ILoSA: Interactive Learning of Stiffness and Attractors

Giovanni Franzese*†, Anna Mészáros†, Luka Peternel, and Jens Kober

***Abstract*— Teaching robots how to apply forces according to our preferences is still an open challenge that has to be tackled from multiple engineering perspectives. This paper studies how to learn variable impedance policies where both the Cartesian stiffness and the attractor can be learned from human demonstrations and corrections with a user-friendly interface. The presented framework, named ILoSA, uses Gaussian Processes for policy learning, identifying regions of uncertainty and allowing interactive corrections, stiffness modulation and active disturbance rejection. The experimental evaluation of the framework is carried out on a Franka-Emika Panda in three separate cases with unique force interaction properties: 1) pulling a plug wherein a sudden force discontinuity occurs upon successful removal of the plug, 2) pushing a box where a sustained force is required to keep the robot in motion, and 3) wiping a whiteboard in which the force is applied perpendicular to the direction of movement.**

I. Introduction

Robots have long been a tool for efficiently carrying out repetitive or mundane tasks. As of late, more robotic applications are targeted towards interacting with varying and unknown environments in order to aid people in daily tasks. Quite often, the exact behaviour required for interacting with such environments cannot be directly modelled or is simply too complex to do so. However, people already possess intuitions on how to interact with the world around them and can transfer this knowledge. In this direction, learning through demonstration has become increasingly popular for teaching robots familiar yet complex tasks in an intuitive manner [1].

Learning is especially handy for manipulation tasks, which come with the requirement of exerting a certain degree of force. The goal of a manipulation operation is not only to perform a desired trajectory but to learn the desired force that the robot has to exercise on its environment in order to accomplish the desired goal. Different methods exist for controlling the robot to perform contact tasks, from the use of force control, hybrid position-force control [2], as well as impedance control methods [3]. In addition, when robots are coexisting with humans, it is crucial to consider that for safe interaction the robot should limit the force to the minimum required, as well as be compliant when interacting with elements of the environment that are not the target of the manipulation.

Authors are with Cognitive Robotics, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands (e-mail: g.franzese, l.peternel, j.kober@tudelft.nl, a.meszaros@student.tudelft.nl).

*Corresponding Author

†These authors contributed equally to this work.

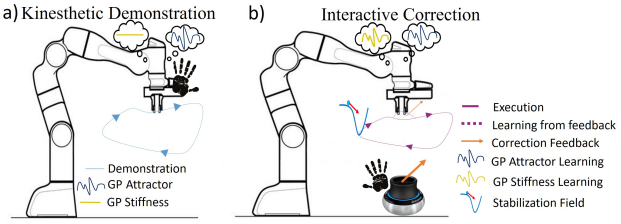

Fig. 1: Overview of the ILoSA framework

Out of the above methods, impedance control is best suited for achieving such behaviour, since losing the opposing external force in a force control would generate dangerous and unstable accelerations, while the impedance control would only converge to the nearby attractor. Furthermore, when using impedance control, for a safer and user-friendly interaction of the robot with the environment, the execution of the trajectory has to be performed in a feedback/reactive manner and not in feed-forward. This avoids the accumulation of error between the attractor and end-effector positions with consequent generation of undesired high interaction forces and/or accelerations.

In the presented framework, both the desired attractor position and the desired stiffness are learned as a function of the robot position. This is done by using a non-linear feedback policy that is learned from kinesthetic demonstrations and teleoperated corrections. More importantly, because the robot learns from a human demonstration, an estimation of the epistemic uncertainty of the policy is necessary for a safe execution of the trajectory. For this purpose, the Gaussian Process Regression (GPR) provides a parameter-free learning method that enables a good generalization in the neighborhoods of the demonstration while providing information on the confidence level of the corresponding prediction. This in turn can be utilised in order to make the robot more compliant in states the robot did not visit before, thus avoiding undesired and dangerous behaviours. To summarise, the motivation of this paper is to establish whether learning both attractor and stiffness policies in a reactive formulation with a GPR allows the performance of manipulation tasks while ensuring a safe interaction between the robot and its environment by exploiting the information of model confidence. Additionally, we introduce a new update rule for a Gaussian Process policy in order to allow data and time efficient learning from teleoperation corrections after the initial kinesthetic demonstration, and to automatically allocate the feedback as attractor or stiffness modulation. Also, we investigate the concept of adaptive disturbance rejection with the use of a stabilization prior

based on a force field proportional to the gradient of the GP variance manifold.

II. Related Work

The impedance controller can be designed in such a manner that it takes a reference force as input, which is internally converted in an attractor position as an output [4]. Alternatively, the impedance controller can directly be given the reference position as an input. This allows it to act both as a force controller when in contact with an object as well as a position controller when in free space. This particular property is utilised within the scope of the developed framework.

In order to automatically learn policies to complete complex tasks, one of the common approaches is to apply reinforcement learning (RL) algorithms such as Policy Improvement with Path Integrals (PI^2) [5]–[7], Natural Actor-Critic [8] and multi-optima policy search [9]. RL methods, however, tend to take a long time to achieve a desired performance level. In addition to this, in order to train a policy through RL, an adequate cost function is needed for which a good understanding of the task mechanics is required.

A faster alternative is to apply learning from demonstration [10], which can additionally be augmented with incremental learning [11] in order to improve a demonstrated policy. Previous works have shown that while using an impedance controller it is possible to learn varying stiffness profiles in order to carry out force interaction tasks [12], [13] as well as to allow compliance in areas outside of force interaction [6].

Additionally, for the purpose of learning an interaction with the environment, feedback from a human operator may be needed in the form of corrections for the learned model. As a possible solution, HG-Dagger [14] proposes to query corrections from the user on the basis of the model's confidence, and to aggregate these corrections online. However, the operation aggregation does not employ a data efficient rule for updating the old database according to the current corrections.

