Robotic Skill Mutation in Robot-to-Robot Propagation During a Physically Collaborative Sawing Task

Maessen, Rosa E.S.; Prendergast, J. Micah; Peternel, Luka

DOI
10.1109/LRA.2023.3307289

Publication date
2023

Document Version
Final published version

Published in
IEEE Robotics and Automation Letters

Robotic Skill Mutation in Robot-to-Robot Propagation During a Physically Collaborative Sawing Task

Rosa E.S. Maessen, J. Micah Prendergast, and Luka Peternel, *Member, IEEE*

***Abstract*—Skill propagation among robots without human involvement can be crucial in quickly spreading new physical skills to many robots. In this respect, it is a good alternative to pure reinforcement learning, which can be time-consuming, or learning from human demonstration, which requires human involvement. In the latter case, there may not be enough humans to quickly spread skills to many robots. However, propagation among robots without direct human supervision can result in robotic skills mutating from the original source. This can be beneficial when better skills might emerge or when a new skill is obtained to be used for other similar tasks. However, it can also be dangerous in terms of task execution safety. This letter studies the mutation of a robotic skill when it is propagated from one robot to another during a physically collaborative task. We chose the collaborative sawing task as a study case since it involves complex two-agent physical interaction/coordination and because its periodic nature can facilitate repetitive learning. The study employs periodic Dynamic Movement Primitives and Locally Weight Regression to encode and learn the motion and impedance required to execute the task. To explore what influences mutation, we varied several control and environment conditions such as the maximum stiffness, robot base position, friction coefficient of the sawed object, and movement period. The results showed that the skill varied over propagation steps and we identified several key aspects of mutation such as movement length, movement offset, and trajectory shape. Based on the results we identified possible benefits (skill mutations useful for different settings or different tasks, and energy efficiency) and dangers (high forces and skill mutations becoming useless for the original task) of the mutation.**

***Index Terms*—Bioinspired robot learning, cooperating robots, physical human-robot interaction, human-robot collaboration, compliance and impedance control.**

Manuscript received 27 March 2023; accepted 12 August 2023. Date of publication 21 August 2023; date of current version 31 August 2023. This letter was recommended for publication by Associate Editor F. Pierri and Editor C. Gosselin upon evaluation of the reviewers' comments. *(Corresponding author: Luka Peternel.)*

The authors are with the Department of Cognitive Robotics, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: rosamaessen@gmail.com; j.m.prendergast@tudelft.nl; l.peternel@tudelft.nl).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2023.3307289, provided by the authors. The video demonstrates the experimental setup and task. The first part of the video shows a novice robot learning collaborative sawing skills from a human. Then, the obtained skill is propagated among robots through a robot-robot learning approach, where the one with the skill is an expert robot and the one without the skill is a novice robot. Various trials with different variations of mutated skills are shown in the subsequent parts of the video.

Digital Object Identifier 10.1109/LRA.2023.3307289

I. Introduction

MODERN manufacturing, logistics and service often involve many robots, whose collaborative tasks and working conditions can change rapidly. Therefore, new robotics skills are needed quickly to enable robots to adapt to these changes and prevent process disruption. Traditional manual programming that is still commonly employed in industry fails in such scenarios due to time-consuming nature of the approach. One of the main alternatives is reinforcement learning, where robots explore new tasks via a trial-and-error approach guided by optimisation functions [1], [2], [3]. These optimisation functions are human-defined and motivate the robot to autonomously improve task performance in an iterative manner. Nevertheless, the approach is still rather slow due to its trial-and-error nature. Furthermore, if not properly constrained, autonomous exploration can result in unsafe behaviour both for the robot and its environment [4].

The other main alternative that overcomes these limitations is Learning from Demonstration, which uses human demonstrations to train robots [5]. The demonstrations are commonly provided by kinesthetic teaching (i.e., physically guiding the robot) [6], [7], [8] or teleoperation (controlling the robot through an interface) [9], [10], [11]. The main advantage of this approach is that the human can demonstrate safe skills directly. However, as the robot learns directly from an expert, the skill is limited by the ability of the expert to perform a task or provide a good demonstration [12].

