by $m^{(i)}$ samples. Formally, $m^{(i)}$ is the largest integer such that $\Big(\forall\; u^{(i)} = [u_0^{(i)}, \dots, u_{N-1}^{(i)}]^T, u'^{(i)} = [u_0'^{(i)}, \dots, u_{N-1}'^{(i)}]^T\Big)$ $y_k^{(i)} = y_k'^{(i)}$ for $k = [0, \dots, m^{(i)}-1]$. We can then compactly characterize the i-th agent's input/output dynamics by its relative degree $m^{(i)}$ and a map $\mathbf{f}^{(i)} : \mathbb{R}^N \to \mathbb{R}^N$ with

$$\underbrace{\begin{bmatrix} y_{m^{(i)}}^{(i)} \\ \vdots \\ y_{m^{(i)}+N-1}^{(i)} \end{bmatrix}}_{=:\mathbf{y}^{(i)}} = \mathbf{f}^{(i)}\left(\underbrace{\begin{bmatrix} u_0^{(i)} \\ \vdots \\ u_{N-1}^{(i)} \end{bmatrix}}_{=:\mathbf{u}^{(i)}}\right), i \in \{1, \dots, A\}. \tag{3}$$

with $\mathbf{u}^{(i)} \in \mathbb{R}^N$ the *input signal* and $\mathbf{y}^{(i)} \in \mathbb{R}^N$ the corresponding *output signal*.

Remark 1 (Linear Dynamics): Note that $f_l^{(i)}(\mathbf{u}^{(i)}), 1 \leq l \leq N$, *the* l*-th row of operator* $\mathbf{f}^{(i)}$, *depends only on the first* l *inputs. If* $\mathbf{g}^{(i)}$ *and* $h^{(i)}$ *are linear functions, we can rewrite* (3) *as*

$$\mathbf{y}^{(i)} = \mathbf{P}^{(i)} \cdot \mathbf{u}^{(i)} \tag{4}$$

with $\mathbf{P}^{(i)} \in \mathbb{R}^{N \times N}$ *being a lower-triangular Toeplitz (LTT) matrix.*

Assumption 3 (Heterogeneity): We assume the agents' dynamics to be heterogeneous, i.e. in general for any non-trivial $\mathbf{u} \in \mathbb{R}^N$ *and any pair of agents* (i, j)

$$\mathbf{f}^{(i)}(\mathbf{u}) \neq \mathbf{f}^{(j)}(\mathbf{u}) \quad \forall i, j \in \{1, \dots, A\}, i \neq j\,. \tag{5}$$

B. Input & Output Transfer Map

Consider two agents $i, j \in \{1, \dots, A\}, i \neq j$ described by their input/output maps $\mathbf{f}^{(i)}$ and $\mathbf{f}^{(j)}$ and their relative degrees $m^{(i)}$ and $m^{(j)}$. Let $\mathbf{u} \in \mathbb{R}^N$ be an input signal applied to both agents, i.e. $\mathbf{u}^{(i)} = \mathbf{u}^{(j)} = \mathbf{u}$. Let $\mathbf{y}^{(i)} \in \mathbb{R}^N$ and $\mathbf{y}^{(j)} \in \mathbb{R}^N$ be the corresponding output signals according to (3). Given Ass. 3, the two outputs are different if $\mathbf{u}$ is non-trivial. We can describe the relation between $\mathbf{y}^{(i)}$ and $\mathbf{y}^{(j)}$ for any input $\mathbf{u}$ using the *output transfer map*.

Definition 1 (Output Transfer Map): Given two input/output maps $\mathbf{f}^{(i)}$, $\mathbf{f}^{(j)}$ *with corresponding relative degrees* $m^{(i)}$, $m^{(i)} \in \mathbb{N}_0$. *If a solution* $\mathbf{f}^{(i\to j)} : \mathbb{R}^N \to \mathbb{R}^N$ *for the operator equation*

$$\mathbf{f}^{(j)} = \mathbf{f}^{(i\to j)} \circ \mathbf{f}^{(i)} \tag{6}$$

exists, we call $\mathbf{f}^{(i\to j)}$ *an output transfer map and* $m^{(i\to j)} = m^{(j)} - m^{(i)} \in \mathbb{Z}$ *the corresponding degree information.*

Remark 2 (Interpretation of the output transfer map): Given $\mathbf{f}^{(i\to j)}$, $m^{(i\to j)}$ *and* $m^{(i)}$ *we can construct the* *relevant output $\mathbf{y}^{(j)}$ of agent j from the output $\mathbf{y}^{(i)}$ of agent i for the same input $\mathbf{u}^{(i)} = \mathbf{u}^{(j)} = \mathbf{u}$ with*

$$\underbrace{\begin{bmatrix} y^{(j)}_{m^{(j)}} \\ \vdots \\ y^{(j)}_{m^{(j)}+N-1} \end{bmatrix}}_{\mathbf{y}^{(j)}} = \mathbf{f}^{(i\to j)}\left(\underbrace{\mathbf{f}^{(i)}(\mathbf{u})}_{\mathbf{y}^{(i)}}\right) \tag{7}$$

where $m^{(j)} = m^{(i)} + m^{(i\to j)}$.

Remark 3 (Model Inversion): Given (6)*, if $\mathbf{f}^{(i)}$ has an inverse $\left(\mathbf{f}^{(i)}\right)^{-1}$ then $\mathbf{f}^{(i\to j)}$ uniquely exists and can be written as*

$$\mathbf{f}^{(i\to j)} = \mathbf{f}^{(j)} \circ \left(\mathbf{f}^{(i)}\right)^{-1}. \tag{8}$$

We will now introduce a dual concept, namely *input transfer maps*.

Definition 2 (Input Transfer Map): Given two input/output maps $\mathbf{f}^{(i)}$, $\mathbf{f}^{(j)}$. If a solution $\mathbf{t}^{(i\to j)} : \mathbb{R}^N \to \mathbb{R}^N$ for the operator equation

$$\mathbf{f}^{(i)} = \mathbf{f}^{(j)} \circ \mathbf{t}^{(i\to j)} \tag{9}$$

exists, then $\mathbf{t}^{(i\to j)}$ is called an input transfer map.