Inspired by the exploitation of model confidence in HG-DAgger, we learned an overall policy that remains compliant in regions of uncertainty for the purpose of safety. Regression based on Gaussian Processes has proven to be a viable approach towards achieving this form of behaviour in [15] where the uncertainty information was utilised in order to allocate leadership in the form of compliance to interacting agents in a bi-manual task. On the data efficiency side, PILCO [16] successfully employed GPs in RL for learning the system model and using the information about the uncertainty for a faster search of policy optimization. Analogously, Interactive Learning of Stiffness and Attractors (ILoSA) uses the information of the uncertainty for a data efficient update of the policy from user corrections, for stiffness modulation in uncertain regions and for active rejection of disturbances with a stabilization function.

In the literature, other probabilistic methods were employed for modeling the stochasticity of movement primitives [17] and inferring the desired trajectory by conditioning the model on the chosen goal. Furthermore, methods like Gaussian Mixture Models (GMMs) showed a successful application in inferring the desired manipulability ellipsoid from the consistency of multiple demonstrations [18] and in combination with a geometry-aware controller [19]. Similarly, the same method was employed for the fusion of an imitation policy with a stabilization prior [20] reducing the problem of the covariate shift. However, all of these probabilistic approaches do not allow the interactive correction of the policy after the provided demonstration and do not design the stabilization prior as a function of the model confidence, contrary to ILoSA.

Additionally, we propose to teach the desired force through automatic inference on the increase of the attractor distance or stiffness only from directional user corrections. The combination of this with a novel data efficient update rule and the exploitation of the model confidence provided a user-friendly way of teaching force tasks as described in the following Sections.

III. Framework: ILoSA

ILoSA employs two main teaching modalities: **kinesthetic demonstration** and **teleoperation feedback**, see Fig. 1. The first is used for initializing the policy for the desired dynamics of the end-effector. This policy is then executed in the second modality, whereby the user can provide online corrections to the policy.

The aim of the learned policy is to affect two particular aspects of the impedance control: the attractor distance, and the stiffness of the end-effector. Briefly, in a Cartesian impedance control [21], the end-effector dynamics are modelled in the form of a mass-spring-damper system

$$\boldsymbol{\Lambda}(\boldsymbol{q})\ddot{\boldsymbol{x}} = \boldsymbol{K}_s\Delta\boldsymbol{x} - \boldsymbol{D}\dot{\boldsymbol{x}}, \tag{1}$$

where $\boldsymbol{\Lambda}(\boldsymbol{q})$ is the physical system's Cartesian inertia matrix, $\boldsymbol{K}_s$ is a diagonal matrix with the desired stiffness in the principal directions, $\boldsymbol{D}$ is the corresponding critical damping matrix, and $\Delta\boldsymbol{x}$ is the attractor distance.

The aforementioned policy consists of multiple GP models, trained to take the 3D Cartesian position as input, and the desired attractor distance and stiffness along each of the three principal directions as output. The controlled $\Delta\boldsymbol{x}$ and stiffness $\boldsymbol{K}_s$ are then the mean values of GPRs, conditioned on the current Cartesian position of the robot.

In the initialization of the policy, following the kinesthetic demonstration, the hyper-parameters of the GP models are optimized for maximizing the expectancy of the predicted attractor distance of the provided demonstrations. The same parameters are then used for the initialization of the GP models of the stiffness in the three principal directions, however choosing a non-zero mean of K_{mean} in each direction.

In case a force sensor is installed on the end-effector, the stiffness could be initialized proportionally to the recorded external force. Our goal is to show that even if a force sensor

Algorithm 1: ILoSA

1 **a) Kinesthetic Demonstration(s)**
2 **while** *Trajectory Recording* **do**
3 Receive($\boldsymbol{x}_t$)
4 $\Delta\boldsymbol{x}^d(\boldsymbol{x}_{t-1}) = \boldsymbol{x}_t - \boldsymbol{x}_{t-1}$
5 **end**
6 Train(GPs)
7 **b) Interactive Corrections**
Data: $\boldsymbol{x}^d, \Delta\boldsymbol{x}^d, \boldsymbol{K}_s^d$
8 **while** *Control Active* **do**
9 Receive($\boldsymbol{x}$)
10 $[\Delta\boldsymbol{x}, \Sigma]$ = GP$_{\Delta\boldsymbol{x}}(\boldsymbol{x})$
11 $\boldsymbol{K}_s$ = GP$_{\boldsymbol{K}_s}(\boldsymbol{x})$
12 $[\Delta\boldsymbol{x}^i, \boldsymbol{K}_s^i]$=**Interpret**(Human feedback, $\Delta\boldsymbol{x}, \boldsymbol{K}_s$)
13 **if** *Received Human feedback* **then**
14 **if** $\Sigma > \Sigma_{Threshold}$ **then**
15 Append($\boldsymbol{x} \rightarrow \boldsymbol{x}^d, \Delta\boldsymbol{x} + \Delta\boldsymbol{x}^i \rightarrow \Delta\boldsymbol{x}^d, \boldsymbol{K}_s + \boldsymbol{K}_s^i \rightarrow \boldsymbol{K}_s^d$)
16 **else**
17 Correct($\Delta\boldsymbol{x}^i \rightarrow \Delta\boldsymbol{x}^d, \boldsymbol{K}_s^i \rightarrow \boldsymbol{K}_s^d$))
18 **end**
19 Fit(GPs)
20 **end**
21 $\Delta\boldsymbol{x}$ =GP$_{\Delta\boldsymbol{x}}(\boldsymbol{x})$
22 $\boldsymbol{K}_s$ =GP$_{\Delta\boldsymbol{K}_s}(\boldsymbol{x})$
23 $\boldsymbol{f}_{stable} = -\alpha\nabla\Sigma$
24 $[\Delta\boldsymbol{x}, \boldsymbol{K}_s]$=**Modulation**($\Delta\boldsymbol{x}, \boldsymbol{K}_s, \boldsymbol{f}_{stable}, \Sigma$)
25 Send($\Delta\boldsymbol{x}, \boldsymbol{K}_s$)
26 **end**

is not available, the stiffness can be initialized to a base value and the desired deviations can be learned with the interactive human corrections.