Learning from Demonstration can also be combined with an exploration-based method like Reinforcement Learning [13], [14], [15], where the human can provide demonstrations and corrective feedback while the robot then improves upon that. However, these studies were limited to single-agent tasks and did not account for direct collaborative tasks. Furthermore, in a scenario where there are many robots and few available human teachers, the skill transfer bandwidth is limited by the amount of possible human involvement. Concrete examples of this are highly automatised factories or warehouse facilities where human workers are fewer compared to robots and act mostly in supervisory roles.

The work in [16] proposed the concept of skill propagation among robots through online physical collaboration. The concept envisioned a Learning from Demonstration approach where the demonstrators are robots themselves instead of humans. The few robots that obtained the skill from the humans can propagate

it among novice robots and when novice robots become experts they can further propagate it among the remaining novice robots. Thus, this approach can quickly spread the needed skills among many robots. However, the results of the proof-of-concept experiment in [16] showed variability in the behaviour of the learning novice robot compared to the expert teacher robot. Therefore, propagation among robots without direct human supervision can result in what we define as a mutation of robotic skills with respect to the original source.

This can be beneficial when better skills might emerge or when a new skill is obtained to be used for other similar tasks. However, it can also be dangerous in terms of task execution safety. While skill mutation could be observed in the learned trajectories of the desired position and stiffness of the robot in [16], the study was limited only to a few propagation steps and did not explore the underlying reasoning for these mutations. It is important to have insight into these mutations to understand the generation of potentially beneficial or dangerous skills. Therefore, there is a fundamental knowledge gap about skill mutations during robot-to-robot skill propagation.

To address this critical knowledge gap, we perform a study to explore the mutation of a skill. This is examined in a scenario when novice robots are learning a physically collaborative task from expert robots online during the task execution. Learning skills online is challenging but it offers more natural interaction with other agents and the environment for real-time skill adjustments. This is especially important for collaborative tasks where the execution also depends on the actions of other agents. In our study, we first examine when and how the skill mutates. We then test how different factors influence the mutations. Finally, we investigate the repeatability of the mutation when using the same conditions. Therefore, the key contribution is new insights into skill mutations during robot-to-robot propagation.

We gain these insights through simulations and experiments using two different types of KUKA LBR iiwa robots (i.e., 7 and 14) performing and learning a collaborative sawing task. During the initial propagation of the skill, one robot is an expert while the other is a novice without the target skill. To teach the novice robot the skill, we extend an online learning approach based on [16]. After learning, the novice is assumed to have become an expert, and this new expert then propagates its learned skill among other novice robots in a similar manner.

II. Methods

The mutation of robotics skills during skill propagation was investigated on a setup involving a collaborative sawing task (Fig. 1). The robots were controlled using a hybrid force/impedance controller, where force control is responsible for maintaining contact with the environment, and impedance control governs the sawing movement and coordination between the agents. At the start, one robot was an expert and the other was a novice. During the task execution, the expert propagated its skill to the novice, who became an expert and could then spread the skill to a new novice (Section II-B). To learn the sawing task, a three-staged learning scheme is used (Section II-C). The skills were encoded using Dynamic Movement Primitives (DMPs) and learned online via Locally Weighted Regression (LWR) (Section II-D).

