In Proc. 10th IEEE-RAS International Conference on Humanoid Robotics, Dec. 6-8, Nashville, TN, USA (2010).

# Transferring Impedance Control Strategies Between Heterogeneous Systems via Apprenticeship Learning

Matthew Howard, Djordje Mitrovic and Sethu Vijayakumar

*Abstract*— **We present a novel method for designing controllers for robots with variable impedance actuators. We take an imitation learning approach, whereby we learn impedance modulation strategies from observations of behaviour (for example, that of humans) and transfer these to a robotic plant with very different actuators and dynamics. In contrast to previous approaches where impedance characteristics are directly imitated, our method uses task performance as the metric of imitation, ensuring that the learnt controllers are directly optimised for the hardware of the imitator. As a key ingredient, we use apprenticeship learning to model the optimisation criteria underlying observed behaviour, in order to frame a correspondent optimal control problem for the imitator. We then apply local optimal feedback control techniques to find an appropriate impedance modulation strategy under the imitator's dynamics. We test our approach on systems of varying complexity, including a novel, antagonistic series elastic actuator and a biologically realistic two-joint, six-muscle model of the human arm.**

## I. Introduction

In recent years, variable impedance actuation has become increasingly popular in the design and control of novel robotic mechanisms [10], [4]. Variable impedance actuators (VIAs) (Fig. 8) promise many benefits for the next generation of robots, including (i) increased safety in settings where there is human-robot interaction, (ii) increased dynamic range (e.g., when throwing, energy may be stored in spring-like VIAs, before being released explosively for the throw) and (iii) increased energy efficiency when interacting with the environment. However, despite these benefits, there are still a number of challenges associated with deploying such actuators to the current generation of robots. One major problem is that of how to control such mechanisms, and in particular, how to best utilise variable impedance so that the benefits (such as compliance) are realised, while compromise on other aspects of performance (such as precision) is avoided.

A promising approach to finding appropriate impedance control strategies on robots is to take examples from human behaviour and attempt to mimic it. The human musculoskeletal system, actuated by antagonistic muscles with inherent visco-elastic properties [7], represents one of the best examples of a system controlled with variable impedance actuation. A large body of research studying human impedance modulation exists in the biological literature and, as such, may be a rich source of inspiration for designing controllers for robots [6]. However, the difficulty with this is that human impedance strategies are highly adapted to the specific properties of the human body and may not transfer directly to those of robotic plants. For example, it is well-known that the human musculoskeletal system suffers from signal-dependant noise (SDN), that is, noise in the kinematics of movement in direct proportion to the control signal [5]. To counter the effects of SDN, humans adapt their impedance in different ways, depending on the task, e.g., in tasks requiring high precision, humans tend to increase stiffness by co-contracting [3]. However, most robotic systems do not suffer from such noise characteristics (e.g., noise is more commonly constant, additive and much smaller in magnitude) so direct transfer of the human impedance strategy may be inappropriate: maintaining the same level of stiffness on a less noisy robot would waste energy and reduce compliance without significantly improving accuracy.

To overcome problems such as these, in this paper, we suggest a novel approach to the problem of transferring impedance control strategies across plants with *heterogeneous dynamics and actuation*. Specifically, we employ an apprenticeship learning (AL) approach [11], [1], whereby we use recordings of optimal behaviour of a VIA system (such as a human), and seek optimisation criteria which, under that system's dynamics, can reproduce the behaviour. Having extracted these criteria in the form of a cost function, we then apply local optimal feedback control (OFC) techniques [12] to transfer the essential characteristics of the behaviour, to a new system with a very different dynamics and actuation. In our experiments, we assess the effectiveness of our approach for transferring behaviours across plants despite significant differences in their embodiment.

## II. Problem Definition

Our aim is to transfer optimal impedance control strategies from an expert demonstrator ($e$) to an apprentice learner ($l$) given that the expert and learner have a very different embodiment[1], both in terms of their dynamics and actuation. Specifically, we assume the expert has state ${}^e\mathbf{x} \in \mathbb{R}^n$, controls movement with commands ${}^e\mathbf{u} \in \mathbb{R}^m$, and has dynamics

$$ {}^e\dot{\mathbf{x}} = {}^e\mathbf{f}({}^e\mathbf{x}, {}^e\mathbf{u}) \quad \in \mathbb{R}^n. \tag{1}$$

Note that the effect of the commands ${}^e\mathbf{u}$ on the dynamics (i.e. the form of ${}^e\mathbf{f}(\cdot)$) depends on the actuation mechanism of the expert. In particular, we can rewrite (1) as

$$ {}^e\dot{\mathbf{x}} = {}^e\mathbf{g}({}^e\mathbf{x}, {}^e\boldsymbol{\tau}) \quad \in \mathbb{R}^n $$

[1] In principle, our method avoids making any assumption on the extent to which the expert and learner plants may differ. However, in order to make a meaningful comparison between their respective behaviours, we assume that there is a sufficient overlap in their capabilities, that they may both achieve similar success at a given task.

M. Howard, D. Mitrovic and S. Vijayakumar are with the Institute of Perception Action and Behaviour, University of Edinburgh, Scotland, UK. matthew.howard@ed.ac.uk

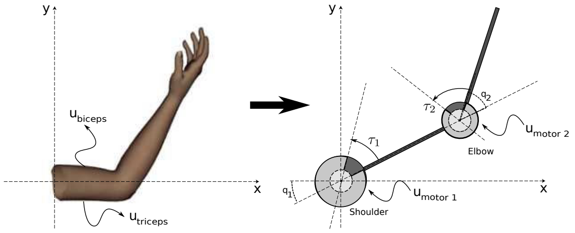


Fig. 1. Correspondence problem between human and robotic actuation systems. Left: Humans use muscle activations (e.g., $u_{triceps}$ and $u_{biceps}$) to control movement. Right: Robotic systems are controlled with command signals to the different motors (e.g., $u_{motor1}$ and $u_{motor2}$). The torque generated by those motors depends on the actuators used.

where ${}^e\boldsymbol{\tau} = {}^e\boldsymbol{\tau}({}^e\mathbf{x}, {}^e\mathbf{u})$ is the (in general, state-dependent) relationship between the expert's command signal ${}^e\mathbf{u}$ and the torques applied by the expert's actuators.