ILoSA additionally incorporates two safety features. The first is a stabilization prior which ensures robust control. The second is a modulation function which pulls the stiffness down to zero in regions of high uncertainty. These two aspects will be explained further in the course of this section.

In the following subsections, details are reported on how the GP learns from demonstration and corrections (Sec. III-A), how the directional feedback is spread between attractor and stiffness (Sec. III-B), and how a stabilization prior (Sec. III-C) and stiffness modulation (Sec. III-D) are obtained as a function of the process variance.

A. Interactive Learning with Gaussian Processes

A Gaussian Process is a non-parametric regression method that provides the means for inferring prediction and epistemic uncertainty with a clear mathematical expression. The two equations that govern the mean and the variance of the process are

$$\mu(\hat{\boldsymbol{x}}) = \boldsymbol{k}_*(\boldsymbol{x}, \hat{\boldsymbol{x}})^\top (\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}) + \sigma_n^2 \boldsymbol{I})^{-1} \boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}, \hat{\boldsymbol{x}}) \boldsymbol{y}, \quad (2)$$

$$\Sigma = k(\hat{\boldsymbol{x}}, \hat{\boldsymbol{x}}) - \boldsymbol{k}_*(\boldsymbol{x}, \hat{\boldsymbol{x}})^\top (\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}) + \sigma_n^2 \boldsymbol{I})^{-1} \boldsymbol{k}_*(\boldsymbol{x}, \hat{\boldsymbol{x}}), \quad (3)$$

where k is the variance of the evaluation point $\hat{\boldsymbol{x}}$, $\boldsymbol{k}_*$ is the covariance between $\hat{\boldsymbol{x}}$ and the training inputs $\boldsymbol{x}$, $\boldsymbol{K}$ is the covariance matrix of the training inputs, σ_n^2 is the variance of the Gaussian noise of the training points, and $\boldsymbol{y}$ denotes the training outputs [22]. Both k and $\boldsymbol{k}_*$ as well as $\boldsymbol{K}$ are a function of the kernel and its hyper-parameters used to incorporate prior knowledge in the process. After the kinesthetic demonstration(s) (Fig. 1-a), the hyper-parameters are automatically determined through Expectation Maximisation with the L-BFGS method. We are going to distinguish between *Training* (l. 6 of Alg. 1), when the optimization of the hyperparameters is performed, and *Fitting*, when the hyperparamenters are kept constant (l. 19 of Alg. 1). That is, when the interactive corrections are provided through teleoperation with the human-in-the-loop, the hyper-parameters are kept invariant because the correlation between samples (and validity of the same kernel) can also be considered invariant.

The interactive correction in ll. 13-20 of Alg. 1 summarizes how ILoSA exploits the use of the uncertainty measure for understanding if the robot is in a new unvisited region (l. 14). This requirs to add the corrective sample to the database. Otherwise, it determines how to spread the correction on all the existing samples that are correlated with the current end-effector position without adding additional samples. Thus, the update rule of the database is

$$\mu + \epsilon_\mu = \boldsymbol{A}(\boldsymbol{x}, \hat{\boldsymbol{x}})(\boldsymbol{y} + \boldsymbol{\epsilon_y}) \rightarrow \boldsymbol{y}^{new} = \boldsymbol{y} + \boldsymbol{A}(\boldsymbol{x}, \hat{\boldsymbol{x}})^+ \epsilon_\mu, \quad (4)$$

where $\boldsymbol{A}(\boldsymbol{x}, \hat{\boldsymbol{x}})^+$ is the pseudoinverse of $\boldsymbol{A}$ and ϵ_μ is the output of the teleoperation correction. $\boldsymbol{A}^+$ can be seen as a selector that automatically modulates how much the correlated elements in the database should be modified for matching the user's desired corrections. This rule was applied for correcting the attractor distance, stiffness or both according to the interpretation of the human feedback, see Sec. III-B.

This way of spreading corrections on the database showed to be more user-friendly, as well as time and data efficient. As also shown in other researches [23], [24], having contradictory, incremental or multimodal data can generate a bias of the predicted solution towards the more frequent samples. When doing policy correction, this can result in the unobservability of the effect of feedback and in the user's frustration. The proposed update rule resolves this problem and proved to also be effective in rapidly adjusting mistaken corrections provided in the previous policy roll-outs without the accumulation of them as noise in the database.

B. Directional Feedback Interpretation

Our aim is to make the teaching as simple as possible for non-skilled users without any knowledge about robot control. As the name of the framework suggests, the goal is to learn the attractor and stiffness for the robot end-effector. Related works like [25] and [26] already investigated how to teleoperate the robot while also teaching the desired stiffness ellipsoid of the end-effector. Contrary to this, our goal is to be able to infer the stiffness not from an explicit labelling of it but through the same directional teleoperated feedback