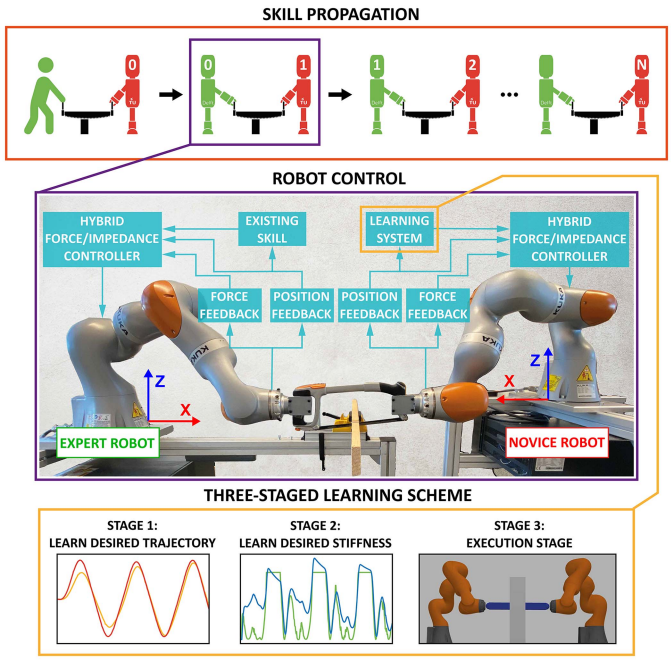

Fig. 1. Skill propagation workflow. An expert robot (green) propagates its skill to a novice robot (red) online during collaborative task execution. Robots were controlled by a hybrid force/impedance controller, which uses an existing skill (for the expert) or a three-staged learning scheme (for the novice). The first stage is used for learning the desired motion, the second for the stiffness, and the final one to execute the task as equal partners.

A. Task and Robot Control

We chose a collaborative task because it is more challenging and offers more variability in terms of multiple agents influencing each other. For this purpose, we use a collaborative sawing task as a use case. The advantage of this task is that the skill consists of a periodic movement, which enables repetitions and the evolution of the mutation is clearly visible. In addition, it incorporates the leader/follower roles that are periodically exchanged in different phases during the execution. When executing the collaborative sawing task, it is desired to have a high stiffness when pulling (leader) and a low stiffness (high compliance) when pushing (follower) not to oppose each other in different stages [17].

We controlled the robots using a hybrid force/impedance controller, which enables simultaneous control of force and motion on different axes. The force controller ensured that the saw maintained contact with the environment while the impedance controller was used to control the movement of the saw. We employed a PI controller to maintain the desired contact force with the object in the vertical (z-axis). The desired force was set to $\boldsymbol{F}_d = -5$ N. The impedance controller was defined as

$$\boldsymbol{F}_{imp} = \boldsymbol{K}\left(\boldsymbol{x}_d - \boldsymbol{x}_a\right) + \boldsymbol{D}\left(\dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}}_a\right), \tag{1}$$

where $\boldsymbol{x}_a$ and $\dot{\boldsymbol{x}}_a$ are the actual position and velocity of the end-effector of the robot respectively, $\boldsymbol{K}$ is the stiffness matrix, and $\boldsymbol{D}$ is the damping matrix. The goal of the learning process (Section II-C) is to learn the desired position $\boldsymbol{x}_d$ and the corresponding stiffness matrix. The damping matrix is defined as a

function of the stiffness matrix to achieve a critically damped system [18].

B. Skill Propagation

The process of skill propagation describes how one agent passes a skill to another agent, who then passes it to the next, and so on. One of these steps is implemented using the three-staged learning scheme, as described in Section II-C. For the sake of a controlled and systematic study, this research implements the steps of skill propagation linearly (robot 1 teaches robot 2, robot 2 teaches robot 3, and so on). However, the reality would look like an exponentially growing tree. If robot 1 has taught robot 2 the task, it will continue teaching robot 3, followed by robot 4, and so on.

To standardise the process, we mathematically defined the skill of the initial expert robot, consisting of the desired motion and corresponding stiffness. Since human sawing movements from past studies exhibit periodic sinusoidal form [16], [19], [20], [21], we define the mathematical model as $x_d(\phi) = \sqrt{\frac{1+n^2}{1+n^2 x_{sin}^2(\phi)}} x_{sin}(\phi) \frac{\Delta x_0}{2}$, where $x_{sin}(\phi) = \sin(\frac{2\pi}{\tau}t)$, $n = 0.5$ is a constant responsible for "flattening out" the sine function, $\tau = 2$ seconds is the period of the signal, and $\Delta x_0 = 0.15$ m is the initial stroke length. The last two values were defined based on the results of [16].