Our goal is to transfer behaviour to a learner with a different embodiment, both in terms of the dynamics and actuation. For example, we may wish to take control strategies measured from the human arm (actuated by antagonistic muscles) and apply them to a robotic manipulator (actuated by VIAs). We denote the learner's state as ${}^l\mathbf{x} \in \mathbb{R}^p$, command signal ${}^l\mathbf{u} \in \mathbb{R}^q$ and dynamics

$$ {}^l\dot{\mathbf{x}} = {}^l\mathbf{f}({}^l\mathbf{x}, {}^l\mathbf{u}) = {}^l\mathbf{g}({}^l\mathbf{x}, {}^l\boldsymbol{\tau}) \quad \in \mathbb{R}^p \tag{2} $$

where ${}^l\boldsymbol{\tau} = {}^l\boldsymbol{\tau}({}^l\mathbf{x}, {}^l\mathbf{u})$ denotes the torques produced by the learner's actuators. Note that, in general, the state and action space (${}^e\mathbf{x}, {}^e\mathbf{u}$ and ${}^l\mathbf{x}, {}^l\mathbf{u}$) may differ significantly between the two plants (for example, for a human expert ${}^e\mathbf{u}$ may correspond to muscle activations whereas for a robot learner ${}^l\mathbf{u}$ may correspond to desired position of a servo-motor). In addition, ${}^l\mathbf{f}(\cdot)$ and ${}^e\mathbf{f}(\cdot)$ may also differ, both in terms of the parameter values (e.g., inertia, link lengths, joint axis positions and orientations), and the way in which they enter the dynamics equations.

### A. *Correspondence Problem*

Clearly, these differences in embodiment cause difficulties when attempting to transfer behaviour and this correspondence problem is particularly severe in the dynamics domain with differences in actuation. As an example, consider the problem of transferring the control strategy used by a human to perform some task (e.g., punching a target) to a robotic imitator, as illustrated in Fig. 1. Imagine that we are given a set of recordings of the behaviour (e.g, in the form of muscle activation profiles) and we wish to use this data to reproduce the movement on a robotic system. Depending on the hardware, there are a number of approaches we may take.

Firstly, if there is a close correspondence between the robot and the human, simplest approach would be to attempt to *directly imitate the behaviour*, i.e., define ${}^e\mathbf{u} \approx {}^l\mathbf{u}$. This may be possible in special cases where the dynamics and actuation of the robot are especially similar to that of the human, for instance, if the robot is actuated with artificial muscles (e.g., McKibben muscles [8]), it may be possible to directly feed the recorded muscle activations as a command signal to the robot actuators. Evidently, this approach has the benefit of simplicity but its applicability is very limited since such direct correspondence between demonstrator and imitator is rare.

A second, and by far more common approach, is to do *feature-based imitation* of the observed behaviour. The basis of this approach is to define correspondence between salient features of the demonstrated behaviour ${}^e\psi({}^e\mathbf{x}(t), {}^e\mathbf{u}(t))$ and certain 'equivalent' features of the robot's behaviour ${}^l\psi({}^l\mathbf{x}, {}^l\mathbf{u})$ [2]. For example, in the example in Fig. 1, these features might include the stiffness and damping profiles of the human arm that occur during movement. By drawing an equivalence between these and the impedance of the robot, the feature-based approach imitates behaviour by matching those features as closely as possible during the movement.

The downside of this approach, however, is that it does not take into account the way in which the features affect *task performance* under the dynamics of different plants. For example, in a point-to-point reaching task, the impedance profile of the human may be relatively high toward the end of the movement to ensure that the target is hit accurately (i.e., to counter the effects of SDN). Naturally, this comes at the cost of increased energy expenditure, since the human must co-contract to achieve this. However, for a (less noisy) robotic imitator, this may not be optimal, since the robot may be fairly accurate (compared to the human) even at relatively low impedance. As such, a better strategy for the robot might be to keep the impedance at a steady, low level throughout the movement, thereby avoiding unnecessary energy consumption, but still achieving the task to the desired level of accuracy.

### B. *Apprenticeship Learning for Task-based Imitation*

To avoid these problems, in this paper we take a different approach, in which the goal is to imitate *the objectives of the movement*, rather than mimicking specific features. Our approach is based on apprenticeship learning [11], [1], where the aim is to model the demonstrated behaviour indirectly in the form of an *objective function* with respect to which, the behaviour can be described as optimal. Representing the behaviour in this way, we can then seek equivalent *task goals* for the imitator by defining correspondence at the level of *the objective function that defines the task*. Furthermore, having learnt these objectives, we can then optimise the imitated behaviour in a way that also takes into account the *imitator's dynamics*.

Specifically, we assume that we are given a set of demonstrations $D$ of an expert performing a task, in the form of trajectories through the state-action space of the demonstrator, ${}^e\mathbf{x}, {}^e\mathbf{u}$ of duration[2] $T$. We assume that these trajectories can be described as optimal with respect to some (unknown) objective function

$$ {}^eJ = {}^eh({}^e\mathbf{x}(T)) + \int_0^T {}^el({}^e\mathbf{x}, {}^e\mathbf{u}, t)\, dt \tag{3} $$

where ${}^eh(\cdot), {}^el(\cdot) \in \mathbb{R}$ are cost functions defined on the state-action space of the demonstrator. For example, ${}^el({}^e\mathbf{x}, {}^e\mathbf{u}, t)$ may describe the instantaneous work done by the demonstrator's actuators (e.g., the energy consumed by human muscles at a given activation). Note that here, since the optimality of

[2] For simplicity, through the paper we assume finite length trajectories of equal length. However, as discussed in [1], AL techniques are also readily extended to infinite horizon tasks.

the trajectories $D$ depends on the demonstrator's dynamics ${}^e\mathbf{f}(\cdot)$, the recorded trajectories will not, in general, be optimal under the dynamics of a different (learner) system ${}^l\mathbf{f}(\cdot)$, i.e., $\{{}^e\bar{\mathbf{x}}, {}^e\bar{\mathbf{u}} \,|\, {}^e\mathbf{f}(\cdot)\} \neq \{{}^l\bar{\mathbf{x}}, {}^l\bar{\mathbf{u}} \,|\, {}^l\mathbf{f}(\cdot)\}$. In other words, direct imitation on the learner plant is, in general, *suboptimal* when considering the imitator's dynamics.

Instead, we propose to imitate behaviour based on *correspondence in the objective functions* between expert and learner. The key to our approach is to define an *equivalent objective function*

$$
{}^lJ = {}^lh({}^l\mathbf{x}(T)) + \int_0^T {}^ll({}^l\mathbf{x}, {}^l\mathbf{u}, t)\, dt \tag{4}
$$

defined on the learner's state-action space, where the terms ${}^lh(\cdot), {}^ll(\cdot) \in \mathbb{R}$ define cost terms with a meaningful correspondence to those of the expert ${}^eh(\cdot), {}^el(\cdot)$. For example, if the term ${}^el({}^e\mathbf{x}, {}^e\mathbf{u}, t)$ of a human demonstrator represents the energy consumption of the muscles, one might define ${}^ll({}^l\mathbf{x}, {}^l\mathbf{u}, t)$ as the power consumed by the motors of a robotic manipulator. The goal of imitation then, is to find the optimal behaviour for the learner $\{{}^l\bar{\mathbf{x}}, {}^l\bar{\mathbf{u}}\}$ under the dynamics ${}^l\mathbf{f}(\cdot)$ with respect to the *equivalent objective function* (4).