Algorithm 2: Interpret Correction Feedback

Interpret($input$, $\Delta\boldsymbol{x}$,$\boldsymbol{K}_s$)
if $\text{sign}(\Delta\boldsymbol{x}) = \text{sign}(input)$ **then**
  $[\Delta\boldsymbol{x}^i = input; \boldsymbol{K}_s^i = 0]$
  if $||\Delta\boldsymbol{x} + input|| > \Delta_{lim}$ **then**
    $\Delta\boldsymbol{x}^i = \text{sign}(\Delta\boldsymbol{x})(\Delta_{lim} - ||\Delta\boldsymbol{x}||))$
    $\boldsymbol{K}_s^i = \min(\boldsymbol{K}_s(\frac{||\Delta\boldsymbol{x}+input||}{\Delta_{lim}} - 1), \boldsymbol{K}_{max})$
  end
else
  $[\boldsymbol{K}_s^i = \boldsymbol{K}_s\left(\frac{input}{\Delta\boldsymbol{x}}\right); \Delta\boldsymbol{x}^i = 0]$
  if $\boldsymbol{K}_s + \boldsymbol{K}_s^i < \boldsymbol{K}_{mean}$ **then**
    $\Delta\boldsymbol{x}^i = \Delta\boldsymbol{x}\left(\frac{\boldsymbol{K}_s+\boldsymbol{K}_s^i}{\boldsymbol{K}_{mean}} - 1\right)$
    $\boldsymbol{K}_s^i = \boldsymbol{K}_s - \boldsymbol{K}_{mean}$
  end
end
return $\Delta\boldsymbol{x}^i$, $\boldsymbol{K}_s^i$

without the use of any expensive device.Accordingly, the main idea is to enable the user to incrementally correct the distance of the attractor up to a limited magnitude. Beyond this limit, the desired corrections are translated into a change of the stiffness in order to match the desired dynamics of the end-effector.

This approach does not only simplify the feedback modality but also facilitates the teaching of force tasks with abrupt discontinuities. For example, in the scenario of cable unplugging, having a closer attractor with a higher stiffness helps to prevent the "recoil" effect when the object separation happens. Similarly, if we are pushing a heavy box, the limitation of the attractor distance bounds the robot velocity in case the contact point with the box is lost (see Fig. 5). This allows a safer coexistence of the robot in anthropocentric environments [27]. Details about the implemented feedback interpretations are described in Alg. 2 where also the case of feedback in the opposite direction of the motion are considered.

Finally, in the interactive session, for the purpose of explicitly labelling the desired goal point with zero velocity and high stiffness, a further teleoperation input was employed.

C. Stabilizing Attractive Field

External forces can lead the robot end-effector in previously unvisited regions of the workspace where the extrapolation of the desired $\Delta\boldsymbol{x}$ and $\boldsymbol{K}_s$ can have high uncertainty and lead to a dangerous and undesired dynamics of the robot. This problem, known as covariate shift is common when applying Behavioural Cloning, and some solutions like DART [28] or (HG-)DAgger [14] investigated the injection of noise in the supervised policy execution in order to lead the robot in unvisited regions and collect a database in a larger portion of the environment. This technique could also be applied in the Interactive Corrections segment (Fig. 1-b), however collecting many correction points can be time consuming and highly data inefficient.

As an alternative, we want to exploit the information of the model variance and its continuous differentiability for modelling how to reject external disturbing forces, i.e. not related with the manipulation operation. Intuitively, we can imagine the variance manifold as a hyper-plane with a furrow that is generated in proximity of the labelled regions of the workspace, as we do when we create the circuit for a marble race on the beach. In the absence of external disturbances, the end-effector would lay in the regions of minimum variance and move inside there. However, the robot should reject forces, that are leading its motion to a region of uncertainty, proportionally to the rate of change of the same measure. Equivalently, when the external forces are not disturbing the motion anymore, ideally, the robot should go back in regions where the predictions have higher confidence. It is similar to adding a gravitational term in the variance manifold inducing the end-effector to always "fall" into regions of minimum variance, as a marble would come back on track when disturbed by any collision. The implementation of this stabilization prior is straightforwardly a force field that is proportional to the gradient of the variance manifold according to:

$$\boldsymbol{f}_{stable}(\hat{\boldsymbol{x}}) = -\alpha\nabla\Sigma = \alpha\left(2\boldsymbol{k}_*^\top(\boldsymbol{K} + \sigma_n^2\boldsymbol{I})^{-1}\frac{\partial\boldsymbol{k}_*}{\partial\hat{\boldsymbol{x}}}\right), \quad (5)$$

where $\hat{\boldsymbol{x}}$ is the evaluation point. Since the kernel is a function of a set of hyperparamenters, that are fitted for the different tasks, this turns into a different rejection behaviour according to the fitted model. For example, if the model gets uncertain faster it will act with a stronger force because it does not want to go in regions where it would not know how to act. Additionally, the use of this prior also results in another interesting behaviour when multiple demonstrations are provided. In the regions of lower consistency of the demonstration we can imagine a broader furrow in the variance manifold compared to regions of highly consistent demonstrations. In the first case, there is a larger track where the "marble" can move before reaching the borders and getting pulled down; on the contrary, in the second case the narrow furrow forces the robot to stay closer to the consistent demonstrations. This different behaviour of reacting to external disturbances can be interpreted as adaptive disturbance compliance of the robot along the lines of [29], where higher variance in the demonstration results in higher robot affordance and vice-versa.

D. Stiffness and Attractor Modulation

Finally, before sending the desired attractor and the stiffness to the robot, we want to make sure to spread the effect of $\boldsymbol{f}_{stable}$ as stiffness and attractor modulation in order to respect the constraint of having a limited attractor and to obtain the desired force with an increase of stiffness, similarly to how it is done in Alg. 2.

