Perturbation-Based Stiffness Inference in Variable Impedance Control

Edoardo Caldarelli, Adrià Colomé, and Carme Torras

Abstract—One of the major challenges in learning from demonstration is to teach the robot a wider set of task features than the plain trajectories to be followed. In this sense, one key parameter is stiffness, i.e., the rigidity that the manipulator should exhibit when performing a task. The required robot stiffness is often not known *a priori* and varies along the execution of the task, thus its profile needs to be inferred from the demonstrations. In this work, we propose a novel, force-based algorithm for inferring time-varying stiffness profiles, leveraging the relationship between stiffness and tracking error, and involving human-robot interaction. We begin by gathering a set of demonstrations with kinesthetic teaching. Then, the robot executes a perturbed reference, obtained from these demonstrations by means of Gaussian process regression, and the human intervenes if the perturbation makes the manipulator deviate from its expected behaviour. Human intervention is measured and used to infer the desired control stiffness. In the experiments section, we show that our algorithm can be combined with different types of force sensors, and provide suitable processing algorithms. Our approach correctly infers the stiffness profiles from the force and electromyography sensors, their combination permitting also to comply with the physical constraints imposed by the environment. This is demonstrated in three experiments of increasing complexity: a motion in free Cartesian space, a rigid assembly task, and bed-making.

Index Terms—Compliance and impedance control, learning from demonstration, probabilistic inference.

I. INTRODUCTION

RECENTLY, robotic manipulation has been challenged by increasingly complex applications that involve interaction with humans outside standard industrial scenarios. In such cases, human experts need to transfer skills to robots in a flexible, yet accurate manner. One popular way of teaching these skills is by means of *learning from demonstration* (LfD) [1]. In LfD, the tasks are implicitly encoded within a set of multiple repetitions, executed by a human. These demonstrations can be therefore processed with learning algorithms, and reproduced by the robot. One effective way of achieving LfD is by means of kinesthetic teaching [2]. In this method, the human directly moves the robot, so as to make it perform the desired task correctly, while recording the data to be processed.
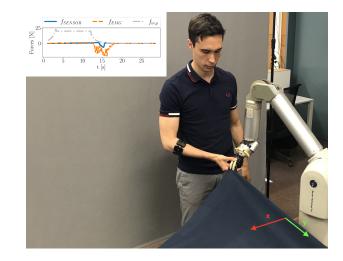
Fig. 1. The main human-robot interaction phase involved in our learning algorithm. The robot follows a reference subject to a perturbing force in x (the grey curve in the plot), while the human corrects its trajectory if the task specification is violated. The force measurements along the perturbation's axis, obtained with a force sensor and with an elecromyography bracelet, are shown in blue resp. orange in the plot.

One key aspect of LfD is the algorithm that is chosen to process the available human demonstrations, especially the trajectories to be followed by the robot. Among the most popular approaches, we can consider *probabilistic movement primitives* (ProMPs) [3]. While largely used, these models are *parametric*, and require an explicit tuning of the model's parameters to fit the trajectory. To alleviate the burden of parameter tuning, other models can be used, such as *Gaussian processes* (GPs) [4], which offer a flexible, *non-parametric* model to be used to encode the robot's trajectories [5].

In addition to the choice of the trajectories' processing algorithm, LfD poses the challenge of identifying which information, besides the trajectory, should be taught to the robot. In particular, one parameter that is often involved in LfD is the *stiffness*, or impedance, that the robot should exhibit while performing a task. A complex skill may require a time-varying rigidity in the arm, that adapts to the interaction with the environment while fulfilling the other requirements in the task specification (i.e., precision). This leads to the so-called *variable impedance control* (VIC).

A relevant research question in VIC is how to retrieve a stiffness specification from the human demonstrations. A well-established way of obtaining these profiles is by means of the *variance* of the human demonstrations [6]–[9]. In principle, a high variance should correspond to a low stiffness, and vice versa. However, as extensively discussed in [10], a variance-based approach may be sub-optimal,

or even harmful, in presence of environmental constraints. In this case, a low stiffness is desirable, even though the precision is high thanks to the constraints. Building on these observations, [10] proposes to derive the stiffness directly from the forces of the human controlling the learning process, estimated by means of an electromyography (EMG) signal. This approach introduces the fundamental concept of force-based stiffness inference. However, directly inferring the stiffness from force measurements may be in turn suboptimal, e.g., when muscle contraction is due to the weight of an object being moved by the human demonstrator. In this case, the weight could be easily compensated by the manipulator's control algorithm, without the need of changing the stiffness values. A complementary view on VIC proposes to adapt the stiffness to the contact forces between the robot and the environment [11]. In this case, the adaptation of the stiffness is learnt by means of a continuous dynamic interaction between the robot and the environment, both during the training phase and during task execution. As discussed in [11], this might be suboptimal in cases where the robot has small or null dynamic interaction with the environment, e.g., in quasi-static tasks.

In this work, we propose a novel, force-based method for stiffness inference in VIC. In particular, we achieve this by leveraging the relationship between stiffness, perturbations and *tracking error*, firstly explored in [12], [13]. After an initial phase of kinesthetic teaching, the manipulator executes multiple perturbed motions, and the human is required to correct such motions if the task specification is being violated, as shown in Fig. 1. The measured interaction forces, along with the perturbations, are used for the stiffness' inference. In contrast to [12], our perturbations are generated automatically with an optimization-based algorithm, instead of pre-defined motions of the human supervisor. In this way, full control on the perturbations' profile is achieved. Moreover, the human supervisor is only required to correct the trajectory according to their *expert knowledge* of the task being considered, rather than the actual learning algorithm used. Once the stiffness information is collected in the form of symmetric positive-definite matrices, further adaptation of the stiffness profile can be performed, as discussed, e.g., in [14], where such matrices are assumed to be known.