Remark 4 (Interpretation of the input transfer map): Given an input $\mathbf{u}^{(i)} \in \mathbb{R}^N$ for agent i and $\mathbf{t}^{(i\to j)}$ we can readily construct an input $\mathbf{u}^{(j)} \in \mathbb{R}^N$ for agent j, such that both agents generate the same relevant output $\mathbf{y} \in \mathbb{R}^N$ with

$$\underbrace{\begin{bmatrix} y^{(j)}_{m^{(j)}} \\ \vdots \\ y^{(j)}_{m^{(j)}+N-1} \end{bmatrix}}_{\mathbf{y}^{(j)}} = \mathbf{f}^{(j)}(\underbrace{\mathbf{t}^{(i\to j)}(\mathbf{u}^{(i)})}_{\mathbf{u}^{(j)}}) = \mathbf{f}^{(i)}(\mathbf{u}^{(i)}) = \underbrace{\begin{bmatrix} y^{(i)}_{m^{(i)}} \\ \vdots \\ y^{(i)}_{m^{(i)}+N-1} \end{bmatrix}}_{\mathbf{y}^{(i)}} \tag{10}$$

Remark 5 (Model Inversion): Given (9)*, if $\mathbf{f}^{(j)}$ has an inverse $\left(\mathbf{f}^{(j)}\right)^{-1}$ then $\mathbf{t}^{(i\to j)}$ uniquely exists and can be written as*

$$\mathbf{t}^{(i\to j)} = \left(\mathbf{f}^{(j)}\right)^{-1} \circ \mathbf{f}^{(i)}. \tag{11}$$

See Fig. 2 for a graphical comparison of input and output transfer maps. From (6) and (9), we can see that the relationship between the input transfer map and output transfer map can be written as

$$\mathbf{f}^{(j\to i)} \circ \mathbf{f}^{(j)} = \mathbf{f}^{(j)} \circ \mathbf{t}^{(i\to j)}. \tag{12}$$

We know that if $\mathbf{f}^{(j)}$ is invertible, $\mathbf{f}^{(j\to i)}$ and $\mathbf{t}^{(i\to j)}$ uniquely exist and can be written as

$$\mathbf{f}^{(j\to i)} = \mathbf{f}^{(i)} \circ \left(\mathbf{f}^{(j)}\right)^{-1} \tag{13}$$

$$\mathbf{t}^{(i\to j)} = \left(\mathbf{f}^{(j)}\right)^{-1} \circ \mathbf{f}^{(i)}. \tag{14}$$

If, additionally, $\mathbf{f}^{(i)}$ and $\mathbf{f}^{(j)}$ commute, i.e., if

$$\mathbf{f}^{(i)} \circ \mathbf{f}^{(j)} = \mathbf{f}^{(j)} \circ \mathbf{f}^{(i)}, \tag{15}$$

then

$$\mathbf{f}^{(i)} \circ \left(\mathbf{f}^{(j)}\right)^{-1} = \left(\mathbf{f}^{(j)}\right)^{-1} \circ \mathbf{f}^{(i)} \tag{16}$$

and therefore

$$\mathbf{f}^{(j\to i)} = \mathbf{t}^{(i\to j)}. \tag{17}$$

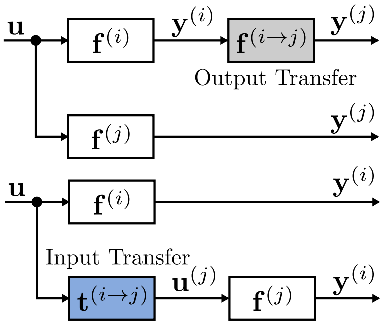

Fig. 2: Output transfer map $\mathbf{f}^{(i\to j)}$(top) and input transfer map $\mathbf{t}^{(i\to j)}$(bottom). The output transfer map describes the relation in outputs of two heterogeneous systems to the same input, whereas the input transfer map describes the relation in inputs for heterogeneous systems that yield the same output.

Remark 6 (Commutation of input/output maps): In the linear case (see Remark 1), the input/output maps $\mathbf{f}^{(i)}$, $\mathbf{f}^{(j)}$ are (left) multiplications by LTT-matrices $\mathbf{P}^{(i)}$ and $\mathbf{P}^{(j)}$. As LTT-matrices of the same dimensions commute and all LTT-matrices are invertible, (17) *holds in the linear case. Conditions for non-linear maps can be found in [19].*

C. Estimation of the Input Transfer Map and Input Transfer

We consider the following problem. There is a source agent i and a set of $A_\mathrm{t} \in \mathbb{N}$ target agents $j \in \{1, \ldots, A_\mathrm{t}\}$, all of which, source and target, with unknown and possibly non-linear dynamics $\mathbf{f}^{(i)}$ and $\mathbf{f}^{(j)}$, respectively. Furthermore, we are given $R \in \mathbb{N}$ pairs of input/output signals $(\mathbf{u}_r^{(i)}, \mathbf{y}_r^{(i)})$ of the source agent i, with $r \in \{1, \ldots, R\}$. For each agent j and each pair r of input/output signals $(\mathbf{u}_r^{(i)}, \mathbf{y}_r^{(i)})$, we want to find the input signal $\mathbf{u}_r^{(j)}$, that, when applied to the target agent j, yields the corresponding output signal $\mathbf{y}_r^{(i)}$ of the source agent. The obvious approach would be to identify the individual dynamics of the source and target systems directly and use (11) to calculate the corresponding input transfer map. However, this approach has two major drawbacks:

1) Identifying the individual system dynamics of source and all target systems can require large amounts of data and can be difficult for high-order systems.
2) To obtain a stable input transfer map, the source dynamics and the inverse of the target dynamics need to be stable.