Additionally, when the robot is in a position where the uncertainty approaches the maximum, it is safer to pull the robot stiffness down to zero, rather than the predicted mean

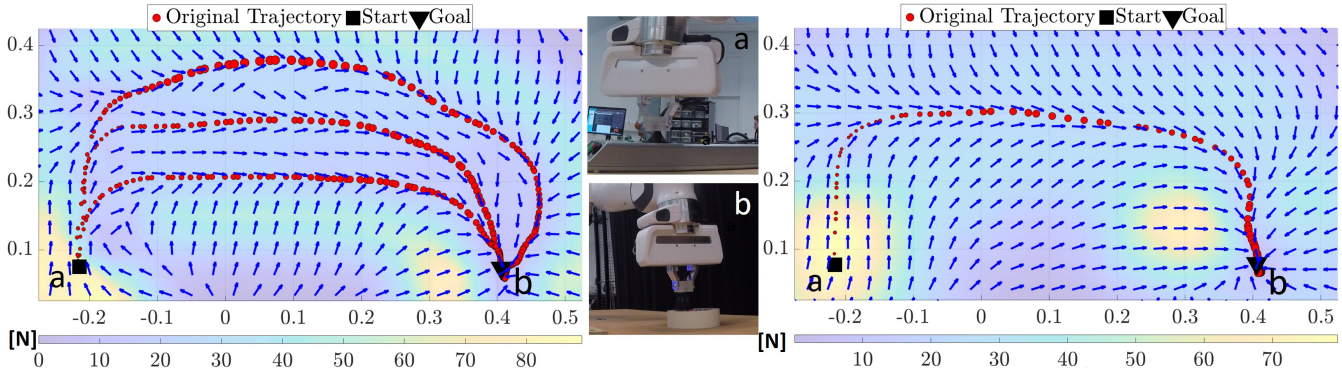

Fig. 2: Example of attractor fields for unplugging with multiple demonstrations (left) and a single demonstration (right).

value of the GPR, according to

$$\boldsymbol{K}_s = GP_{\boldsymbol{K}_s}(\boldsymbol{x}) \left(\frac{1 - \Sigma/\Sigma_{max}}{1 - tr} \right) \text{ when } \Sigma/\Sigma_{max} > tr, \quad (6)$$

where Σ_{max} is variance of the unconditioned GP with the defined kernel. These two operations are summarized in l. 23 of Alg. 1. Thanks to the property of distance-based kernels of having the prediction to vanish in high-uncertainty regions, and the modulation of the stiffness, the risk of moving in unknown regions of the workspace with possible undesired behaviours is mitigated. Moreover, this behaviour results in the robot stopping with high compliance and can be seen as a non-verbal request of teaching, or re-positioning into regions closer to the demonstration.

IV. Validation Experiments

In order to validate our approach we carry out experiments on three different manipulation tasks, each with its own variations intended to test the different aspects of ILoSA.

The first involves removing a plug from its socket and bringing it to a specified goal (Sec. IV-A). In this scenario, we test the effect of the number of demonstrations in broadening the variance furrow. Additionally, the effect of the stabilisation prior for rejecting disturbances is analysed by carrying out the control, in one case with the prior active, and in another without it, all while injecting a randomised force disturbance.

The second scenario is pushing a box to a goal (Sec. IV-B) and observing the handling of contact loss. While impedance control already ensures that the acceleration does not increase progressively upon contact loss, ILoSA additionally enables the limitation of the observed velocities by limiting the maximum attractor distance. In contrast, if the force were to be achieved by increasing the attractor distance while maintaining a low constant stiffness, these velocities could be noticeably greater. To test this, an ablation study was carried out.

Lastly, a periodic perpetual movement scenario in the form of cleaning a whiteboard is analysed (Sec. IV-C). This scenario brings with it the additional challenge that the desired attractor position is located behind the board. Due to the lack of a force sensor, this cannot be inferred from the demonstrations. Instead, user corrections are required for the robot to learn to exert the required force on the board for successfully cleaning it. An additional challenge was addressed to ILoSA in this scenario: validate the flexibility of altering a taught behaviour to new situations. To showcase this, an obstacle was placed along the original trajectory, limiting the possible height of the motion. The aim was then to provide corrections such that the robot would perform the task while simultaneously avoiding the obstacle.

All of the experiments in the three scenarios are carried out five separate times.

For our experiments we utilise the 7 DoF Franka-Emika Panda with an impedance controller and a ROS communication network for the online update of the attractor and stiffness using the ILoSA algorithm with a frequency of 15 Hz. A 3Dconnexion SpaceNavigator (see Fig. 3) was used for providing teleoperation feedback, whereof one of the two available buttons (seen circled in the image), was used for explicitly marking the desired goal position as noted at the end of Sec. III-B. For all of these experiments, the following parameters were chosen within ILoSA; a mean stiffness of $K_{mean} = 600\,\mathrm{N/m}$ with the stiffness limited to $K_{max} = 2000\,\mathrm{N/m}$, a maximum attractor distance of $0.04\,\mathrm{m}$ along each principal axis, the epistemic uncertainty threshold for adding new points to the database set to $\Sigma_{Threshold} = 0.2\Sigma_{max}$, and the threshold for modulating the stiffness $tr = 0.99$. The squared exponential kernel was selected within the GP models.

Fig. 3: SpaceNavigator

For quantifying data efficiency, we compute the ratio between the amount of corrections that result in a modification of the existing database and the total amount of provided feedback inputs. The feedback time was computed as the amount of time the user spent explicitly providing corrective inputs.

A video of the learning and execution of the tasks can be found attached to this paper [1].

[1] https://youtu.be/3j8GN5E5NiI

TABLE I: Performance in Unplugging

	Demo Time [s]		Feedback Time [s]		Data Efficiency [%]		Goal Error [m]	
	Single	Multiple	Single	Multiple	Single	Multiple	Single	Multiple
Max	8.00	28.27	3.47	2.13	97.06	97.06	0.016	0.030
Mean	6.73	23.84	1.96	1.61	95.36	96.10	0.009	0.014
Min	5.67	21.40	1.27	1.40	92.86	95.65	0.003	0.008

TABLE II: Effect of Stabilization

	Goal Error [m]	
	Without Prior	With Prior
Max	0.756	0.040
Mean	0.337	0.033
Min	0.073	0.019

A. *Unplugging*

Two variations of this scenario were performed. In the first, three separate demonstrations were carried out with different heights of the trajectory towards the goal. In the second, a single demonstration was provided. The primary focus of the corrections is placed on the successful unplugging as well as reaching the goal within a tolerance of 3 cm. A standard type F plug was used for which specific 3D-printed gripper jaws were designed to ensure a firm grip throughout the interaction. The interaction commences from the point in which the robot is already gripping the plug.