With our learning pipeline, the robot has full knowledge about the stiffness profiles at each time instant, without the need of further external stimuli. Moreover, our algorithm performs a decoupling of the different degrees of freedom (DOFs) considered. This choice allows to tune the stiffness in all DOFs, or in a chosen subset only, and does not require prior knowledge about which DOFs need to be compliant or stiff. Furthermore, along with the GP-based encoding of the trajectories, it reduces the number of parameters to be inferred from data. Lastly, we test our algorithm with different force sensory equipment, providing suitable processing algorithms for each one of them.

This paper is structured as follows. In Sec. II, we list the main concepts at the basis of our method. In Sec. III, we present our novel contribution. The experiments are shown in Sec. IV, while the conclusions are reported in Sec. V.
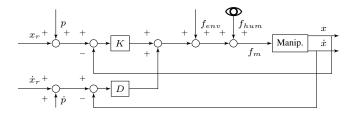

Fig. 2. The block scheme of our human-robot interaction phase, for a certain DOF. The human is compensating the perturbations p by watching the arm reproduce the demonstrated trajectory, with an additive perturbation.

II. PRELIMINARIES

Here, we list the main preliminaries to our work.

1) GP-Based Encoding of Trajectories: Let us consider a 1-dimensional time series, $x(t)$. A GP [4] specifies a prior distribution over the function $x(\cdot)$, depending on a mean function $\mu(\cdot)$ and a kernel $k(\cdot, \cdot)$. In particular, any finite subset of the values of $x(\cdot)$, which we denote $x(\mathbf{t})$, is a Gaussian random vector whose mean and covariance are given by $\mu(\mathbf{t})$ and $k(\mathbf{t}, \mathbf{t})$. While the mean function is commonly assumed to be 0, the kernel function is parametrized with a set of hyperparameters ϕ, inferred, e.g., by maximum-likelihood estimate (MLE). We can assume that we have access to noisy measurements of $x(\mathbf{t})$, obtained with a zero-mean Gaussian noise with constant variance. From this, we obtain a posterior distribution of the time series at some test inputs, given the noisy observations. In an LfD scenario, the trajectory to be encoded consists of multiple degrees of freedom (DOFs). In this case, assuming that all DOFs are fully observed, each of them can be processed by fitting an independent GP [5]. In a realistic scenario, the noise corrupting the function's values is time-dependent. Thus, a second GP prior can be used to model the noise variance. The combination of the GP on the mean of the observations, and the GP on the noise variance forms the well-known *heteroscedastic GP* (HGP). These chained GPs lead to intractable integrals, and approximate inference needs to be done in order to fit the HGP to the data. Here, we use the expectation-maximization (EM) based approach described in [15]. Usually, multiple demonstrations of the same task are performed. Here, we pre-process all the demonstrations of a task by aligning them with dynamic time warping (DTW) [16], and fit an HGP to the union of the demonstrations.

2) Impedance Control of Manipulators: Several control schemes can be used to control the manipulator executing the task. In particular, we consider control in Cartesian space, as it is particularly suitable for the LfD scenario [17]. One popular control scheme leverages the well-known relationship between the generalized forces applied to the end-effector (EE) of a manipulator ($\mathbf{f}_m$), and the resulting joint torques ($\boldsymbol{\tau}$). Let $\mathbf{J}$ be the Jacobian matrix of the robot. We can use an impedance control scheme, to obtain the following control law, for stiffness and damping matrices $\mathbf{K}$ and $\mathbf{D}$, reference $\mathbf{x}_r$ and measured EE position and velocity $\mathbf{x}$, $\dot{\mathbf{x}}$:

$$\boldsymbol{\tau} = \mathbf{J}^T[\mathbf{K}(\mathbf{x}_r - \mathbf{x}) + \mathbf{D}(\dot{\mathbf{x}}_r - \dot{\mathbf{x}})]. \qquad (1)$$

If $\mathbf{K}$ and $\mathbf{D}$ are diagonal, the manipulator is controlled by means of *diagonal control*. This approach assumes no coupling
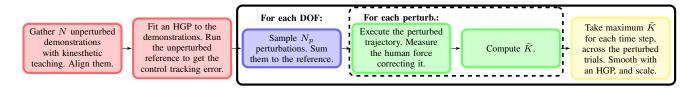
Fig. 3. The flow chart describing the steps of our algorithm for stiffness inference. In the chart, we can identify the two main phases that compose our pipeline. In red, we report the standard kinesthetic teaching steps. Then, the blocks within the solid black contour represent the interactive learning phase, where the human-robot interaction forces are measured. The forces can be measured with different sensors (force sensor, EMG bracelet).

between the DOFs of the manipulator [10].

3) Bayesian Optimization: Bayesian optimization (BO) is a gradient-free optimization technique that can be used to find the extrema of non-convex functions. In this work, we use BO with expected improvement, as presented in [18].