Identifying the input transfer map directly from input/output data of source and target agents would overcome the need for individual system identification. Moreover, when identifying an estimate $\hat{\mathbf{t}}^{(i\to j)}$ of the input transfer map $\mathbf{t}^{(i\to j)}$, we may impose several convenient restrictions (at the expense of estimation accuracy). For we example we could restrict $\hat{\mathbf{t}}^{(i\to j)}$ to be linear and low-order, only considering the relevant frequencies for the desired output signals. Restricting $\hat{\mathbf{t}}^{(i \to j)}$ to be stable is not strictly necessary, however, for large N or in the presence of noise, may be beneficial.

However, we cannot identify $\mathbf{t}^{(i \to j)}$ from input/output data without knowledge of at least one dynamic model or without an iterative process. We instead use the relationship between input and output transfer as shown in (17) and assume that this relationship is at least approximately valid for the underlying dynamic models.

Assumption 4 (Commutation of input/output maps): Given a source agent i and target agents j with dynamics $\mathbf{f}^{(i)}$ and $\mathbf{f}^{(j)}$, respectively, we assume that (17) *approximately holds for the source and target agent dynamics, so that*

$$\mathbf{t}^{(i \to j)} \approx \mathbf{f}^{(j \to i)} . \tag{18}$$

To obtain an estimate $\hat{\mathbf{t}}^{(i \to j)}$ from input/output data of the source agent i and one target agent j, we propose the following general procedure:

1) Excite source agent i and target agent j with the same input $\mathbf{u}_{\text{est}} \in \mathbb{R}^{N_{\text{est}}}$ and denote the corresponding output signals by $\mathbf{y}_{\text{est}}^{(i)} \in \mathbb{R}^{N_{\text{est}}}$ and $\mathbf{y}_{\text{est}}^{(j)} \in \mathbb{R}^{N_{\text{est}}}$.
2) Determine an estimate $\hat{\mathbf{f}}^{(j \to i)}$ of the output transfer map $\mathbf{f}^{(j \to i)}$ from the outputs $\mathbf{y}_{\text{est}}^{(i)}$ and $\mathbf{y}_{\text{est}}^{(j)}$. Depending on the applications, restrictions such as the ones discussed above, may be imposed on the estimate.
3) Given Assumption 4, let the input transfer map of agent i and j be $\hat{\mathbf{t}}^{(i \to j)} \approx \hat{\mathbf{f}}^{(j \to i)}$.
4) Use $\hat{\mathbf{t}}^{(i \to j)}$ to transfer the input signals $\mathbf{u}_r^{(i)}$, i.e., $\mathbf{u}_r^{(j)} = \hat{\mathbf{t}}^{(i \to j)}(\mathbf{u}_r^{(i)}), r \in \{1, \dots, R\}$ (see Fig. 3).

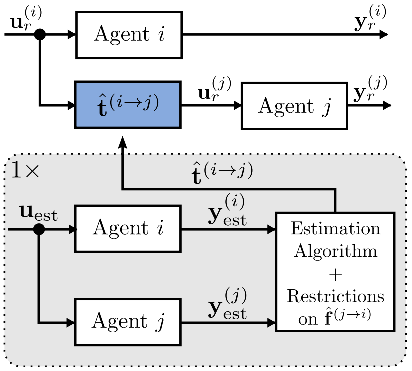
$\mathbf{u}_r^{(i)}$
Agent i
$\mathbf{y}_r^{(i)}$
$\hat{\mathbf{t}}^{(i \to j)}$
$\mathbf{u}_r^{(j)}$
Agent j
$\mathbf{y}_r^{(j)}$
1×
$\hat{\mathbf{t}}^{(i \to j)}$
$\mathbf{u}_{\text{est}}$
Agent i
$\mathbf{y}_{\text{est}}^{(i)}$
Estimation Algorithm + Restrictions on $\hat{\mathbf{f}}^{(j \to i)}$
Agent j
$\mathbf{y}_{\text{est}}^{(j)}$

Fig. 3: Estimating the input transfer map $\mathbf{t}^{(i \to j)}$ from agents i and j's outputs $\mathbf{y}_{\text{est}}^{(i)}$ and $\mathbf{y}_{\text{est}}^{(j)}$ to a common input signal $\mathbf{u}_{\text{est}}$. The estimated input transfer map can then be used to transfer inputs $\mathbf{u}_r^{(i)}$ from the source agent i to the target agent j with $\mathbf{u}_r^{(j)} = \hat{\mathbf{t}}^{(i \to j)}(\mathbf{u}_r^{(i)})$.

The procedure leaves two degrees of freedom for implementation:

1) the choice of estimation input $\mathbf{u}_{\text{est}}$
2) the restrictions on and estimation algorithm for $\hat{\mathbf{f}}^{(j \to i)}$, respectively $\hat{\mathbf{t}}^{(j \to i)}$.

In most multi-agent systems, we can assume that source and target dynamics have an equal relative degree $m^{(i)} = m^{(j)}$. Then, from Def. 1, $m^{(j \to i)} = 0$. In the following, we restrict the estimate $\hat{\mathbf{f}}^{(j \to i)}$ of the output transfer map to be linear and, because of $m^{(j \to i)} = 0$, model it as a biproper transfer function $F^{(j \to i)}(z)$ of fixed degree $D \in \mathbb{N}$, i.e.

$$F^{(j \to i)}(z) = \alpha \cdot \frac{1 + \sum\limits_{d=1}^{D} a_d z^{-d}}{1 + \sum\limits_{d=1}^{D} b_d z^{-d}} , \tag{19}$$

with $\alpha, a_d, b_d \in \mathbb{R}, d \in \{1, \dots, D\}$ being $2D + 1$ unknown parameters.

As $\mathbf{u}_{\text{est}}$ we choose an input sequence with a frequency spectrum that covers the frequencies found in the input signals to be transferred. A common choice is to use a sine-sweep as shown in Fig. 4. We use standard system identification methods to identify $F^{(j \to i)}(z)$ from the outputs $\mathbf{y}_{\text{est}}^{(i)}$ and $\mathbf{y}_{\text{est}}^{(j)}$.