Fig. 2 visualises examples of the resulting attractor fields for both the single and multiple demonstrations. As expected, the highest forces are exerted at the beginning, during the unplugging. Instances of moderate forces can be observed leading towards the trajectories. In particular for the single demonstration, moderate forces are close to the demonstration itself and are, in fact, directed towards the demonstrated trajectory. For the multiple demonstrations these moderate forces are primarily present outside the region of demonstration. This is attributed to the larger variance furrow, which reduces the effect of the stabilisation prior in the demonstrated region, and in turn enables the robot to move more freely within the region when perturbed.

The results regarding the precision in reaching the goal indicate that for both variations, the robot was able to successfully complete its task. Slightly larger errors in the case of multiple demonstrations can be seen. This can, however, be attributed to the variations in the final positions provided during the multiple demonstrations. The time spent giving corrections in order to complete the task successfully was similar between the two experiments with an average time of 1.96 s for the single demonstration and an average time of 1.61 s for the case with multiple demonstrations. For additional details refer to Table I. For both experiment variations, the majority of feedback inputs did not increase the size of the database, showing in both cases a high data efficiency of more than 95% on average.

In the tests with the randomised perturbations, the disturbance was sampled for each of the three axes from a normal distribution $\mathcal{N}(10, 5)$N at 1/3 of ILoSA's update frequency. Here, the benefit of the stabilisation prior is clear. When using the stabilisation prior, the error from the goal was on average ten times lower than when the prior was not present. When using the prior, despite the perturbations, the robot remained close to the goal, with the highest observed error being 4 cm, indicating high robustness. Furthermore, when the prior was not present, the robot had diverged in 3 of the 5 trials and was unable to reach the vicinity of the goal.

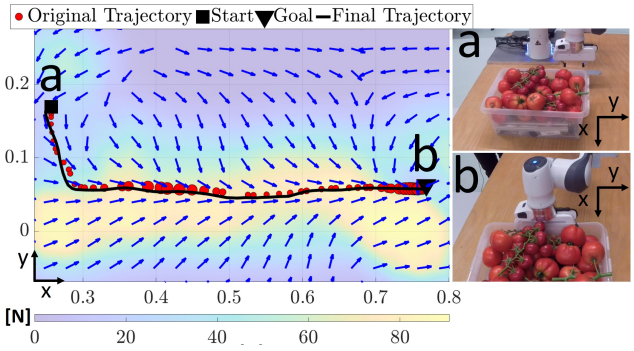

Fig. 4: Example of an attractor field for the box pushing task

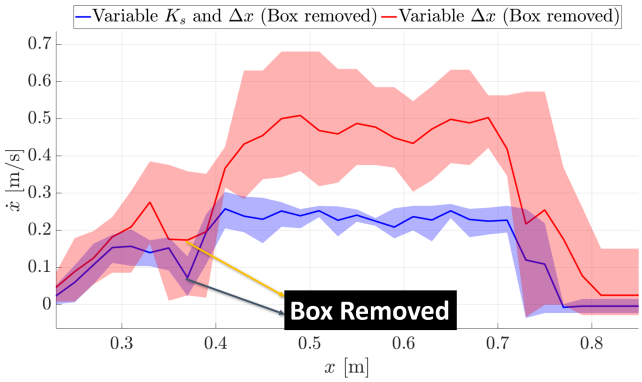

Fig. 5: Robot velocity w.r.t. current position along trajectory

Additional details are presented in Table II.

B. *Pushing a Box*

In real-world scenarios it can easily happen that objects are unwieldy for the robot's gripper, such that the only remaining option for manipulating said objects is pushing them. In such an interaction, it can happen that the object is removed prematurely, resulting in an unexpected contact loss for the robot. For the ablation study, the task was first learned with ILoSA, wherein both stiffness and attractor distance are variable; afterward it was learned with a variation of the ILoSA algorithm, wherein the force is altered only through a variable attractor distance. In both the cases, once the interaction was learned, the task was executed with the box being removed while it was being pushed. Fig. 5 displays the resulting velocities when varying solely the attractor distance (red), as opposed to concurrently varying the attractor distance and stiffness (blue).

As can be seen, the peak velocity for the combined variation of both stiffness $\boldsymbol{K_s}$ and attractor distance $\Delta\boldsymbol{x}$ is less than half compared to only varying the attractor distance.

TABLE III: Performance in Pushing a Box

	Demo [s]	Fdbk [s]	Eff. [%]	Goal Err. [m]
Max	6.80	4.53	98.57	0.016
Mean	5.23	4.16	95.82	0.008
Min	4.47	3.87	90.00	0.001

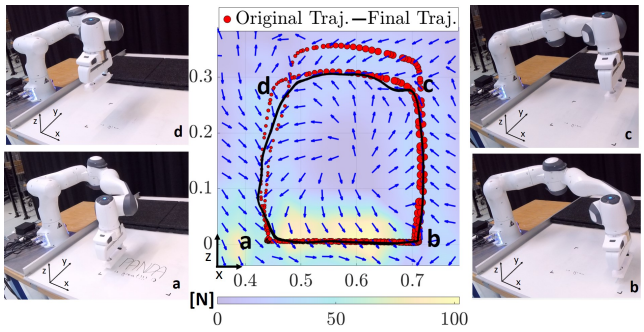

Fig. 6: Example of an attractor field for wiping a board

This in turn allows the limitation of potential impact forces should a person cross the robot's trajectory.