III. PROPOSED METHOD

In this section, we present our pipeline for learning the time-varying stiffness profiles of a robotic manipulator from human demonstrations. We begin by recording multiple demonstrations of a task, from which we obtain a reference. Then, as a second phase of the algorithm, the robot executes a perturbed version of the reference. A human supervises the motion and counteracts the perturbations, if the task specification is violated. The interaction force is measured and used to infer the stiffness. As we will discuss in Sec. IV, several options exist to measure this force. In particular, we consider using a force sensor on the robot's EE, and an EMG bracelet on the supervisor's forearm. If the robot is moving in free Cartesian space, then the two sensors are equivalent. However, the EMG signal needs a heavier processing, as we shall discuss in Sec. IV. Thus, if available, the force sensor is preferable. If the robot's motion is constrained by the environment, we must combine the two sensors, to differentiate between the environment's and the human forces. Note that, in the following, we consider one DOF at a time, i.e., the quantities involved are scalars, in line with diagonal control. Thus, we drop the bold notation used in Eq. (1) for vectors and matrices. Furthermore, we assume that a minimal representation of the orientation is used (e.g., with Euler angles). If this is not the case, we can refer to the processing described in appendix. To lighten the notation, in the following parts, the DOF index which all quantities depend on is omitted. The forces shown are *generalized* forces, i.e., forces for the position, and torques for the orientation DOFs.

A. Stiffness and Perturbations

As a start, let us consider the impedance control scheme presented in Sec. II. In such a control scheme, the stiffness relates the *error* in the EE's pose, caused by a perturbation along a certain axis, and the *force* exerted on the EE, along the same axis of the error. Thus, if we want to measure the stiffness of a human demonstration, we might perturb it with a known disturbance, and measure the human force reacting to it. However, applying such perturbations with humans only is not straightforward, since it might prevent us from having full control on the perturbation model. Thus, we can adopt a

framework in which both a human demonstrator and a robot are involved, and interact, as we shall see in the following.

1) Unperturbed Trajectories: Our algorithm begins by recording a set of trajectories for a given task, by means of kinesthetic teaching. This constitutes a set of *unperturbed* samples. Such samples are processed by means of HGP regression as described in Sec. II, to retrieve a reference (the posterior mean of the GP) and the posterior variance.

2) Introducing Perturbations: Once the unperturbed trajectories have been collected and processed, the robot executes a trajectory where the reference is spoiled by an additive perturbation. Naturally, the robot would follow the spoiled path and thus would move away from the true desired trajectory. A human supervising the execution corrects the trajectory by dragging the robot's EE, if the task specification is being violated[1]. Note that, if the task is already constrained by the *environment*, the human does not need to interact with the robot. Once the interaction force is measured, a newly perturbed trajectory is executed.

3) Perturbations and Variance: Intuitively, the perturbations should affect different parts of the trajectory, to fully explore the stiffness profile. However, we do not need to perturb where the posterior variance of the unperturbed trajectories is large. Intuitively, a large variance means that the human demonstrator was not caring about precision in this part of the trajectory, fully exploring the space of available positions of the EE. On the other hand, a small variance at a certain part of the trajectory might correspond to three different scenarios:

- a disturbance does not affect that portion of the trajectory because the stiffness is high;
- a disturbance does not affect that portion of the trajectory thanks to environmental constraints;
- that portion of the trajectory has not been perturbed yet, so we cannot know whether the stiffness needs to be high or low.

4) Knowledge Function: It is clear from the observations above that the variance and the disturbances contribute to our knowledge of the stiffness profiles. In particular, we should place the new disturbance at a point in time where we have a small variance and we have not registered a relevant perturbation yet. We can therefore introduce the notion of *knowledge function*, that will be the key of our perturbation algorithm.

Definition III.1. Let I_p be the number of perturbed trajectories already executed, and $\sigma(t)$ be the posterior standard deviation

---

[1]The stiffness of the robot in this phase should be small enough to allow the human to correct the robot's trajectory. In turn, the tracking needs to be performed slowly.

(a) Reference signal (GPs)
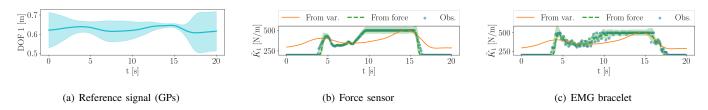
(b) Force sensor

(c) EMG bracelet

Fig. 4. Fig. 4(a): reference trajectory in our PoC experiment, with standard deviation. The posterior mean is slightly curved, since only a few demonstrations are used, and the demonstrator was trying to explore the EE positions while remaining within the admissible bounds. In Figs. 4(b) & 4(c): the corresponding estimated stiffness profile, obtained by means of HGP regression. In blue, we show the empirical profile obtained after merging the different perturbed samples, while in green we show the posterior mean (with standard deviation) of the GP. The force measurements are obtained with a force sensor on the WAM's EE (left) resp. an EMG bracelet on the supervisor's forearm (right). Furthermore, the orange curve shows the stiffness profile obtained from the variance of the DOF. Here, our algorithm and its variance-based counterpart lead to similar results, due to: the absence of environmental constraints; the human demonstrations fully exploring the WAM's configurations. Nonetheless, our approach leads to a clearer separation between the different stiffness regions, being also able to infer intermediate stiffness values (such as between 5s and 10s).
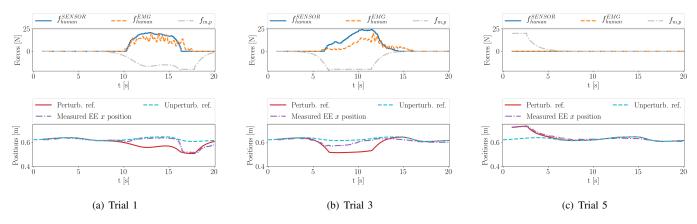