We construct the corresponding $N \times N$ LTT-matrix

$$\mathbf{T}^{(i \to j)} = \begin{bmatrix} p_1 & 0 & \cdots & 0 \\ p_2 & p_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ p_N & p_{N-1} & \cdots & p_1 \end{bmatrix} \tag{20}$$

where $p_k \in \mathbb{R}, k \in [1, \dots, N]$ are the first N samples of the impulse response of $F^{(j \to i)}(z)$. As discussed above, we assume that this also represents a valid approximate of the input transfer map, i.e., $\hat{\mathbf{T}}^{(i \to j)} \approx \hat{\mathbf{F}}^{(j \to i)}$, and transfer the input signals $\mathbf{u}_r^{(i)}, r \in \{1, \dots, R\}$ by

$$\mathbf{u}_r^{(j)} = \hat{\mathbf{t}}^{(i \to j)} \left(\mathbf{u}_r^{(i)} \right) = \hat{\mathbf{T}}^{(i \to j)} \cdot \mathbf{u}_r^{(i)} . \tag{21}$$

Remark 7 (Assumptions in the proposed approach): Recall that the suggested procedure rests on the assumption that $\hat{\mathbf{f}}^{(j \to i)} \approx \hat{\mathbf{t}}^{(i \to j)}$ (see Condition (12)*). Adequateness of the assumption and the restrictions imposed on the estimation $\hat{\mathbf{f}}^{(j \to i)}$ clearly depend on the particular applications. For the application scenarios in the following sections, we do not attempt to justify adequateness ex ante, but argue ex post by demonstrating successful input transfer by evaluating simulation or experimental results.*

IV. EVALUATION

We evaluate the proposed method in two scenarios:

- a simulation of muscle stimulation dynamics in Sec. IV-B and
- real-world experiments using a group of inverted pendulum robots and a sim-to-real scenario in Sec. IV-C.

We aim to show that the proposed method can

1) estimate an input transfer map $\hat{\mathbf{t}}^{(i \to j)}$ from short training sequences recorded on a source and (multiple) target systems,
2) allow the transfer of different input signals from a source to multiple target systems such that the resulting outputs on the target systems are close to the desired outputs of the source system and
3) work in an experimental sim-to-real environment with real robots with unknown, non-linear and non-minimum phase dynamics .

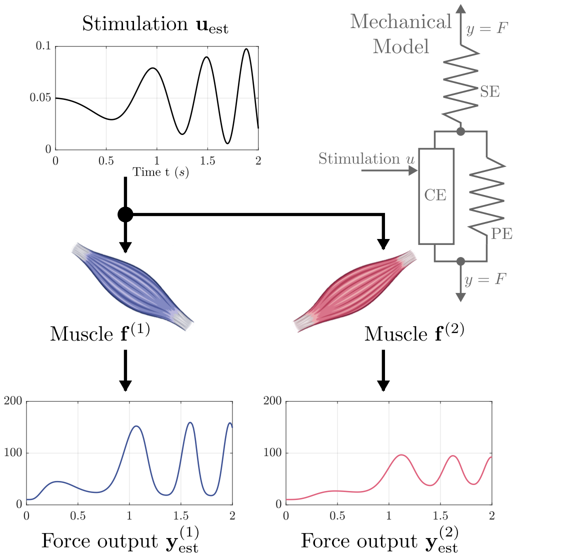
Stimulation $\mathbf{u}_{\text{est}}$
0.1
0.05
0
0
0.5
1
1.5
2
Time t (s)
Mechanical
Model
$y = F$
SE
Stimulation u
CE
PE
$y = F$
Muscle $\mathbf{f}^{(1)}$
Muscle $\mathbf{f}^{(2)}$
200
100
0
0
0.5
1
1.5
2
200
100
0
0
0.5
1
1.5
2
Force output $\mathbf{y}_{\text{est}}^{(1)}$
Force output $\mathbf{y}_{\text{est}}^{(2)}$

Fig. 4: Simulation of muscle dynamics using the Hill muscle model. It consists of a serial element (SE), a parallel element (PE) and a contractile element (CE) modelling the muscle's force response F to a stimulation input u. The short input signal $\mathbf{u}_{\text{est}}$ is applied to both muscles $\mathbf{f}^{(1)}$ and $\mathbf{f}^{(2)}$ and from their responses $\mathbf{y}_{\text{est}}^{(1)}$, $\mathbf{y}_{\text{est}}^{(2)}$ we estimate the corresponding input transfer map.

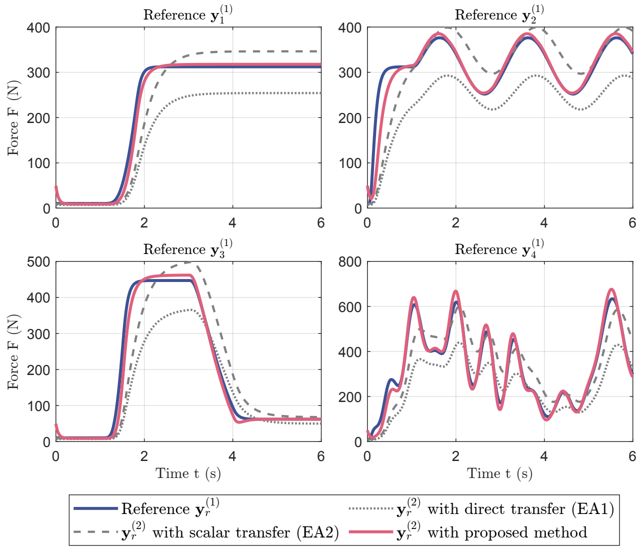

Fig. 5: Reference outputs $\mathbf{y}_r^{(1)}$ of muscle 1 (blue solid line) and outputs $\mathbf{y}_r^{(2)}$ of system 2 using the different methods for input transfer. The proposed method (red solid line) yields outputs close to the desired reference outputs. Direct transfer (grey dotted line) and scalar transfer (gray dashed line) do not generate outputs that are close to the desired references.