Note that, similar to feature-based approaches to imitation, the ease with which we can define correspondent cost functions (3)-(4) will depend on the specific embodiments of the two plants. For example, cost terms dependent on features such as end-effector position may be defined as exactly correspondent, whereas terms dependent on other properties such as the applied torque or impedance may require more complex definitions. However, a major benefit of our approach is that, often it is much easier to define correspondence at the level of the task, rather than at the detailed control level of the plants. For instance, when imitating human behaviour (Fig. 1), the selection of which dynamics characteristics to match (e.g., impedance profiles, torques etc.) in a feature-based imitation approach will depend critically on the effect those have on the dynamics of the two plants. In contrast, with task-based imitation, we only need to specify the salient features (e.g., target accuracy, energy consumption) and the low-level details of the behaviour will automatically be handled by optimal control. In the next section we turn to the implementation details of our approach.

## III. Method

A schematic overview of the proposed approach is illustrated in Fig. 2, showing the processing steps, and the inputs required at each stage. Reading from the top left, we first collect demonstrations from an expert (e.g., a human) performing some task. This is fed into a module for apprenticeship learning along with information about the expert's dynamics. Based on this information, a parametric model of the expert's cost function is learnt with parameters $\hat{\mathbf{w}}$.

The output of this module is then fed to a second module for optimal feedback control. This takes the learnt parameters and applies them to a correspondent cost function model. The OFC module finds the optimal control strategy for the imitator, with respect to this learnt cost function, using a model of the imitator dynamics. The resultant controller is finally sent to the robot for execution. In the following we briefly describe the details of the AL and OFC components.

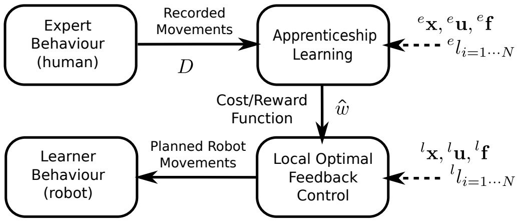


Fig. 2. Schematic of our task-based imitation framework for behaviour transfer.

### A. Multiplicative Weights Apprenticeship Learning

For the AL component, we use an approach called Multiplicative Weights Apprenticeship Learning (MWAL) recently proposed in [11]. The algorithm is based on principles of adversarial game theory, and as such has been shown to be a robust method for AL. Furthermore, due to its efficiency it is well suited for learning in the robotics domain, where state-action spaces are typically high-dimensional and continuous.

The method works on data that is given as a set of $K$ trajectories $D = \left\{({}^e\mathbf{x}_0^k, {}^e\mathbf{u}_0^k), \cdots, ({}^e\mathbf{x}_T^k, {}^e\mathbf{u}_T^k)\right\}_{k=0}^K$ of states ${}^e\mathbf{x}$ and actions ${}^e\mathbf{u}$ recorded from the demonstrator with dynamics (1). These are assumed to be optimal with respect to a cost function of the form

$$
{}^eJ = \sum_{i=1}^{n_T} w_i {}^eh_i({}^e\mathbf{x}(T)) + \int_0^T \sum_{i=n_T}^{N} w_i {}^el_i({}^e\mathbf{x}, {}^e\mathbf{u}, t)\, dt \tag{5}
$$

with unknown weights $w_i$ (with $w_i > 0 \,\forall\, i$ and $\sum_i w_i = 1$). Here ${}^eh_i(\cdot), {}^el_i(\cdot) \in \mathbb{R}$ are a set of (known) basis functions: these may be made up of a set of bases for a generic function approximator (e.g., Gaussian radial basis functions), or a set of salient features of the task (e.g., energy or accuracy costs).

The idea behind MWAL is that the weights $w_i$ specifying the importance of the different components of the objective function (5) can be determined efficiently by comparing the expected value of the observed behaviour $D$ with that of a second set of trajectories ${}^mD$ that are optimal with respect to an estimate of (5) with weights $\hat{w}_i$. Specifically, since the cost bases ${}^eh_i(\cdot), {}^el_i(\cdot)$ are assumed known, we can estimate the value of the trajectories in $D$ and ${}^mD$, with respect to each of the bases separately. That is, for the $i$th basis function

$$
\tilde{v}_i = \frac{1}{K} \sum_{k=0}^{K} \int_0^T {}^el_i({}^e\mathbf{x}_k(t), {}^e\mathbf{u}_k(t), t)\, dt \tag{6}
$$

if it is a running cost and

$$
\tilde{v}_i = \frac{1}{K} \sum_{k=0}^{K} {}^eh_i({}^e\mathbf{x}_k(T)) \tag{7}
$$

if it is a terminal cost. We can then compare the difference in these value estimates to adjust the weights $\hat{w}_i$, by scaling up those for which the value of the expert trajectories is lower (indicating a stronger preference to minimise these components of the cost), and scaling down those for which the values are higher (indicating the opposite). In successive iterations, MWAL alternates between solving the forward optimal control problem under the current estimate of $\hat{\mathbf{w}}$ to find trajectories ${}^mD$, and then updating the estimate based on the difference in estimated values ${}^e\tilde{\mathbf{v}} = (\tilde{v}_1, \ldots, \tilde{v}_N)_D$ and

**Algorithm 1** MWAL (modified from [11])

```
1: Given ${}^e\mathbf{x}, {}^e\mathbf{u}, {}^e\mathbf{f}, {}^el_{i=1\cdots N}, D$
2: Estimate ${}^e\tilde{\mathbf{v}} = (\tilde{v}_1, \ldots, \tilde{v}_N)$ from expert trajectories $D$
   for all $i$. Normalise: ${}^e\hat{\mathbf{v}} = {}^e\tilde{\mathbf{v}}/\|{}^e\tilde{\mathbf{v}}\|$.
3: Let $\beta = \left(1 + \sqrt{\frac{2\log k}{M}}\right)^{-1}$.
4: Initialise ${}^m\hat{w}_i = \frac{1}{k}$ for all $i$
5: for $m = 1, \ldots, M$ do
6:     • Find trajectories ${}^mD$ that optimise $J = \sum_{i=1}^{N} {}^m\hat{w}_i {}^eJ_i$ under dynamics ${}^e\dot{\mathbf{x}} = {}^e\mathbf{f}({}^e\mathbf{x}, {}^e\mathbf{u})$
7:     • Estimate ${}^m\hat{\mathbf{v}}_i$ from trajectories ${}^mD$ for all $i$
8:     • Let ${}^{m+1}\hat{w}_i = {}^m\hat{w}_i\beta^{-\alpha({}^e\hat{v}_i - {}^m\hat{v}_i)}$
9:     • Re-normalise $\hat{w}$
10: end for
11: Return $\hat{w}$
```

${}^m\tilde{\mathbf{v}} = (\tilde{v}_1, \ldots, \tilde{v}_N)_{{}^mD}$. This proceeds until convergence to a set of weights that, when optimised, reproduces the demonstrated behaviour $D$. MWAL is summarised in Algorithm 1, and full details can be found in [11]). Please note that, for our implementation, we made two adjustments to the basic approach described there. These were (i) introduction of a learning rate parameter, $\alpha$ to adjust the speed of learning, and (ii) normalisation of the vectors ${}^e\hat{\mathbf{v}} = {}^e\tilde{\mathbf{v}}/\|{}^e\tilde{\mathbf{v}}\|$. and ${}^m\hat{\mathbf{v}} = {}^m\tilde{\mathbf{v}}/\|{}^m\tilde{\mathbf{v}}\|$. We found that the latter improved the robustness of learning especially for the high-dimensional, continuous systems considered in our experiments. Furthermore, for the forward optimisation step (Step 6 of Algorithm 1) we use the ILQG algorithm [12], details of which are described below.