(a) Trial 1

(b) Trial 3

(c) Trial 5

Fig. 5. The x-forces involved in our PoC experiment, in the most illustrative perturbed trials. The force has been estimated both from a force sensor (f_{human}^{SENSOR}) and from an EMG bracelet (f_{human}^{EMG}). We further show the unperturbed reference, the perturbed one, and the actual EE position along the x axis. As expected, the human tries to steer the manipulator to the correct position *if needed*, compensating the force induced by the perturbation injected in the reference signal, $f_{m,p}$. Although more noisy, the EMG force profiles match the ones obtained when using an EE force sensor, i.e., muscle activity is detected in presence of a relevant perturbation only. The perturbations are constrained to keep the WAM within its workspace.

at time step t of the GP trained on the *unperturbed* human demonstrations. Let $p_i(\cdot)$ be the perturbation corresponding to the i-th perturbed demonstration. This will be defined as the standard deviation of the unperturbed demonstrations, weighted with a Gaussian curve with an optimal center. We define the knowledge function $u(\cdot)$ as

$$u(t) = \sigma(t) + \bar{p}(t), \qquad (2)$$

where

$$\bar{p}(t) = \sum_{i \in \{1,\ldots,I_p\}} p_i(t). \qquad (3)$$

We can observe that this choice for $u(\cdot)$ leads to a continuous and differentiable (provided that we use a differentiable kernel for the GP) knowledge function, where the standard deviation and the cumulative perturbations are equally weighted. Furthermore, if the perturbations are sharp enough around a certain time step, summing their tails does not hallucinate overly large values for $\bar{p}(t)$, and this in turn does not spoil the knowledge profile.

5) Optimization: At each iteration of the algorithm, we solve the following optimization problem, to retrieve the point

TABLE I
AVERAGE PEAK OF EACH COMPONENT OF $\mathbf{f}_{human}(\mathbf{t})$, ACROSS THE
PERTURBED SAMPLES OF OUR PoC EXPERIMENT.

Axis:		x	y	z
Average $\left\| f_{human}^{axis}(\mathbf{t}) \right\|_\infty$ **[N]:**		12.0 ± 8.6	2.7 ± 1.5	3.2 ± 1.7

t^* of *minimum knowledge*, with a time horizon T:

$$t^* = \arg \min_{t \in [0,T]} u(t). \qquad (4)$$

This optimization problem has a scalar domain. Furthermore, the objective function is costly to evaluate (due to the poor scaling of GPs to large datasets), and non-convex. Thus, it is optimized by means of BO.

6) Perturbation Profile: Once we have found t^* as in the previous paragraph, we can design a suitable perturbation profile. In particular, from the scheme in Fig. 2, it is clear that the perturbation is the difference between the actual reference sent to the controller, and the true posterior mean of the *unperturbed* GP. If we rescale this deviation by the posterior standard deviation $\sigma(\cdot)$, we obtain the well-known Mahalanobis distance $\mathcal{D}_m(\cdot)$ [19] between the *perturbed* tra-
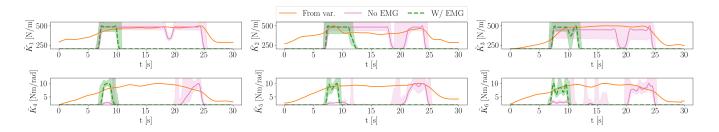
Fig. 6. The six stiffness profiles inferred in our assembly task, processed both without and with a combination of sensed EE forces and EMG signals. As expected, if we use the forces only, we obtain hallucinated stiffness profiles (in pink), that do not correspond to an optimal choice of the stiffness values. Conversely, the EMG helps the algorithm filtering out the forces exerted by the environment. The trajectory is unconstrained when the WAM is approaching the trail (at approximately 8s). Once inside the trail, the EE pose is constrained, and the stiffness can (and should) be low. Again, we show in orange the profile obtained from each DOF's variance. This approach is indeed suboptimal, as the stiffness is overestimated.

jectory sample and the posterior *unperturbed* distribution at the same time instant:

$$\mathcal{D}_m(t) = p(t)/\sigma(t). \qquad (5)$$

Ideally, we want $\mathcal{D}_m$ to have a known profile, so that it reaches a maximum at the point of minimum knowledge, t^*. This guarantees that, at such a point, we have a large perturbation that moves the robot far from the desired trajectory. This can be achieved by setting $\mathcal{D}_m$ to be a Gaussian, $sg(w, c, t)$, where $g(w, c, t) \in (0, 1]$, $\forall t$. Such a curve is parameterized with a width w, that identifies the interval around t^* in which we have a relevant deviation, a center $c = t^*$, and a scale $s > 1$, that modulates the magnitude of the perturbation[2]. Thus, we obtain the following:

$$p(t) = s\sigma(t)g(w, c, t). \qquad (6)$$

This model guarantees that the disturbance is pushing the reference trajectory at most s standard deviations away from the reference. This has two beneficial effects. On one hand, we are *exploring* by making the robot deviate from the true reference. On the other hand, we take into account the precision information coming from the demonstrations, to avoid overly large perturbations that may *damage* the robot and/or the environment. To make the perturbations clearly visible to the human supervising the learning process, we slightly modify Eq. (6) by setting a minimum offset on the posterior unperturbed standard deviation. This offset can be computed as the standard deviation between the *real* trajectory executed by the robot and the *unperturbed* reference, which we denote by $\hat{\sigma}$. Therefore, we obtain

$$p(t) = s\left[\sigma(t) + \hat{\sigma}\right]g(w, c, t). \qquad (7)$$

Our perturbation model is a generalization of a simpler perturbation policy, that would consist of applying a perturbation proportional to $\sigma(\cdot)$ (i.e., $w \to \infty$). This approach, however, might be harmful for the robot, in presence of environmental constraints, as a prolonged pressure might damage the EE. For this reason, a limited bandwidth is advisable, in compliance with the experimental setup available. Likewise, s needs to be