A. Comparison and Transfer Metric

We compare the proposed method to the following established approaches:

1) *(AE1)* Direct transfer of the input signal from source to target system (e.g. used in [11], [13]), i.e. taking $\hat{\mathbf{t}}^{(i\to j)}$ as the identity map.
2) *(AE2)* Use scalar multiplication as a transfer map, i.e. $\hat{\mathbf{t}}^{(i\to j)}(\mathbf{u}) = \alpha^* \cdot \mathbf{u}$ (see [4], [5], [16]), with

$$\alpha^* = \arg\min_{\alpha} \left\| \alpha \cdot \mathbf{y}_{\text{est}}^{(i)} - \mathbf{y}_{\text{est}}^{(j)} \right\|_2 \in \mathbb{R}\,. \tag{22}$$

To evaluate the input transfer map $\hat{\mathbf{t}}^{(i\to j)}$ we are using the following metrics:

Let $\varepsilon^{(i\to j)}(\mathbf{u}) \in \mathbb{R}$ be the *input transfer error* with

$$\varepsilon^{(i\to j)}(\mathbf{u}) = \sqrt{\frac{1}{N}\left[\mathbf{y}^{(i)} - \mathbf{y}^{(j)}\right]^T \left[\mathbf{y}^{(i)} - \mathbf{y}^{(j)}\right]} \tag{23}$$

with $\mathbf{y}^{(i)}$ being the desired output of the source agent i and $\mathbf{y}^{(j)}$ the output of target agent j after applying the input $\hat{\mathbf{t}}^{(i\to j)}(\mathbf{u})$. Furthermore let $\varepsilon_{\text{rel}}^{(i\to j)}(\mathbf{u}) \in \mathbb{R}$ be the *relative input transfer error* with

$$\varepsilon_{\text{rel}}^{(i\to j)}(\mathbf{u}) = \frac{\varepsilon^{(i\to j)}(\mathbf{u})}{\varepsilon_{\text{dir}}^{(i\to j)}(\mathbf{u})}\,. \tag{24}$$

The error $\varepsilon_{\text{dir}}^{(i\to j)}$ is the input transfer error when using a direct transfer of the input signal without transformation of the input.

B. Simulation Example

As a simulative example, we use the Hill muscle model [20], which describes a muscle's force response F to a stimulation signal u as shown in Fig. 4. The model is used frequently to simulate the non-linear muscle dynamics for stimulation.

We consider two muscles with dynamics $\mathbf{f}^{(1)}$, $\mathbf{f}^{(2)}$ which differ in their model parameters and their activation dynamics. Four different pairs of input/output signals $(\mathbf{u}_r^{(1)}, \mathbf{y}_r^{(1)}), r \in \{1, \ldots, 4\}$ from the source agent 1 are given. The goal is to estimate an input transfer map $\hat{\mathbf{t}}^{(1\to 2)}$ that can be used to calculate the input signals $\mathbf{u}_r^{(2)}$ that, when applied to agent 2, yield the outputs $\mathbf{y}_r^{(2)} = \mathbf{y}_r^{(1)}$. A short sine-sweep input signal $\mathbf{u}_{\text{est}}$ is applied to both muscles and the outputs $\mathbf{y}_{\text{est}}^{(1)}$ and $\mathbf{y}_{\text{est}}^{(2)}$ are recorded (see Fig. 4). Based on the outputs of both systems, we use the method proposed in Sec. III-C with an order $D = 2$ (an for comparison the established approaches EA1, EA2 from Sec. IV-A) to estimate the input transfer map $\hat{\mathbf{t}}^{(1\to 2)}$.

The estimated input transfer map is then used to transfer the reference inputs $\mathbf{u}_r^{(1)}, r \in \{1, \ldots, 4\}$ to system 2 with $\mathbf{u}_r^{(2)} = \hat{\mathbf{t}}^{(1\to 2)}(\mathbf{u}_r^{(1)})$. In Fig. 5 the desired output signals $\mathbf{y}_r^{(1)}$ and the output signals $\mathbf{y}_r^{(2)}$ of system 2 after applying the transferred inputs $\mathbf{u}_r^{(2)}$ are shown for the three different methods.

The input transfer errors $\varepsilon_r^{(1\to 2)}$ and relative input transfer

TABLE I

Method	ε_1	$\varepsilon_{\text{rel},1}$	ε_2	$\varepsilon_{\text{rel},2}$	ε_3	$\varepsilon_{\text{rel},3}$	ε_4	$\varepsilon_{\text{rel},4}$
Direct Transfer	69.4	1	88.4	1	86.5	1	118	1
Scalar Transfer	42.2	0.61	59.5	0.67	71.9	0.83	92.6	0.78
Proposed Method	**11.6**	**0.16**	**25.2**	**0.29**	**21.0**	**0.24**	**22.8**	**0.19**

Input transfer error ε_r and relative input transfer error $\varepsilon_{\text{rel},r}$ for the four different reference signals and three different methods in the simulation study. Smaller values indicate a better transfer. Bold values denote the best result for the corresponding signal.

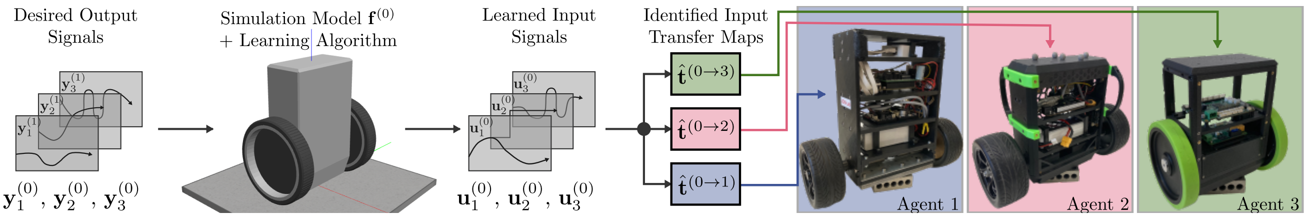
Desired Output Signals
$\mathbf{y}_3^{(1)}$
$\mathbf{y}_2^{(1)}$
$\mathbf{y}_1^{(1)}$
$\mathbf{y}_1^{(0)}, \mathbf{y}_2^{(0)}, \mathbf{y}_3^{(0)}$
Simulation Model $\mathbf{f}^{(0)}$ + Learning Algorithm
Learned Input Signals
$\mathbf{u}_3^{(0)}$
$\mathbf{u}_2^{(0)}$
$\mathbf{u}_1^{(0)}$
$\mathbf{u}_1^{(0)}, \mathbf{u}_2^{(0)}, \mathbf{u}_3^{(0)}$
Identified Input Transfer Maps
$\hat{\mathbf{t}}^{(0\to3)}$
$\hat{\mathbf{t}}^{(0\to2)}$
$\hat{\mathbf{t}}^{(0\to1)}$
Agent 1
Agent 2
Agent 3