### B. Task-based Behaviour Transfer

Having completed the AL stage to find a model of the demonstrator's objectives, our next task is to find an appropriate behaviour for the imitator. For this, we use local OFC to optimise an equivalent cost function to that used by the demonstrator. Specifically, we parametrise the learner's cost function as a similar weighted combination of terms

$$ {}^lJ = \sum_{i=1}^{n_T} \hat{w}_i {}^ll_i({}^l\mathbf{x}(T)) + \int_0^T \sum_{i=n_T}^{N} \hat{w}_i {}^ll_i({}^l\mathbf{x}, {}^l\mathbf{u}, t)\, dt. \quad (8) $$

Here, ${}^lh_i(\cdot), {}^ll_i(\cdot) \in \mathbb{R}$ are a set of basis functions that correspond to those of the expert (5), and $\hat{w}_i$ are the weights learnt by MWAL in the previous step. At this point a design decision must be made as to the appropriate correspondence between the learner's cost bases ${}^lh_i(\cdot), {}^ll_i(\cdot)$ and those of the expert ${}^eh_i(\cdot), {}^el_i(\cdot)$. In general, this will depend on the specific embodiments (dynamics and actuators) of the two plants. However, as noted in Sec. II-B in practical settings this is relatively easily resolved (and at worst, is no more difficult than specifying features ${}^e\psi(\cdot), {}^l\psi(\cdot)$ for feature-based imitation). For example, different terms might include work done by the two plants, or accuracy (e.g., in terms of the end-effector positions of the two plants). Further examples are given in the experiments (Sec. IV).

Having defined correspondence in terms of these bases, and given the learnt weights $\hat{\mathbf{w}}$, all that remains is to solve the optimal control problem defined by (8) and (2). Here, since we are interested in high-dimensional, continuous robot control problems, our method of choice is local OFC. In the next section we briefly describe the details.

### C. Local Optimal Feedback Control with ILQG

In our framework, solving the forward optimal control problem enters at two points. First, in the MWAL stage, the optimal trajectories ${}^mD$ with respect to the estimated cost function are sought at every iteration for updating the weights. Second, as discussed above, given the learnt cost function we seek the optimal movement for the imitator plant. In both cases we need a technique that (i) can cope with high-dimensional, non-linear systems and (ii) has high efficiency (since it is called multiple times during MWAL). For these reasons, our algorithm of choice is the iterative local quadratic Gaussian (ILQG) algorithm [12]. The latter is an efficient, approximate solver of optimal control problems, based on their local approximation as linear-quadratic-Gaussian and iterative improvement of solutions around a nominal trajectory.

Briefly, the ILQG algorithm starts with a time-discretised initial guess of a control sequence $\bar{\mathbf{u}}^j$ of length $T$. At each iteration $j$ this is used to find the corresponding state sequence $\bar{\mathbf{x}}^j$ under the deterministic forward dynamics $\mathbf{f}(\cdot)$ via Euler integration. Next, the dynamics are linearly approximated with a Taylor expansion, and, similarly, a quadratic approximation of the cost function around $\bar{\mathbf{x}}_t^j$ and $\bar{\mathbf{u}}_t^j$ is made. Both approximations are formulated as deviations $\delta\mathbf{x}_t^j = \mathbf{x}_t^j - \bar{\mathbf{x}}_t^j$ and $\delta\mathbf{u}_t^j = \mathbf{u}_t^j - \bar{\mathbf{u}}_t^j$ from the current trajectory and therefore form a 'local' LQG problem. The latter can be solved efficiently via a modified Ricatti-like set of equations.

With the solution to these equations, we find a correction to the control signal $\delta\bar{\mathbf{u}}^j$ which is used to improve the control sequence for the next iteration: $\bar{\mathbf{u}}^{j+1}(t) = \bar{\mathbf{u}}^j(t) + \delta\bar{\mathbf{u}}^j$. Finally, $\bar{\mathbf{u}}^{j+1}(t)$ is applied to the system dynamics and the new total cost along the trajectory is computed. The algorithm stops once the cost ceases to decrease significantly. After convergence, ILQG returns a control sequence $\bar{\mathbf{u}}$, gains $\bar{\mathbf{L}}$ and a state sequence $\bar{\mathbf{x}}$ which represents the optimal trajectory. In our framework, these trajectories are then either collected as sample data for Step 6 of the MWAL algorithm, or used for optimal control of the imitator plant, using the gains to provide local optimal feedback control.

## IV. Experiments

In this section, we evaluate our task-based imitation approach in three impedance control scenarios. In the first two experiments, we conduct simulation studies into behaviour transfer from (i) 1-link and (ii) 2-link systems with antagonistic actuation, to VIA systems with decoupled control of impedance. We then report experiments in learning from human demonstrations for behaviour transfer to the Edinburgh series elastic actuator (SEA) [9].

### A. Impedance Modulation on a Single Joint

In our first experiment, we investigate behaviour transfer from a 1-link system with an antagonistic VIA to another, similar system with simpler MACCEPA-like actuation [4]. The purpose of this experiment is to evaluate our approach

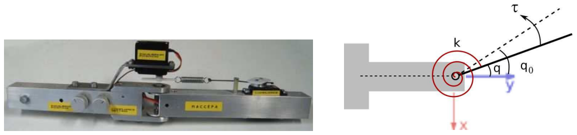


Fig. 3. MACCEPA VIA [4] and simplified dynamics model (image taken from http://mech.vub.ac.be/multibody/topics/maccepa.htm).

on a relatively small system where the ground truth is known, before scaling up to more complex problems.