[2]To ensure variability across the perturbations, these parameters are sampled independently from suitable Gaussian distributions. To ensure symmetry of the perturbations, their sign follows a Bernoulli distribution with probability 0.5.
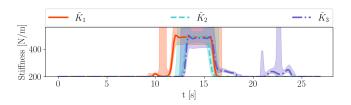

Fig. 7. The position stiffness inferred in our bed-making experiment. At the beginning, the manipulator can be compliant. Once the bed sheet has been stretched, it needs to be pulled firmly to remove the wrinkles from the surface. Thus, the WAM needs to stiffen up along all axes of motion. Then, the WAM lowers the sheet's corner, so that the it lays on the mattress completely. The contact forces between the mattress and the sheet guarantee that no wrinkles appear, in case of perturbations acting on the EE along the x and y axes (environmental constraints). A non-minimum z-gain between 20s and 25s guarantees that the corner is lowered enough.

heuristically tuned, by performing some preliminary trials. In our experiments, we kept it *constant*.

B. Stiffness Inference

Having outlined the two phases of our algorithm and defined a suitable model of the perturbations, we can move on to derive the equations used to infer the stiffness. Let us first consider the execution of a single perturbed trial, summarized by the control scheme in Fig. 2. From this control scheme, we observe that the overall force being exerted on the EE in this phase is given by:

$$f_m = \underbrace{K\overbrace{e}^{x_r-x} + D\overbrace{\dot{e}}^{\dot{x}_r-\dot{x}}}_{:=f_{m,e}} + \underbrace{Kp + D\dot{p}}_{:=f_{m,p}} + \overbrace{f_{env} + f_{human}}^{\text{External forces}}. \quad (8)$$

The perturbation p creates a hallucinated pose error, $e + \Delta e$, driving the robot to a wrong position. In particular, this error offset is equal to p, and is responsible for generating the force $f_{m,p}$ compensated by f_{human}, if needed.

1) Stiffness Calculation: Once f_{human} is measured, we can consider a *standard* impedance control scheme. Now, the robot should be able to follow the reference trajectory, and the stiffness of the controller, which we denote by $\tilde{K}$, should be time-varying and tuned according to the data gathered before. In this standard configuration, the perturbation p affects the pose and velocity of the robot's EE, as in standard control, and

is compensated by the *feedback loop* of the impedance control scheme (i.e., the reference trajectory will not be perturbed any more). Let $\tilde{D}$ be the final damping. The controller now reacts to p by exerting a force

$$\tilde{f}_{m,p} = -\tilde{K}p - \tilde{D}\dot{p}. \tag{9}$$

From the observations in the previous paragraph, we know that the human-in-the-loop has provided a force profile that represents the desired reaction to the perturbation p. Thus,

$$f_{human} = -\tilde{K}p - \tilde{D}\dot{p}. \tag{10}$$

2) Damping Calculation: The damping $\tilde{D}$ in Eq. (10) can be retrieved by enforcing that the system

$$\ddot{x} + \tilde{D}\dot{x} + \tilde{K}x = 0 \tag{11}$$

is *stable* and *critically damped*. Conditions on stability and critical damping lead to

$$\tilde{D} = 2\sqrt{\tilde{K}} > 0. \tag{12}$$

Combining this result with Eq. (10), we get that the desired stiffness is a solution of the equation

$$p\tilde{K} + 2\dot{p}\sqrt{\tilde{K}} + f_{human} = 0. \tag{13}$$

In particular, we get that, whenever $f_{human} \neq 0^3$,

$$\sqrt{\tilde{K}} = \begin{cases} \left[-\dot{p} + \sqrt{\dot{p}^2 - pf_{human}}\right]/p & \text{if } p \geq 0, \\ \left[-\dot{p} - \sqrt{\dot{p}^2 - pf_{human}}\right]/p & \text{if } p < 0. \end{cases} \tag{14}$$

C. Final Processing

Having run multiple perturbed trials as described in the previous section, we can solve Eq. 13 at each time-step, and retrieve several stiffness profiles. From these, we can retrieve a global, noisy stiffness profile as the element-wise maximum stiffness across all perturbed samples. This series is then smoothed by means of HGP regression (a logarithmic link function is used to ensure positivity of the stiffness). Our algorithm returns stiffness values in the range $[0, \hat{K}]$, according to the actual stiffness and damping employed when the perturbed samples are executed. Each value can be re-scaled in the user-defined range $[K_{min}, K_{max}]$ as

$$\tilde{K}_{final} = \tilde{K}(K_{max} - K_{min})/\hat{K} + K_{min}. \tag{15}$$

To summarize, Fig. 3 reports the steps involved in our interactive learning pipeline. The initial values of the inferred stiffness are all equal to K_{min}. Then, the repeated perturbed trials refine this estimate.

---

[3]If $f_{human} = 0$ and sign$\{p\} \neq$ sign$\{\dot{p}\}$, $\sqrt{\tilde{K}} \neq 0$. This hallucinated stiffness value is actually induced by the perturbation fading, and is zeroed.

IV. EXPERIMENTS

In this section, we test our algorithm in three experiments, namely a task in free Cartesian space, a task with environmental constraints, and an application involving *cloth manipulation*. As discussed in Sec. III, we measure the human force with a force sensor and with surface EMG sensors. In our experiments, we use a 7-DOF Barrett WAM robotic arm, with a diagonal impedance controller as described in Sec. II, and as done in [9]. As force sensor, we use an ATI force and torque sensor, mounted on the EE of the WAM. For the EMG measurements, we use the Myo Armband, a custom bracelet originally devised for remote gesture control, and equipped with 8 EMG channels [20]. In both cases, the EE pose is represented by using 7-dimensional vectors (3 DOFs for position and a quaternion for the orientation). In order to process the orientation DOFs with HGP regression, we project the quaternion in a 3-dimensional Euclidean space, and then reconstruct the reference scalar as described in the appendix of [9]. The orientation is perturbed as discussed in appendix.