Fig. 6: Experimental evaluation using a sim-to-real scenario for a group of three inverted pendulum robots. For three desired output signals $\mathbf{y}_r^{(0)}, r \in \{1,2,3\}$ the corresponding input signals $\mathbf{u}_r^{(0)}$ are learned in-silico using a simplified simulation model $\mathbf{f}^{(0)}$ and then transferred to the three real agents using the identified input transfer maps $\hat{\mathbf{t}}^{(0\to j)}, j \in \{1,2,3\}$.

errors $\varepsilon_{\text{rel},r}^{(1\to2)}$ for the four different signals and three different methods are shown in Tab. I. Direct transfer of the input signals (EA1) leads to a mean input transfer error of 90.6 N with significant difference between the desired and actual output signals as seen in Fig. 5. The scalar input transfer map (EA2) achieves a reduction in error as compared to the direct transfer with a mean relative input error of 0.72 and a mean absolute error of 66.6 N. However, the output signals also show significant differences to the desired output signals. The proposed method yields output signals close to the desired reference and achieves the smallest errors for all signals, with a mean input error of 20.2 N and a mean relative input error of 0.22 in comparison to the direct transfer. However, due to the non-linearity of the underlying systems (i.e. not fulfilling the assumptions as outlined in Remark 7) the transfer is not perfect.

For the considered example, we have shown that we can successfully estimate an input transfer map from only a short initial stimulation despite the non-linearity and dissimilar dynamics of the individual muscles. We have also shown that we can use this map to successfully transfer multiple different input signals from the source muscle to the target muscle and achieve force outputs that are close to the output of the source system.

C. Experimental Evaluation

To evaluate the method in a real-world robotic sim-to-real scenario, we are using so-called two-wheeled inverted pendulum robots (TWIPR), which have unstable, non-linear and non-minimum phase dynamics [21] (see Fig. 6). For the robots to stay upright, we use a state feedback controller and we consider the feedback-controlled system as the agent. The input to the controlled system $u^{(j)}$ is the external input to the feedback-controller and has the unit of a torque. As output $y^{(j)}$ we are using the robot's pitch angle θ in radians. The output angle θ is determined using an IMU mounted on the robot.

We consider a scenario with three real-world robotic agents $j \in \{1,2,3\}$ with physical differences (see Tab. II) that lead to significantly different behaviour of the closed-loop systems. We are further considering three different desired output signals $\mathbf{y}_r^{(0)}, r \in \{1,2,3\}$ of differing lengths. The goal is to find for all agents $j \in \{1,2,3\}$ the input signals $\mathbf{u}_r^{(j)}, r \in \{1,2,3\}$ that yield the corresponding desired outputs $\mathbf{y}_r^{(0)}$.

TABLE II

Agent j	**Mass** m **(kg)**	**COG** l_{cg} **(mm)**	**Inertia** I_{yy} **(g·cm^2)**
1	3.9	34.1	165200
2	1.8	20.7	102400
3	0.9	14.1	34300

Physical differences of the robotic agents $i \in \{1,2,3\}$

We are using a simplified, linear simulation model $\mathbf{f}^{(0)}$ of arbitrary TWIPR dynamics and an iterative learning control algorithm to learn the corresponding input signals $\mathbf{u}_r^{(0)}, r \in \{1,2,3\}$ in-silico that yield the desired output signals $\mathbf{y}_r^{(0)}$ for the simulation model (see [3], [22]).

We apply a short test input $\mathbf{u}_{\text{est}}$ to the simulation model and all real-world agents with a length of $T_{\text{est}} = 10\,\text{s}$ ($N_{\text{est}} = 1000$) and record the corresponding output signals (see Fig. 7). From the recorded outputs of the simulation model $\mathbf{y}_{\text{est}}^{(0)}$ and the three robotic agents $\mathbf{y}_{\text{est}}^{(j)}$ we estimate the three input transfer maps $\hat{\mathbf{t}}^{(0\to j)}$ from the simulation to the real-world agents using the method proposed in III-C with a degree of $D = 2$. These maps are then used to determine the inputs $\mathbf{u}_r^{(j)} = \hat{\mathbf{t}}^{(0\to j)}(\mathbf{u}_r^{(0)}), r \in \{1,2,3\}$.

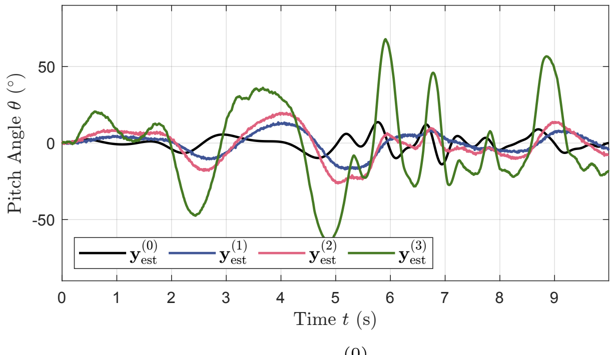

Fig. 7: The simulation model's output $\mathbf{y}_{\text{est}}^{(0)}$ and the three agents' outputs $\mathbf{y}_{\text{est}}^{(1)}, \mathbf{y}_{\text{est}}^{(2)}, \mathbf{y}_{\text{est}}^{(3)}$ to the input $\mathbf{u}_{\text{est}}$. Note the significant difference between the individual responses, indicating different closed-loop dynamics.