As the antagonistic plant, in this experiment we used a simulation of the Edinburgh SEA (Fig. 8). While full details about this actuator can be found in [9], here we briefly discuss the salient features. Inspired by human antagonistic muscles, the Edinburgh SEA uses two motors, connected to a pair of springs to adjust equilibrium position and stiffness (and therefore torque) around the joint. The adjustments in stiffness are achieved by 'co-contraction', that is, simultaneous tensioning of the springs. Specifically, the joint is controlled by commanding target angles for the motors ${}^{e}\mathbf{u} = (\alpha, \beta) \in \mathbb{R}^2$ where $\alpha, \beta$ are the angles shown in Fig. 8(b). Under the assumption that the motors are infinitely stiff, the torque $\tau$ around the joint is given by

$$\tau(q, \mathbf{u}) = \hat{\mathbf{z}}^T((\mathbf{F}_2 - \mathbf{F}_1) \times \mathbf{a}) \tag{9}$$

where $\mathbf{a} = (a\cos q, a \sin q, 0)^T$, $\hat{\mathbf{z}}$ is the unit vector along the joint rotation axis and $\mathbf{F}_1, \mathbf{F}_2$ are the forces acting along the springs:

$$\mathbf{F}_1 = \kappa(s_1 - s_0)\frac{\mathbf{s}_1}{s_1} \quad \text{and} \quad \mathbf{F}_2 = \kappa(s_2 - s_0)\frac{\mathbf{s}_2}{s_2} \tag{10}$$

(ref. Fig. 8(b)). Here, $s_0$ is the rest length of the springs, $\kappa$ is the spring constant, $\mathbf{s}_1 = \mathbf{s}_1(\alpha, q)$ and $\mathbf{s}_2 = \mathbf{s}_2(\beta, q)$ are the vectors CA, and DB respectively (see Fig. 8(b)), and $s_1$ and $s_2$ their respective lengths. Note that, there is a non-linear relationship between the latter and the commanded servomotor positions (see [9] for details). Finally, we represent the state of the joint as ${}^{e}\mathbf{x} = (q, \dot{q}) \in \mathbb{R}^2$, i.e., the instantaneous joint angle and velocity.

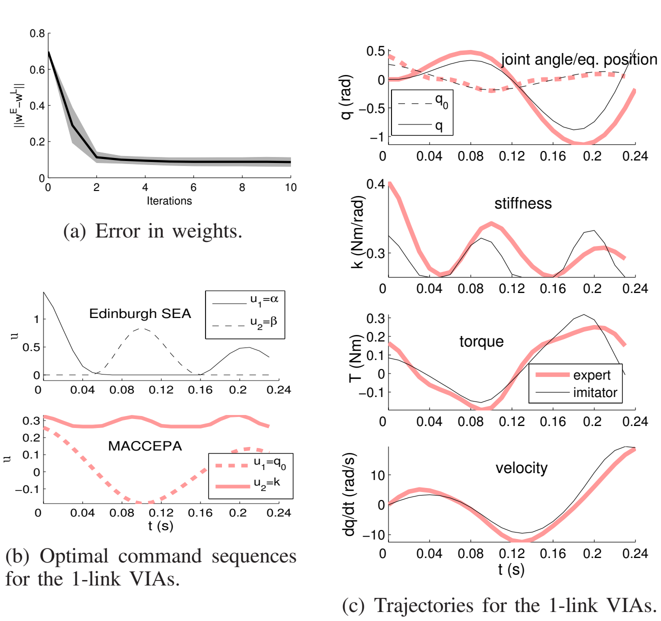
To generate examples of optimal behaviour for this plant, ILQG was used to plan trajectories for a 'ball hitting' task. Specifically, a set of trajectories minimising the objective

$${}^{e}J = w_1(q(T) - q^*)^2 - w_2\dot{q}(T) + \int_0^T w_3\tau^2\,dt \tag{11}$$

were planned and executed, where $q^* = 30°$ is the target angle and $\tau$ is the torque applied around the joint. The weighting of the three terms of (11) respectively correspond to (i) minimising the distance to the target (ball) at the time of impact $T$, (ii) maximising the angular velocity at $T$, and (iii) minimising energy consumption during the movement. The trade-off between these objectives is determined by the weights $w_i$.

We collected $K = 30$ such trajectories from random start states as training data, and used MWAL to estimate the weights $\hat{\mathbf{w}}$. To assess learning performance, we repeated this on 50 such data sets and measured the error in the estimate during learning. The latter was measured by the $l_2$-norm difference in the true and estimated weights, i.e.

$$E_{\mathbf{w}}[\mathbf{w}, \hat{\mathbf{w}}] = \|\mathbf{w} - \hat{\mathbf{w}}\|_2. \tag{12}$$



(a) Error in weights.

(b) Optimal command sequences for the 1-link VIAs.

(c) Trajectories for the 1-link VIAs.

Fig. 4. Results for the 1-link experiment. Shown are (a) error in weights against MWAL iteration $m$ (mean±s.d. over 50 trials), (b) optimal command sequence $\mathbf{u}$ for the Edinburgh SEA (thin black) and the MACCEPA-like joint (thick red), (c) time profile of the actual (solid) and equilibrium position (dashed lines), stiffness, resultant torque and velocity over time for the two plants.

Fig. 4(a) shows the weight error $E_{\mathbf{w}}$ over 10 iterations of the MWAL algorithm with a relatively high learning rate ($\alpha = 50$). As can be seen, there was rapid convergence to a low error, with final error $0.0941 \pm 0.0247$.

We then investigated the transfer of this behaviour to a second, similar plant, but a different actuation. For this, we selected an actuator where the stiffness and equilibrium position can be directly controlled, similar to the MACCEPA joint [4] (see Fig. 3). More specifically, the second plant had command vector ${}^{l}\mathbf{u} = (q_0, k)^T \in \mathbb{R}^2$ (where $q_0$ denotes equilibrium position and $k$ the stiffness) in order to control the applied torque

$$\tau(q, \mathbf{u}) = -k(q_0 - q). \tag{13}$$

For ease of comparison, all other dynamics parameters (e.g. link length, inertia etc.) were kept identical to those of the first plant. Note that, for the two plants under consideration, correspondence in the first two terms of (11) is exact (since the joints dynamics are identical), but there is a difference in the functional form of the third term, due to the different relationships between $\mathbf{u}$ and $\tau$. Using the weights learnt with MWAL, we applied ILQG to find optimal movements for this plant and compared the results (see Fig. 4).

The first thing that we notice is that at the level of the commands (Fig. 4(b)), very different strategies appear to be optimal for the two plants. This reinforces the fact that, considering the differences in actuation, *direct imitation* here is inappropriate. However, looking at Fig. 4(c), we see that at the behavioural level, there is similarity in several features of the movement (e.g. the strategy of swinging the equilibrium position away from the current actual position to build up energy in the system - see Fig. 4(c), top). The correspondence is not exact due to the plants' mechanical

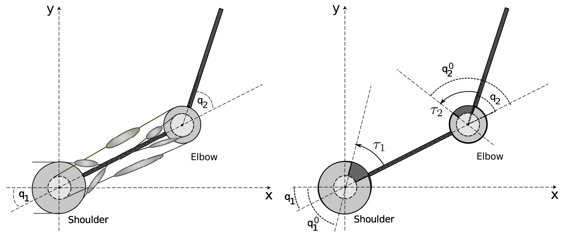


Fig. 5. Left: Two-link, six-muscle human arm model; Right: rc manipulator with active stiffness control.

differences (e.g., coupling in the Edinburgh SEA prevents some stiffness values being reached for a given joint angle) but there is clear qualitative similarity.