A. Proof-of-Concept Experiment

We begin our experiments with a proof-of-concept (PoC) experiment. In particular, the WAM is required to move above a rectangular piece of steel along the y axis of the Cartesian space, staying within varying margins around the piece, as shown in Fig. 4(a). We are interested in tuning the stiffness along the x axis. 6 unperturbed demonstrations are gathered, followed by 9 perturbed samples.

1) Force sensor: We firstly consider the force sensor. As shown in Table I, the force exerted by the human is mostly visible along the axis of perturbation, x. Furthermore, as shown in Fig. 5, the forces applied by the human compensate the perturbations, whenever these make the arm exit from the admissible task region. Small force values due to the noise (between $-0.5N$ and $0.5N$) were zeroed out.

2) EMG Sensor: When using the EMG bracelet, we process the 8-dimensional EMG profile associated to each perturbed sample as follows. We compute a series of 8×8 covariance matrices, using a window sliding on the data (each window comprises 25 points), and compute the *trace* of each matrix. The trace is *invariant* w.r.t. permutations of the bracelet's channels, and is therefore robust in terms of different positions of the bracelet on the forearm. After zeroing out negligible trace values ($\leq 5\%$ of the largest EMG value recorded across all trials), we rescale each series in the interval $[0, \bar{f}_{m,p}]$. $\bar{f}_{m,p}$ is the maximum possible value of $f_{m,p}$ in the current perturbed trial. Each positive trace value, denoted $\mathcal{E}$, is taken as an estimate of $|f_{human}|$. This is motivated by observing that the human compensation mainly acts along the perturbation's axis, as mentioned in the previous paragraph. Again, some illustrative examples can be found in Fig. 5.

3) Stiffness Profile: Once the force measurements are available, the stiffness $\tilde{K}$ is computed for each perturbed sample, as of Eq. (13). The values of f_{human} in Eq. (13) are replaced with

$$f'_{human} = -\text{sign}\{p\} * \min\{|f_{m,p}|, |f_{human}|\}. \tag{16}$$

This choice guarantees that the values of the force are not overly large, while taking into account those parts of the trajectory where the human is not interacting with the WAM (i.e., $f_{human} = 0$). This is useful also when considering that the human force and the perturbing force are not perfectly synchronized. The trajectory for the DOF of interest, and the stiffness profiles, are shown in Fig. 4, along with the profile obtained from the DOF's variance, as proposed by [9]. Note that our choice of HGPs for the trajectory representation resides in their *non-parametric* nature. However, our inference algorithm is *agnostic* to the chosen movement primitive representation, as long as the latter provides a probabilistic distribution from which reliable variance estimates can be obtained.

B. Assembly Task

As a next step, we apply our algorithm to a task with environmental constraints, inspired by the experiments carried out in [10]. In particular, we use the WAM to insert a T-shaped piece inside a trail, and move it until it reaches the opposite side. In this way the WAM pose is constrained as long as the EE piece remains inside the trail, which is 37.5cm long. Friction inside the trail was kept low with a lubricant. We gather 7 unperturbed samples, and sample 5 perturbations per DOF.

1) Combining sensors: Here, we measure both the EE force, f_{EE}, and the EMG, processed as described in Sec. IV-A to retrieve the scalar signal $\mathcal{E}$. We propose to combine both quantities so that

$$|f_{human}| = \text{sign}\{\mathcal{E}\} |f_{EE}|. \tag{17}$$

This model has the benefit of filtering out those EE-forces that are due to environmental constraints, and are consequently not associated to the forearm's muscle activation. Without considering the EMG signal, it would be impossible for the algorithm to discriminate between human and environmental forces acting on the EE, as shown in Fig. 6.

C. Bed-Making Experiment

As a final experiment, we use the WAM in the task of bed-making, and we aim at tuning the position gains. In this task, the EE holds one corner of a folded bed sheet, and is required to move it so that the bed is covered and no wrinkles appear on the surface of the sheet. As in a realistic scenario, the sheet is partially blocked under the bed's mattress. *A priori*, it is not straightforward to identify how the constraints imposed by the mattress would affect the stiffness. Furthermore, due to the deformable and slightly elastic nature of cloth, the WAM must pull the sheet firmly once it has been stretched, in order to remove the wrinkles, even though, at that point, the sheet tends to oppose the EE motion. The force sensor was combined with the EMG sensor as described in the previous subsection, since we have environmental constraints. 5 unperturbed demonstrations were used, along with 5 perturbed samples per DOF. As shown in Fig. 7, our algorithm is able to retrieve meaningful stiffness profiles, that correspond to the different phases of the motion.[4]

---
[4]In the supplementary video, we show the robot performing the task with the inferred stiffness profiles, without and with disturbances on the EE.

TABLE II
COVERAGE $\mathcal{C}_\alpha$ [%] ACHIEVED IN OUR EXPERIMENTS, FOR $\alpha = \{1, 2, 3\}$. OUR KNOWLEDGE-BASED PLACEMENT APPROACH IS COMPARED WITH A UNIFORMLY RANDOM PLACEMENT OF THE PERTURBATIONS (MEAN $\pm$ 99% CONFIDENCE INTERVAL WITH 5 SEEDS).