Remark 8 (Output Constraints): Each agent has an output constraint $-\frac{\pi}{2} < \theta_k < \frac{\pi}{2}$, i.e. it is not allowed to touch the ground with the robot's body, since this alters the robot's dynamics. If a robot touches the ground during an experiment at a time instant $k < N$, the experiment is aborted. The transfer error is only calculated up to the time instant k.

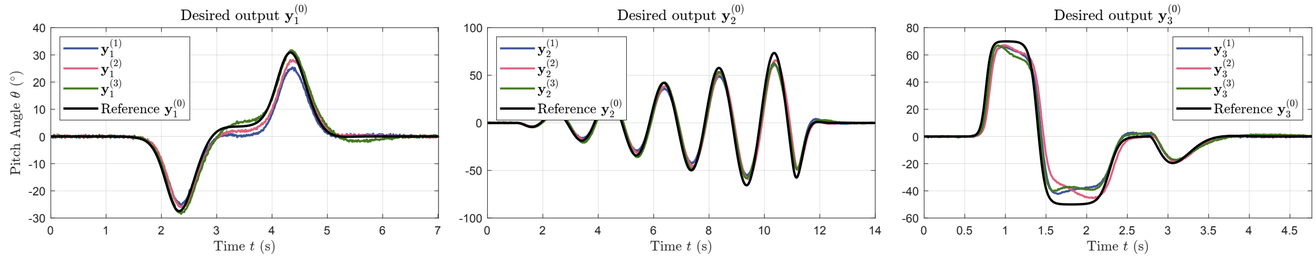

Fig. 8: Desired output signals $\mathbf{y}_r^{(0)}, r \in \{1, 2, 3\}$ and the outputs $\mathbf{y}_r^{(j)}$ of the three agents $j \in \{1, 2, 3\}$ after input transfer using the proposed method.

TABLE III

	$\bar{\varepsilon}_1$	$\bar{\varepsilon}_2$	$\bar{\varepsilon}_3$
Direct Transfer (EA1)	28.1°	35.6°	45.0°
Scalar Transfer (EA2)	9.4°	24.1°	25.2°
Proposed Method	**1.9°**	**5.2°**	**6.5°**

Mean input transfer error $\bar{\varepsilon}_r = \frac{1}{3}\sum_{i=1}^{3} \varepsilon_r^{(0\to i)}$ over the three agents for the desired reference outputs $\mathbf{y}_r^{(0)}, r \in \{1, 2, 3\}$.

The three desired output signals $\mathbf{y}_r^{(0)}$ and the outputs $\mathbf{y}_r^{(i)}$ of the three target agents $j \in \{1, 2, 3\}$ after input transfer using the proposed method are shown in Fig. 8. Reference output $\mathbf{y}_1^{(0)}$ describes a backwards and forwards motion with moderate pitch angles. Reference output $\mathbf{y}_2^{(0)}$ is a sinusoidal pendulum motion with increasing amplitude up to 70°. The third reference $\mathbf{y}_3^{(0)}$ is a quick forward diving motion and abrupt breaking. It features high pitch angles and a steep transient in the breaking motion. For all three desired output signals and all agents, the agents' outputs $\mathbf{y}_r^{(i)}$ with the proposed method for input transfer are close to the desired output with only short deviations. In Tab. III, the mean input transfer errors $\bar{\varepsilon}_r = \frac{1}{3}\sum_{j=1}^{3} \varepsilon_r^{(0\to j)}$ for each desired output signal $\mathbf{y}_r^{(r)}, r \in \{1, 2, 3\}$ over all agents j are shown. For each of the desired outputs, the proposed method performs significantly better than direct transfer or scalar transfer of the input signals, with mean errors below 7°. The mean input transfer errors over all agents and desired output signals $\bar{\varepsilon} = \frac{1}{3}\sum_{r=1}^{3} \bar{\varepsilon}_r$ for direct transfer (EA1) and scalar transfer (EA2) are significantly higher with $\bar{\varepsilon}_{\text{EA1}} = 36.2°$ and $\bar{\varepsilon}_{\text{EA2}} = 19.6°$.

In Fig. 9, the desired output signal $\mathbf{y}_3^{(0)}$ and the outputs $\mathbf{y}_3^{(1)}$ of agent 1 for the three different methods are shown in comparison. Direct transfer (EA1) and scalar transfer (EA2) of the input signal yield output signals with significant errors ($\varepsilon_{3,\text{EA1}}^{(0\to1)} = 29.6°$, $\varepsilon_{3,\text{EA2}}^{(0\to1)} = 33.1°$), whereas the proposed method yields an output close to the desired one with $\varepsilon_3^{(0\to1)} = 6.1°$. Only during the first abrupt breaking motion between $t = 1.5\,\text{s}$ and $t = 2.5\,\text{s}$ does the output of agent 3 differ significantly from the desired reference. This might be due to non-linearities such as friction or slippage, which are possibly not covered well in the input transfer map.

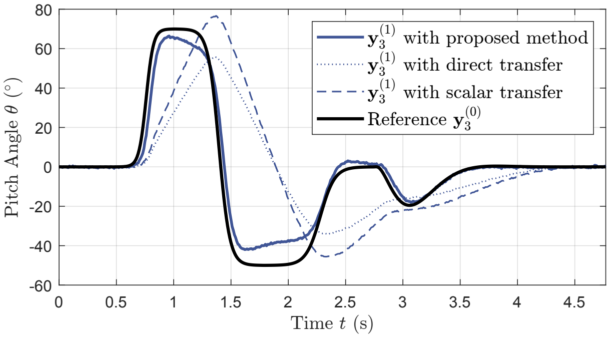

Fig. 9: Comparison between the three methods for input transfer for agent 1 and output reference $\mathbf{y}_3^{(0)}$. Direct transfer and scalar transfer do not yield an output close to the desired reference, where as the proposed method successfully yields an output close to the desired reference with an input transfer error $\varepsilon_3^{(0\to1)} = 6.1°$.