In terms of performance, ILoSA achieves comparable results to those observed in unplugging, both in terms of error from the goal and data efficiency. The overall correction duration as well as the correction duration relative to the one of the demonstration are larger than those observed when unplugging. This is, however, primarily attributed to the fact that the box pushing scenario has a larger portion of the trajectory in which force has to be applied, in turn requiring more user corrections. An example attractor field can be seen in Fig. 4.

On the matter of data efficiency, it should be noted that both in the case of the plug and in the case of the box, a data efficiency of 100% was not achieved due to the use of goal conditioning, which adds the marked goal to the database. For an overview on the task performance, refer to Table III.

C. Cleaning a Whiteboard

In this task, the robot is taught to ensure a whiteboard remains clean and to sustain the movement until the controller is stopped. For this, it is highly desirable that at the end of each operation cycle the robot returns to the same joint configuration; this property is known as "cyclicity" of motion [30]. For a redundant robot, it is possible for the end-effector to return to the same task-space position and yet the robot to be in a completely different joint configuration. This would result in a feeling of unpredictability for the human watching with consequent frustration. In fact, this is generally the result obtained when methods based on Cartesian impedance control are used to control the robot motion. For solving this problem we also learned a null-space control policy (always from demonstrations) and had it running during the normal execution of ILoSA. During the kinesthetic demonstration it assists the user, allowing them to focus only on the motion of the end effector, and during policy execution it guarantees the cyclicity of the operations. However, because the main focus was not about the learning of null-space constraints,

TABLE IV: Performance in Cleaning a Whiteboard

	Demo [s]	Fdbk [s]		Data Eff. [%]		Consist. [m]	
		Orig.	Adap.	Orig.	Adap.	Orig.	Adap.
Max	18.27	6.4	8	100	91.67	0.004	0.004
Mean	16.81	4.81	5.57	98.54	83.14	0.003	0.003
Min	15.20	3.53	3.27	94.51	73.47	0.003	0.003

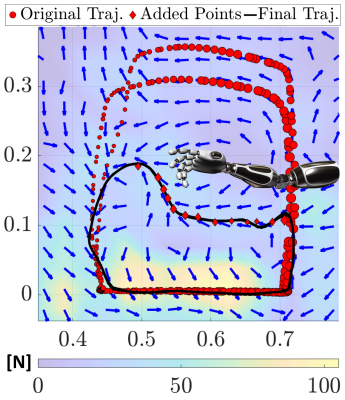

Fig. 7: Vector-field and overlapping of 5 final trajectory (black line) after user corrections for learning obstacle avoidance represented by the arm in the picture.

more details and investigations will be added in future works.

The execution of this task was deemed successful if the desired area of the board was wiped clean after each loop and the motion continued for at least 5 loops. An example of the resulting attractor field can be seen in Fig. 6.

On the quantitative side, not only were the 5 loops executed successfully, but the robot also remained highly consistent in its motion. The consistency was measured as the highest RMSE between each pair of the five loops and it amounted on average to $0.36\,\mathrm{cm}$. Part of the success of a repeatable motion can be credited to the null-space control that ensures a cyclic joint configuration and successively a consistent Cartesian mass matrix and dynamics. In terms of correction time, only a short period was spent providing inputs, with an average time of $4.81\,\mathrm{s}$. Out of these inputs, the majority resulted in modifications of the database attaining an average data efficiency of 98.54%. Additional details are provided in Table IV.

When correcting the original trajectory, adaptations for avoiding the obstacle could be carried out in all five trials. With this we wanted to show how, when close to the original provided samples, the corrections can effectively modify the database to address the desired behaviour and how, when outside the region of certainty, new samples are added, allowing also to shape the stabilization field around them. In Fig. 7, it is possible to see how around the new added points the force field would reject disturbances and stay on the new desired trajectory. Through an additional $5.57\,\mathrm{s}$ of corrections on average, and with an average data efficiency of 83.14%, the motion was successfully adapted, resulting in attractor fields similar to the one seen in the figure.

V. Conclusions and Future Work

Throughout this paper we have introduced a non-parametric and interactive approach to learning different types of force interaction tasks from humans, while exploiting impedance control to ensure safe interaction. All aspects

of the interactions, from the attractor distance and stiffness at the end-effector, to the null-space control were successfully modelled with the help of Gaussian Processes.

Making use of the learned model parameters, it was possible to establish two additional safety features. The first is the reduction of the stiffness, should the robot be too far from the demonstrated region, eventually bringing it to a halt. The second is a stabilisation prior, which helps to steer the robot back to the closest area with low variance, consequently returning it back to the demonstrated region. As a result, this enables the rejection of disturbances.

Moreover, the stabilisation prior was able to infer that its effect should be reduced in the areas between multiple demonstrations, allowing the robot more freedom of movement in those areas. However, in case of desired multi-modal behaviours, e.g., for obstacle avoidance, this could be obtained with a constraint on the maximum lengthscale of the used kernel. This is equivalent to the generation of multiple separate variance furrows rather than a single wider one.

Our investigations further showed that the ILoSA framework can be implemented for carrying out both goal-oriented and periodic movements. When used in combination with a learned null-space control, which enabled cyclicity, a high consistency of the motion was attained. Overall, ILoSA exhibited good reliability in the execution of the examined tasks, while learning in a user-friendly and data efficient manner.

Due to the successful applications in the force tasks in which it was tested, ILoSA will be extended to further challenges in the field of robot manipulation. The learning will not only focus on a single trajectory but in the assembly of movement sequences for more complex tasks, always learning from demonstration. Adaptation of the motion with respect to a particular reference frame in each segment will be investigated using human feedback and the information on the model confidence for solving possible ambiguity as in [31]. Modulation of the velocity from corrections will be further investigated for learning tasks such as fast grasping. Different modalities of feedback, from haptic devices or kinesthetic perturbations, will also be tested in specially designed user experiments, along the lines of [32]. The goal is to prove the possible leap from academic to industrial/household applications.