Experiment	Knowledge-based			Random		
	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_3$	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_3$
PoC	100 ± 0	100 ± 0	100 ± 0	100 ± 0	99 ± 2	98 ± 3
Assembly	89 ± 0.2	62 ± 0.9	55 ± 1	84 ± 3	57 ± 3	49 ± 5
Bed-making	99 ± 0.2	99 ± 2	96 ± 2	92 ± 4	85 ± 4	80 ± 4

TABLE III
NUMBER OF PERTURBED TRIALS PER DOF, $N_\mathcal{C}$, AND TOTAL NUMBER $N_{\mathcal{C},tot}$ OF PERTURBED TRIALS, TO ACHIEVE A RANDOM-BASED AVERAGE COVERAGE $\mathcal{C}_1$ GREATER OR EQUAL TO THE ONE OBTAINED WITH OUR KNOWLEDGE-BASED METHOD (MEAN COMPUTED WITH 5 SEEDS).

Experiment	Knowledge-based		Random	
	$N_\mathcal{C}$	$N_{\mathcal{C},tot}$	$N_\mathcal{C}$	$N_{\mathcal{C},tot}$
Assembly	5	30	10	60
Bed-making	5	15	9	27

D. Validating the Perturbation Policy

As outlined in Sec. III, we use a knowledge-based policy to place the perturbations within the domain of the trajectory. To validate if we are properly reducing the ambiguity associated to the regions with low variance, we introduce the following *coverage* $\mathcal{C}_\alpha$, that counts the number of time-steps at which either the variance or a perturbation are larger than an arbitrary threshold.

Definition IV.1. Let $\mathcal{I}(\cdot)$ be the indicator function, Δ the number of DOFs, M the number of time-steps, $\sigma_d(\cdot)$ the posterior standard deviation of DOF d, and $p_j^d(\cdot)$ the j-th perturbation applied to DOF d. Furthermore, let $\hat{\sigma}_d$ be the controller tracking error of the d-th DOF as of Sec. III, and $\alpha \in \mathbb{R}^+$. We define the coverage as

$$\mathcal{C}_\alpha = \frac{1}{M\Delta} \sum_{i,d} \mathcal{I} \left\{ \sigma_d(t_i) \geq \alpha \hat{\sigma}_d \vee \left(\bigvee_j p_j^d(t_i) \geq \alpha \hat{\sigma}_d \right) \right\}. \tag{18}$$

Table II reports the coverage achieved in the tasks we described before. To show the robustness of our evaluation scheme, different values of α are used. The coverage is a good indicator for quantifying the improvement associated to one perturbed trial, and can be used when deciding how many perturbed trajectories to sample. The performance of our algorithm is compared with a uniformly random placement in $[0, t_M]$. Our algorithm outperforms the baseline, achieving a higher average coverage, and a smaller deviation than the random policy. This is a key feature, since running the perturbed samples is costly, as it involves using the WAM. Our knowledge-based approach is therefore *sample-efficient*, as shown in Table III.

V. CONCLUSIONS

In this work, we have presented a novel methodology for inferring the stiffness' profiles of a task to be performed by a robotic manipulator. Our approach, based on the concept of

perturbations and *human supervision*, easily adapts to different sensory equipment. While our work focused on introducing the framework, it can form the basis of interesting future research. For instance, the relationship between the impedance control law, the perturbations and the resulting VIC controller might be studied in terms of stability. Furthermore, the variance-based methodology could be extended to the inference of full stiffness matrices, in case large datasets of demonstrations are available. Lastly, our perturbation strategy can also be employed if the force measurements are not available. In this case, the external forces could be estimated by means of an observer as done in [21], under the assumption of negligible modeling error. Such an approach would still require the EMG measurements, as they allow to distinguish the human and the environmental forces.

APPENDIX
PERTURBING THE ORIENTATION

Let us consider a non-minimal representation of the orientation, such as quaternions. Given a desired quaternion $\mathbf{q}_d = [\eta_d, \boldsymbol{\epsilon}_d]^T$ and the true EE quaternion $\mathbf{q}_e = [\eta_e, \boldsymbol{\epsilon}_e]^T$, we get that the 3-dimensional orientation error is

$$\mathbf{e}_{ORI}(\mathbf{q}_d) = (\eta_e \boldsymbol{\epsilon}_d - \eta_d \boldsymbol{\epsilon}_e - \boldsymbol{\epsilon}_d \times \boldsymbol{\epsilon}_e) \in \mathbb{R}^3. \qquad (19)$$

Now, let us consider a local quaternion $\mathbf{q}_p$ perturbing the orientation reference [22], i.e., composed as

$$\mathbf{q}_{d,p} = \mathbf{q}_d * \mathbf{q}_p := h(\mathbf{q}_p). \qquad (20)$$

According to Eq. (19), and considering that our reference has become $\mathbf{q}_{d,p}$, this perturbation will lead to a hallucinated orientation error, $\mathbf{e}_{ORI}(\mathbf{q}_{d,p})$, seen by the impedance controller. Note that, in this case, a perturbation on the reference orientation might affect multiple components of the error at the same time, leading to a perturbation on the tracking error

$$\Delta \boldsymbol{e}_{ORI} = \mathbf{e}_{ORI}(\mathbf{q}_{d,p}) - \mathbf{e}_{ORI}(\mathbf{q}_d). \qquad (21)$$