We have successfully shown, that the proposed method works well in an experimental scenario in which different in-silico learned input signals are successfully transferred to three different real-world agents, despite their non-linear, non-minimum phase dynamics. The proposed method works significantly better than the two established approaches we compared it to.

V. CONCLUSIONS AND FUTURE WORKS

We proposed a new method for input transfer between heterogeneous dynamics without individual model knowledge. The method estimates an input transfer map from short experimental input/output sequences on source and target systems, which can then be used to transfer different input signals from source to target systems. We have shown in a simulation example and in experiment with a robotic non-linear MAS that the proposed method performs significantly better than established alternative methods, namely direct or scalar transfer in the same scenarios.

The proposed method allows for data-efficient estimation of the input transfer map and the transfer of input signals from source to multiple target systems. This can significantly improve the learning time in a sim-to-real scenario, in which a simulation is used for learning and the learned input signals are transferred to the target systems. It also enables the transfer of signals between heterogeneous agents in a MAS during cooperative learning.

Up to now, we have only justified the assumption $\mathbf{f}^{(j\to i)} \approx \mathbf{t}^{(i\to j)}$ and the restrictions imposed on the identified input transfer map $\hat{\mathbf{t}}^{(i\to j)}$ by evaluating the outcome of simulation and experimental studies. While we know that the assumption holds for linear dynamics, future work will attempt to characterize classes of nonlinear dynamics for which the assumption is also valid. Future work will also include investigating the possibility of estimating the input transfer map directly, without the need for the output transfer map. This might allow the use of dynamic input transfer in more applications and for a wider class of system dynamics.

VI. ACKNOWLEDGMENTS

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2002/1 "Science of Intelligence" – project number 390523135.

REFERENCES

[1] D.A. Bristow, M. Tharayil, and A.G. Alleyne, "A survey of iterative learning control," *IEEE Control Systems*, vol. 26, no. 3, pp. 96–114, 2006.

[2] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, May 1996. [Online]. Available: https://doi.org/10.1613/jair.301

[3] M. Meindl, F. Molinari, D. Lehmann, and T. Seel, "Collective iterative learning control: Exploiting diversity in multi-agent systems for reference tracking tasks," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 4, pp. 1390–1402, 2022.

[4] K. V. Raimalwala, "Transfer learning for robotics: Can a robot learn from another robot's data?"

[5] K. V. Raimalwala, B. A. Francis, and A. P. Schoellig, "An upper bound on the error of alignment-based transfer learning between two linear, time-invariant, scalar systems," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 5253–5258.

[6] B. Bocsi, L. Csato, and J. Peters, "Alignment-based transfer learning for robot models," in *The 2013 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2013, pp. 1–7.

[7] V. L. Donatone, S. Meraglia, and M. Lovera, "H ∞ -based transfer learning for uav trajectory tracking," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2022, pp. 354–360.

[8] J. García, Fern, and o Fernández, "A comprehensive survey on safe reinforcement learning," *Journal of Machine Learning Research*, vol. 16, no. 42, pp. 1437–1480, 2015. [Online]. Available: http://jmlr.org/papers/v16/garcia15a.html

[9] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: a survey," in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2020, pp. 737–744.

[10] P. Christiano, Z. Shah, I. Mordatch, J. Schneider, T. Blackwell, J. Tobin, P. Abbeel, and W. Zaremba, "Transfer from simulation to real world through learning deep inverse dynamics model." [Online]. Available: http://arxiv.org/pdf/1610.03518v1

[11] S. Zhou, M. K. Helwa, A. P. Schoellig, A. Sarabakha, and E. Kayacan, "Knowledge transfer between robots with similar dynamics for high-accuracy impromptu trajectory tracking," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 1–8.

[12] M. J. Sorocky, S. Zhou, and A. P. Schoellig, "Experience selection using dynamics similarity for efficient multi-source transfer learning between robots," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2739–2745.

[13] M. Sorocky, S. Zhou, and A. P. Schoellig, "To share or not to share? performance guarantees and the asymmetric nature of cross-robot experience transfer." [Online]. Available: http://arxiv.org/pdf/2006.16126v1

[14] K. Pereida, M. K. Helwa, and A. P. Schoellig, "Data-efficient multirobot, multitask transfer learning for trajectory tracking," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1260–1267, 2018.

[15] K. Pereida, D. Kooijman, R. R. P. R. Duivenvoorden, and A. P. Schoellig, "Transfer learning for high-precision trajectory tracking through l1 adaptive feedback and iterative learning," *International Journal of Adaptive Control and Signal Processing*, vol. 33, no. 2, pp. 388–409, 2019.

[16] K. Raimalwala, B. Francis, and A. Schoellig, "A preliminary study of transfer learning between unicycle robots," 2016.

[17] M. K. Helwa and A. P. Schoellig, "Multi-robot transfer learning: A dynamical system perspective: Ieee/rsj international conference on intelligent robots and systems, vancouver, bc, canada september 24-28, 2017 : conference digest," pp. 4702–4708, 2017. [Online]. Available: http://ieeexplore.ieee.org/servlet/opac?punumber=8119304

[18] Z.-P. Jiang and Y. Wang, "Input-to-state stability for discrete-time nonlinear systems," *Automatica*, vol. 37, no. 6, pp. 857–869, 2001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109801000280

[19] J. F. Ritt, *Permutable Rational Functions*. Transactions of the American Mathematical Society, 1923.

[20] A. V. Hill, "The heat of shortening and the dynamic constants of muscle," *Proceedings of the Royal Society of London. Series B - Biological Sciences*, vol. 126, no. 843, pp. 136–195, Oct. 1938. [Online]. Available: https://doi.org/10.1098/rspb.1938.0050

[21] S. Nawawi, m. n. Ahmad, and J. Osman, "Development of a two-wheeled inverted pendulum mobile robot," 01 2008, pp. 1 – 5.

[22] M. Meindl, D. Lehmann, and T. Seel, "Bridging reinforcement learning and iterative learning control: Autonomous motion learning for unknown, nonlinear dynamics," *Frontiers in Robotics and AI*, vol. 9, 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frobt.2022.793512