To quantitatively assess this, we evaluated the cost according to the true weights $w$ for $K = 30$ trajectories (i) from the expert (ii) planned by our task-based imitation approach and (iii) generated by feeding the equilibrium position and stiffness profiles of the expert trajectories directly as command sequences for the imitator (i.e., a naive, feature-based imitation approach where we set ${}^l\mathbf{u}(t) = ({}^e q_0(t), {}^e k(t))$ for each trajectory). As a result we found that the average cost of the expert's trajectories was $-1.337 \pm 0.058$, compared to $-1.507 \pm 0.133$ for the naive approach and $-2.508 \pm 0.787$ with our AL approach. By taking the proposed approach then, we can potentially find trajectories that far surpass those of the naive approach in terms of task performance measured by the expert's objective function.

### *B. Impedance Modulation on a Two-joint Human Arm Model*

To test scalability, in our second experiment we assess our method for transferring behaviour between more complex systems with much higher dimensionality. For this, we aim to transfer control strategies from a biologically realistic simulation of the human arm [7] to that of a 2-link robotic manipulator with active stiffness control, as shown in Fig. 5.

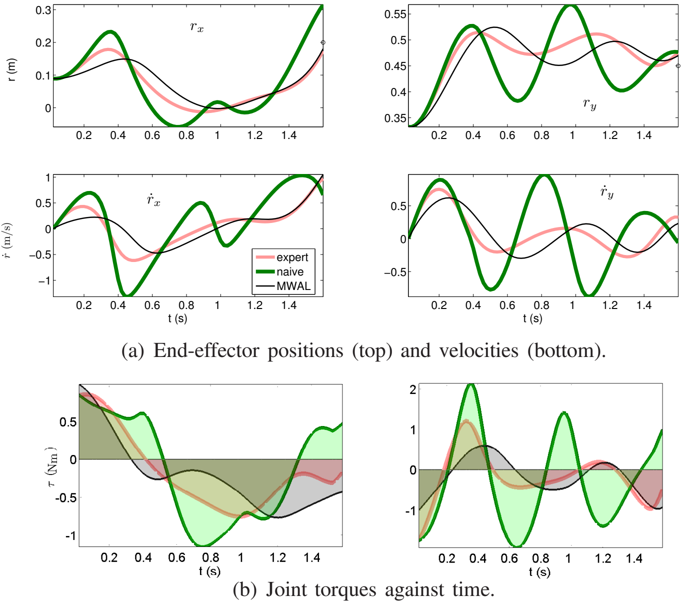
While details of the human arm simulation are described in [7], we briefly discuss the salient properties. The human arm is modelled as a two-joint planar rigid body system, actuated by two pairs of monarticular and one pair of biarticular antagonistic muscles. The dynamic parameters of the arm are based on human measurements, including values for muscle stiffness and viscosity. The arm is controlled by specifying muscle activations, i.e., ${}^e\mathbf{u} \in \mathbb{R}^6$, and its state is represented as ${}^e\mathbf{x} = (\mathbf{q}, \dot{\mathbf{q}})^T \in \mathbb{R}^4$ where $\mathbf{q} \in \mathbb{R}^2$ and $\dot{\mathbf{q}} \in \mathbb{R}^2$ denote joint angular position and velocities. For a given muscle activation $\mathbf{u}$ the applied torques are given by

$$\tau(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}) = -\mathbf{A}^T \mathbf{T}(\mathbf{l}, \dot{\mathbf{l}}, \mathbf{u}), \tag{14}$$

where $\mathbf{A}$ is the moment arm, and muscle lengths and velocities follow the affine relationship $\mathbf{l} = \mathbf{l}_m - \mathbf{A}\mathbf{q}$ and $\dot{\mathbf{l}} = -\mathbf{A}\dot{\mathbf{q}}$. The muscle tension

$$\mathbf{T}(\mathbf{l}, \dot{\mathbf{l}}, \mathbf{u}) = \mathbf{K}(\mathbf{u})\big(\mathbf{l}_r(\mathbf{u}) - \mathbf{l}\big) - \mathbf{B}(\mathbf{u})\dot{\mathbf{l}}. \tag{15}$$

depends on the muscle stiffness $\mathbf{K}(\mathbf{u}) = diag(\mathbf{k_0} + g_k\mathbf{u})$, viscosity $\mathbf{B}(\mathbf{u}) = diag(\mathbf{b_0} + g_b\mathbf{u})$ and rest lengths $\mathbf{l}_r(\mathbf{u}) = \mathbf{l_0} + g_r\mathbf{u}$. The elasticity coefficient $g_k$, the viscosity coefficient $g_b$, and the constant $g_r$ are given from the muscle model. The same holds true for $\mathbf{k_0}$, $\mathbf{b_0}$, and $\mathbf{l_0}$, which are the intrinsic elasticity, viscosity and rest length for $\mathbf{u} = \mathbf{0}$, respectively.



(a) End-effector positions (top) and velocities (bottom).

(b) Joint torques against time.

Fig. 6. Example trajectory for the 'punching' task for the two joint plant when actuated with antagonistic muscles (light red), and direct stiffness control with the MWAL (thin black) and naive (thick green) controllers.

For ease of comparison with the 1-link analysis, in this experiment we chose to investigate behaviour transfer for the similar, but more complex task of 'punching'. For this, as a ground truth, trajectories were collected from the demonstrator that minimised the objective

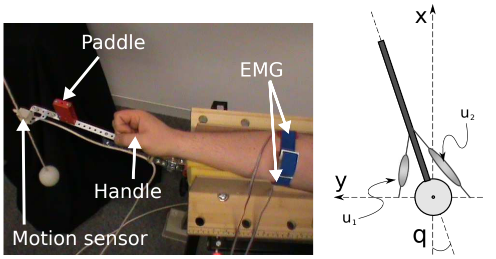
$${}^eJ = w_1\|\mathbf{r}(T) - \mathbf{r}^*\|_2^2 - w_2\dot{r}_x(T) + \int_0^T w_3\|\boldsymbol{\tau}\|_2^2\,dt \tag{16}$$

where $\mathbf{r} = (r_x, r_y)^T \in \mathbb{R}^2$ is end-effector position, $\mathbf{r}^* = (.2, .45)^T m \in \mathbb{R}^2$ is the position of a target in Cartesian space and $\dot{r}_x$ is the end-effector velocity in the $x$ (left lateral) direction. The three terms of (16) respectively correspond to (i) minimising the distance of the end-effector to the punching target at the time of impact $T$ (i.e., accuracy), (ii) maximising the velocity of the end-effector at impact, and (iii) minimising energy consumption during the movement. The trade-off between these objectives is determined by the weights $w_1$, $w_2$, and $w_3$. $K = 10$ such trajectories from random initial joint configurations were collected under the arm dynamics. These were then used as training data for MWAL to estimate the weights $\hat{\mathbf{w}}$.