Conversely, with a minimal representation, and a perturbation on the first DOF, we recall from Sec III-B that $\Delta \boldsymbol{e}_{ORI} = [p, 0, 0]^T$. Interestingly, in the case of quaternions, the mapping $k(\mathbf{q}_p) := \mathbf{e}_{ORI}(h(\mathbf{q}_p))$ is linear in the components of $\mathbf{q}_p$. We can further observe that $\mathbf{e}_{ORI}(\mathbf{q}_d)$ can be thought of as a reference quaternion perturbed by $[1, 0, 0, 0]^T$, i.e.,

$$\mathbf{e}_{ORI}(\mathbf{q}_d) = k([1, 0, 0, 0]^T). \qquad (22)$$

Thus, letting $\mathbf{J}_k$ be the Jacobian of $k(\cdot)$, we have

$$\mathbf{e}_{ORI}(\mathbf{q}_{d,p}) = \mathbf{e}_{ORI}(\mathbf{q}_d) + \mathbf{J}_k(\mathbf{q}_p - [1, 0, 0, 0]^T), \qquad (23)$$

and therefore

$$\Delta \mathbf{e}_{ORI} = \mathbf{J}_k(\mathbf{q}_p - [1, 0, 0, 0]^T). \qquad (24)$$

If we want the quaternion perturbation to affect only one component of the error at a time (e.g., we want an error offset of the form $[\gamma, 0, 0]$), we can obtain it as

$$\mathbf{q}_p = \mathbf{J}_k^\dagger [\gamma, 0, 0]^T + [1, 0, 0, 0]^T. \qquad (25)$$

$\mathbf{q}_e$, needed for the Jacobian calculation along with $\mathbf{q}_d$, can be obtained by running the *unperturbed* trajectory once. In this

example, the standard deviation to be used in the knowledge function and in the profile of γ, to match Def. III.1 and Eq. (6), is the one of the first DOF of $\mathbf{e}_{ORI}(\mathbf{q}_d)$. This is obtained from the (diagonal) covariance matrix of the vector part of $\mathbf{q}_d$, given by the GPs, and by applying the method in the appendix of [9], based on the quaternion rate matrix.

REFERENCES

[1] A. G. Billard, S. Calinon, and R. Dillmann, "Learning from humans," in *Springer handbook of robotics*. Springer, 2016, pp. 1995–2014.
[2] D. Kushida, M. Nakamura, S. Goto, and N. Kyura, "Human direct teaching of industrial articulated robot arms based on force-free control," *Artificial Life and Robotics*, vol. 5, no. 1, pp. 26–32, 2001.
[3] A. Paraschos, C. Daniel, J. Peters, G. Neumann *et al.*, "Probabilistic movement primitives," *Advances in neural information processing systems*, pp. 2616–2624, 2013.
[4] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.
[5] M. Arduengo, A. Colomé, J. Borràs, L. Sentis, and C. Torras, "Task-adaptive robot learning from demonstration with Gaussian process models under replication," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 966–973, 2021.
[6] S. Calinon, I. Sardellitti, and D. G. Caldwell, "Learning-based control strategy for safe human-robot interaction exploiting task and robot redundancies," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 249–254, 2010.
[7] J. Umlauft, Y. Fanger, and S. Hirche, "Bayesian uncertainty modeling for programming by demonstration," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6428–6434, 2017.
[8] J. Silvério, Y. Huang, F. J. Abu-Dakka, L. Rozo, and D. G. Caldwell, "Uncertainty-aware imitation learning using kernelized movement primitives," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 90–97, 2019.
[9] D. Parent, A. Colomé, and C. Torras, "Variable impedance control in Cartesian latent space while avoiding obstacles in null space," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9888–9894, 2020.
[10] L. Peternel, T. Petrič, and J. Babič, "Robotic assembly solution by human-in-the-loop teaching method based on real-time stiffness modulation," *Autonomous Robots*, vol. 42, no. 1, pp. 1–17, 2018.
[11] F. J. Abu-Dakka, L. Rozo, and D. G. Caldwell, "Force-based variable impedance learning for robotic manipulation," *Robotics and Autonomous Systems*, vol. 109, pp. 156–167, 2018.
[12] K. Kronander and A. Billard, "Learning compliant manipulation through kinesthetic and tactile human-robot interaction," *IEEE Transactions on Haptics*, vol. 7, no. 3, pp. 367–380, 2014.
[13] L. M. Doornebosch, D. A. Abbink, and L. Peternel, "The force-feedback coupling effect in bilateral tele-impedance," *8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob)*, pp. 152–157, 2020.
[14] F. J. Abu-Dakka, Y. Huang, J. Silvério, and V. Kyrki, "A probabilistic framework for learning geometry-based robot manipulation skills," *Robotics and Autonomous Systems*, vol. 141, p. 103761, 2021.
[15] K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard, "Most likely heteroscedastic Gaussian process regression," *24th International conference on Machine learning (ICML)*, pp. 393–400, 2007.
[16] M. Müller, "Dynamic time warping," in *Information retrieval for music and motion*. Springer, 2007, pp. 69–84.
[17] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control*. Springer, 2009.
[18] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global optimization*, vol. 13, no. 4, pp. 455–492, 1998.
[19] P. C. Mahalanobis, "On the generalised distance in statistics," *Proceedings of the National Institute of Science of India*, vol. 12, pp. 49–55, 1936.
[20] S. Rawat, S. Vats, and P. Kumar, "Evaluating and exploring the myo armband," *IEEE International Conference System Modeling & Advancement in Research Trends (SMART)*, pp. 115–120, 2016.
[21] A. Colomé, D. Pardo, G. Alenya, and C. Torras, "External force estimation during compliant robot manipulation," *IEEE international conference on robotics and automation (ICRA)*, pp. 3535–3540, 2013.
[22] J. Sola, "Quaternion kinematics for the error-state kalman filter," 2017, arXiv:1711.02508 [cs.RO].