ACKNOWLEDGEMENTS

This research is founded by European Research Council Starting Grant TERI "Teaching Robots Interactively", project reference 804907.

REFERENCES

[1] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 469–483, 2009.

[2] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators," *ASME Journal of Dynamical Systems, Measurement and Control*, vol. 105, p. 126–133, 1981.

[3] N. Hogan, "Impedance control: An approach to manipulation," in *American Control Conf.* IEEE, 1984, pp. 304–313.

[4] B. Siciliano and L. Villani, *Robot force control*. Springer Science & Business Media, 2012, vol. 540.

[5] M. Kalakrishnan, L. Righetti, P. Pastor, and S. Schaal, "Learning force control policies for compliant manipulation," in *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2011, pp. 4639–4644.

[6] J. Buchli, F. Stulp, E. Theodorou, and S. Schaal, "Learning variable impedance control," *The Int. Journal of Robotics Research*, vol. 30, no. 7, pp. 820–833, 2011.

[7] M. Hazara and V. Kyrki, "Reinforcement learning for improving imitated in-contact skills," in *IEEE-RAS 16th Int. Conf. Humanoid Robots (Humanoids)*, 2016, pp. 194–201.

[8] B. Kim, J. Park, S. Park, and S. Kang, "Impedance learning for robotic contact tasks using natural actor-critic algorithm," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 2, pp. 433–443, 2010.

[9] S. Calinon, P. Kormushev, and D. G. Caldwell, "Compliant skills acquisition and multi-optima policy search with em-based reinforcement learning," *Robotics and Autonomous Systems*, vol. 61, no. 4, pp. 369–379, 2013.

[10] J. Kober, M. Gienger, and J. J. Steil, "Learning movement primitives for force interaction tasks," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, 2015, pp. 3192–3199.

[11] T. Petrič, A. Gams, L. Colasanto, A. J. Ijspeert, and A. Ude, "Accelerated sensorimotor learning of compliant movement primitives," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1636–1642, 2018.

[12] F. J. Abu-Dakka, L. Rozo, and D. G. Caldwell, "Force-based learning of variable impedance skills for robotic manipulation," in *IEEE-RAS 18th Int. Conf. Humanoid Robots (Humanoids)*, 2018, pp. 1–9.

[13] L. Rozo, S. Calinon, D. Caldwell, P. Jiménez, and C. Torras, "Learning collaborative impedance-based robot behaviors," in *AAAI Conf. Artificial Intelligence*, vol. 27, no. 1, 2013.

[14] M. Kelly, C. Sidrane, K. Driggs-Campbell, and M. J. Kochenderfer, "HG-DAgger: Interactive imitation learning with human experts," in *Int. Conf. Robotics and Automation (ICRA)*, 2019, pp. 8077–8083.

[15] Y. Fanger, J. Umlauft, and S. Hirche, "Gaussian processes for dynamic movement primitives with application in knowledge-based cooperation," in *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2016, pp. 3913–3919.

[16] M. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data-efficient approach to policy search," in *28th Int. Conf. Machine Learning (ICML)*, 2011, pp. 465–472.

[17] A. Paraschos, C. Daniel, J. Peters, G. Neumann *et al.*, "Probabilistic movement primitives," *Advances in Neural Information Processing Systems*, 2013.

[18] S. Calinon, "Robot learning with task-parameterized generative models," in *Robotics Research*. Springer, 2018, pp. 111–126.

[19] N. Jaquier, L. Rozo, D. G. Caldwell, and S. Calinon, "Geometry-aware manipulability learning, tracking, and transfer," *The Int. Journal of Robotics Research*, 2020.

[20] E. Pignat and S. Calinon, "Bayesian Gaussian mixture model for robotic policy imitation," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4452–4458, 2019.

[21] N. Hogan, "Impedance control - An approach to manipulation. I - Theory. II - Implementation. III - Applications," *ASME Transactions Journal of Dynamic Systems and Measurement Control B*, vol. 107, pp. 1–24, Mar. 1985.

[22] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[23] R. Perez-Dattari, C. Celemin, G. Franzese, J. Ruiz-del Solar, and J. Kober, "Interactive learning of temporal features for control: Shaping policies and state representations from human feedback," *IEEE Robotics & Automation Magazine*, vol. 27, no. 2, pp. 46–54, 2020.

[24] A. Bobu, D. R. Scobee, J. F. Fisac, S. S. Sastry, and A. D. Dragan, "Less is more: Rethinking probabilistic models of human behavior," in *ACM/IEEE Int. Conf. Human-Robot Interaction*, 2020, pp. 429–437.

[25] A. Ajoudani, N. Tsagarakis, and A. Bicchi, "Tele-impedance: Teleoperation with impedance regulation using a body–machine interface," *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1642–1656, 2012.

[26] 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.

[27] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Requirements for safe robots: Measurements, analysis and new insights," *The Int.*

Journal of Robotics Research, vol. 28, no. 11-12, pp. 1507–1527, 2009.

[28] M. Laskey, J. Lee, R. Fox, A. Dragan, and K. Goldberg, "Dart: Noise injection for robust imitation learning," in *Conf. Robot Learning*. PMLR, 2017, pp. 143–156.

[29] S. Calinon, I. Sardellitti, and D. G. Caldwell, "Learning-based control strategy for safe human-robot interaction exploiting task and robot redundancies," in *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2010, pp. 249–254.

[30] H. Seraji, "Configuration control of redundant manipulators: Theory and implementation," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 4, pp. 472–490, 1989.

[31] G. Franzese, C. E. Celemin, and J. Kober, "Learning interactively to resolve ambiguity in reference frame selection," in *Conf. Robot Learning (CoRL)*, 2020.

[32] 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, 2013.