We then transferred the behaviour to a simulated robot with identical kinematics and dynamics, but with different actuation. Specifically, the robot used active stiffness control where the command vector is ${}^l\mathbf{u} = (\mathbf{q}_0, \mathbf{k})^T \in \mathbb{R}^6$, where $\mathbf{q}_0 \in \mathbb{R}^2$ corresponds to the equilibrium position of the two joints and $\mathbf{k} = (K_{11}, K_{12}, K_{21}, K_{22})^T \in \mathbb{R}^4$ where $K_{ij}$ denote the $i, j$th elements of the joint stiffness matrix $\mathbf{K} \in \mathbb{R}^{2\times 2}$ and the applied torques goes as

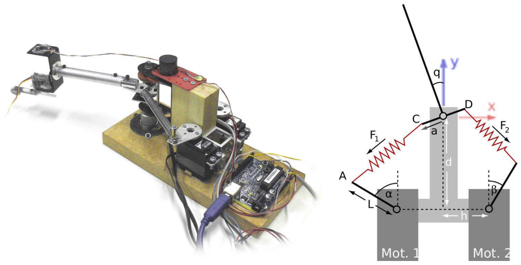
$$\boldsymbol{\tau} = -\mathbf{K}(\mathbf{q}_0 - \mathbf{q}) - \mathbf{B}\dot{\mathbf{q}} \tag{17}$$

where $\mathbf{B} = .06\mathbf{I}$ is a fixed damping matrix. As before, we assessed performance (i) in terms of the accuracy with



(a) Measurement rig. (b) Wrist model.

Fig. 7. Apparatus for recording human demonstrations of the hitting task (left) and forward dynamics model of the human wrist (right).



(a) Edinburgh SEA (b) Dynamics model.

Fig. 8. Edinburgh SEA hardware and rigid body dynamics model.

which the cost function was learnt, and (ii) in terms of task performance as measured by the expert's objective function. We evaluated this for 20 trials on different data sets.

Our results are as follows. First, looking at the learnt weights, we found that the average error attained by MWAL was $0.1682 \pm 0.0151$. Considering the increased complexity and higher dimensionality of the learning problem, we regard this as good performance. Second, evaluating the task performance, we found that over $K = 10$ trajectories, the average cost of the expert's trajectories was $-0.186 \pm 0.035$, that of the AL approach was $-0.296 \pm 0.013$ and that of the naive, feature-based approach (i.e., directly feeding the expert's stiffness and equilibrium positions as robot commands ${}^{l}\mathbf{u}(t)$) was $0.122 \pm 0.121$. The reason for the poor performance of the latter can be seen when plotting out an example trajectory. In Fig. 6 we show the end-effector positions and velocities (Fig. 6(a)) and joint torques (Fig. 6(b)). We see that due to the lower, fixed damping of the robotic plant, the naive feature-based imitation strategy produces highly unstable trajectories, with high cost in terms of the integrated torque (shaded area). On the other hand, by planning appropriate movements for the robot using the AL approach we get smooth trajectories that closely match those of the expert.

### C. Learning from Human Data

In our final experiment, we applied our approach to learning from a set of human demonstrations with the goal of transferring behaviour to the Edinburgh SEA (Fig. 8). For ease of comparison with the simulation studies, we again chose to study a task similar to that described in Sec. IV-A, whereby the demonstrator attempts to hit a target (ball) as hard as possible while minimising the energy consumed. Our goal is to learn a model of the human's objective function in order to transfer it to the robotic hardware. The experimental setup is as follows.

For collecting demonstrations, the measurement rig shown in Fig. 7(a) is used. The rig consists of a hinge joint with a paddle attached, that is aligned to a ball suspended from a string. The rig has a handle which the demonstrator grasps to rotate the joint and hit the ball with the paddle. A magnetic motion sensor (Flock of Birds, Ascension Tech. Corp.) is used to measure the angle of the demonstrator's wrist (corresponding to the hinge angle) at a 500Hz sampling rate. Simultaneously, a pair of surface EMG sensors, placed on the antagonistic muscles of the demonstrator's forearm measure the muscle activations of the demonstrator at the same 500Hz rate. With this setup, we are able to measure trajectories of the human through state (modelled as ${}^{e}\mathbf{x} = (q, \dot{q}) \in \mathbb{R}^2$, the instantaneous wrist angle and velocity) and action space (modelled as the muscle activations ${}^{e}\mathbf{u} = (a_1, a_2) \in \mathbb{R}^2$, measured via EMG).

Using this setup, data was collected from a human attempting to hit the ball (suspended at a point corresponding to wrist angle $q^* = 34.0°$) as hard as possible with the paddle, from a series of start positions, given a fixed time duration in which to complete the movement. Specifically, 3 trajectories were recorded from each of 5 start positions $q = \{10, 0, -10, -20, -30\}°$, with a fixed duration of $0.2\,s$. To reduce the effects of noise and variability in the execution of the trajectories, the data was preprocessed by (i) smoothing the signals with a Butterworth filter and (ii) temporal alignment of trajectories around the time of impact with the ball. The trajectories from each of the start states were then averaged, and the resultant $K = 5$ mean trajectories were then used as training data for the learning.

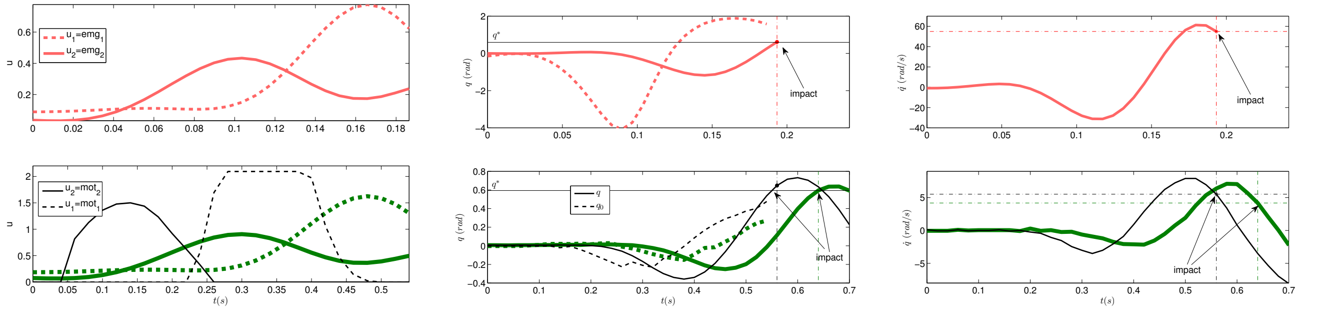
Since the MWAL algorithm requires a model of the expert's forward dynamics, the human wrist dynamics must be approximated. For this we used a simplified two-muscle, single joint model (Fig. 7(b)), with the same hill-like muscle dynamics as described in the preceding section. This yielded a forward dynamics model of the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) = (\,\dot{q},\, \tau(q, \dot{q}, \mathbf{u})/I\,)^T \tag{18}$$

where $\tau(q, \dot{q}, \mathbf{u})$ is the applied torque (as calculated from (14)-(15) for the two-muscle model) and $I$ is the estimated inertia. In order to find the best possible fit to the dynamics of our demonstrator, the parameters (i.e., $I, \mathbf{A}, l_m, l_0, k_0, b_0, g_k, g_b, g_r$) of this model were optimised with respect to the normalised error between the recorded trajectories $D = \{({}^{e}\mathbf{x}_0^k, {}^{e}\mathbf{u}_0^k), \cdots, ({}^{e}\mathbf{x}_T^k, {}^{e}\mathbf{u}_T^k)\}_{k=0}^{K}$ and those predicted by integrating the model under the same command sequence $\hat{D} = \{({}^{e}\hat{\mathbf{x}}_0{}^k, {}^{e}\mathbf{u}_0^k), \cdots, ({}^{e}\hat{\mathbf{x}}_T{}^k, {}^{e}\mathbf{u}_T^k)\}_{k=0}^{K}$.

For estimating the human objective, we again modelled the cost function in the form (11), and sought the best fit to the weighting coefficients $w_1, w_2, w_3$ with MWAL. Note that, in this experiment, as $\tau$ cannot be directly measured during movement, we used the optimised parametric model (18) to estimate the torques for the third term. We trained the model on the $K = 5$ training trajectories, with a high learning rate of $\alpha = 300$ for 20 iterations. Note that, in this experiment, since the true human cost function is unknown we cannot explicitly calculate the error (12). Instead, convergence was measured by examining the magnitude of the weight update

Fig. 9. Comparison of ball-hitting behaviour of (i) the human demonstrator (top row, light red) against (ii) the robot with direct imitation of the command sequence (bottom row, thick green) and (iii) the robot executing the optimal trajectory with respect to the learnt objective function (bottom row, thin black).



(a) Command profiles: Human EMG (top) and desired robot motor angles (bottom) are shown.

(b) Joint-angular trajectory (solid lines). Estimated equilibrium angle (dashed) and target position $q^*$ (thin line) are also shown.

(c) Joint velocity profile. Dashed lines indicate impact time and velocity.

(i.e., step 8 in Algorithm 1).

Finally, to evaluate our approach, we used ILQG to find the optimal controller for the Edinburgh SEA with respect to the cost function (11) using the learnt weights. Specifically, we compared the behaviour of the robot (i) when controlled with the local OFC controller found by ILQG under the learnt cost function and, (ii) the direct imitation approach (whereby the human EMG signal is directly fed as commands to the robot) against the human behaviour. Note that, for the direct approach, the (normalised) EMG data was scaled to ensure that the maximum recorded EMG signal (over the entire data set) corresponded to the maximum admissible angle of the robot motors. Note also that, since the response of the robot's servomotors is significantly lower than than that of the human (in terms of control frequency and other delays), control of the robot was scaled in time so that the command sequence had $0.5\,s$ duration for both of the approaches compared.

The results are shown in Fig. 9 for an example trajectory starting at $q = 0°$). Looking at the joint angle and velocity profiles (Fig. 9(b)-(c)), we can see that the strategy used by the human is to first move the wrist away from the target before rapidly moving it in the positive direction toward the target. A similar movement occurs on the robot when using both the direct and the AL approaches. However, comparing these, we see that for the direct approach, the amplitude of the movement is smaller and the velocity at the time of impact is much smaller. In contrast, the proposed approach optimises the command sequence for the robot dynamics, resulting in earlier onset time for the movement, and a much larger movement of the motors (see Fig. 10(a)). This allows it to achieves a higher hitting velocity (with the ball travelling a greater distance) when executed on the robotic hardware.

## V. Conclusion

In conclusion, we have presented a task-based imitation learning approach for transfer of behaviour across plants with highly heterogeneous dynamics and actuation. Our framework is based on a two-step approach to learning, where in the first step, a parametric model of the objective function underlying observed behaviour is learnt using an apprenticeship learning approach. This enables us to find a task-based representation of the data in terms of the objectives minimised. Using this model of the behaviour, and solving the correspondence problem in terms of the the components of the objective function, we then apply local optimal feedback control techniques to find a similarly optimal behaviour for the imitation, taking into account the differences in actuation. Our experiments show the effectiveness of this approach, where the proposed approach actually exploits the dynamics characteristics of the imitator in order to out-perform standard feature-based imitation approaches, and even surpass the task-performance of the expert.

In future work we intend to build on our results and apply our approach to a number of different impedance control tasks, and achieve task-based imitation on a range of variable impedance actuator designs.

## VI. Acknowledgements

This work was funded by the EU Seventh Framework Programme (FP7) as part of the STIFF project.

## References

[1] P. Abbeel and A. Ng. Apprenticeship learning via inverse reinforcement learning. In *ICML*, 2004.

[2] A. Alissandrakis, C. Nehaniv, and K. Dautenhahn. Correspondence mapping induced state and action metrics for robotic imitation. *IEEE Trans. Sys., Man, Cybernetics*, 37:299–307, 2007.

[3] D. Franklin, G. Liaw, T. Milner, R. Osu, E. Burdet, and M. Kawato. Endpoint stiffness of the arm is directionally tuned to instability in the environment. *J. Neuroscience*, 27:7705–7716, 2007.

[4] R. Van Ham, B. Vanderborght, M. Van Damme, B. Verrelst, and D. Lefeber. Maccepa, the mechanically adjustable compliance and controllable equilibrium position actuator: Design and implementation in a biped robot. *Robotics and Auton. Systems*, 55:761–768, 2007.

[5] C. Harris and D. Wolpert. Signal-dependent noise determines motor planning. *Nature*, 394:780–784, 1998.

[6] N. Hogan. Impedance control - an approach to manipulation. part iii - applications. *J. Dynamic Sys., Meas. and Control*, 107:1–24, 1985.

[7] M. Katayama and M. Kawato. Virtual trajectory and stiffness ellipse during multijoint arm movement predicted by neural inverse models. *Biol. Cybern.*, 69:353–362, 1993.

[8] G. K. Klute, J. M. Czerniecki, and B. Hannaford. Mckibben artificial muscles: Pneumatic actuators with biomechanical intelligence. In *IEEE Int. Conf. Advanced Intelligent Mechatronics*, 1999.

[9] D. Mitrovic, S. Klanke, and S. Vijayakumar. Exploiting sensorimotor stochasticity for learning control of variable impedance actuators. Tech. Report: www.inf.ed.ac.uk/publications/report/1359.html, 2009.

[10] R. Schiavi, G. Grioli, S. Sen, and A. Bicchi. Vsa-ii: a novel prototype of variable stiffness actuator for safe and performing robots interacting with humans. In *ICRA*, 2008.

[11] U. Syed, M. Bowling, and R. E. Schapire. Apprenticeship learning using linear programming. In *ICML*, 2008.

[12] E. Todorov and W. Li. A generalized iterative lqg method for locally-optimal feedback control of constrained nonlinear stochastic systems. In *American Control Conf.*, 2005.