Based on past studies, the human manifest leader/follower behaviour in collaborative sawing in a reciprocal manner [16], [17]. Therefore, we defined the stiffness as a function of the time derivative of the trajectory, the velocity $\dot{x}_d$. When the velocity is negative, a high stiffness ($K_{\max}$) is applied, as the robot should be pulling. When the velocity is positive, the robot should push so the stiffness is set to zero, which means the robot is compliant

$$K = \begin{cases} K_{\max} & \text{if } \dot{x}_d \leq 0, \\ 0 & \text{if } \dot{x}_d > 0. \end{cases} \tag{2}$$

The reference frame for each robot is defined in a way that the x-axis points towards the other robot (see Fig. 1). Equation (2) is from the perspective of a novice robot.

C. Learning Scheme

For the robot to obtain the skill needed for the sawing task, a three-staged learning scheme based on [16] was implemented. See Fig. 1 for an illustration. This method assumes that there are two agents: a novice who has no knowledge of the skill and an expert who is already skilled. In the examined task, the goal was to learn the desired motion and the corresponding stiffness. The desired motion was learned in stage 1 of the learning scheme, and the corresponding stiffness in stage 2. Since the task is periodic, the desired motion and stiffness can be defined as a function of the phase $\phi \in [0, 2\pi]$.

The goal of the *first stage* is to learn the desired motion. During this stage, the expert robot has a high constant stiffness, which allows it to be the demonstrator and thus has complete control of the movement. The novice robot will be fully compliant allowing the expert to guide it. As the two robots are physically coupled through the saw, the movement is transferred between them.

The goal of the *second stage* is to learn the stiffness required for reciprocal leader/follower role coordination. Now the expert assumes the novice knows the desired movement and starts to exchange the stiffening pattern as would be expected in a true collaboration. The novice then infers when it needs to stiffen up by observing whether the actual movement is following the learned desired one: if it does, the novice remains compliant to let the expert lead, if it does not, the novice stiffens up to become the leader. In this research, the desired stiffness K_d is computed using a continuous function as

$$K_d(\phi) = \begin{cases} \left(\frac{|e(\phi)|}{e_{th}}\right)^2 K_{\max} & \text{if } |e(\phi)| < e_{th}, \\ K_{\max} & \text{if } |e(\phi)| \geq e_{th}, \end{cases} \tag{3}$$

$$e(\phi) = x_d(\phi) - x_a(\phi) \tag{4}$$

where $e_{th} = 0.02$ m is the threshold of the error, and $K_{\max}$ the maximum stiffness value. As opposed to the previous work in [16], where the function K_d was discrete, here we made it continuous to improve the approach in terms of smoothness.

The *third stage* is used to demonstrate the learned behaviour in a true collaboration, using the learned trajectory and stiffness obtained in the previous two stages. During this stage, the novice and the expert robot will use their trajectory and stiffness as inputs for their impedance controller.

D. Skill Encoding

We used periodic Dynamic Movement Primitives (DMPs) [22], [23] to encode the skill of the robot, i.e., position and stiffness variables, which are dependent on the given phase of the task. Each variable (position or stiffness) has its own DMP learning system, whose input is the respective measured variable and phase, while the output is the learned phase-dependent variable. The outputs are then used as references for the impedance controller. In order to learn the weights of the DMPs, Locally Weighted Regression (LWR) is used. LWR is suitable for online learning due to its ability to quickly update the model [23]. This method updates the weights using a recursive least-squares method, which is based on the error between the desired trajectory shape and the currently learned trajectory shape. Compared to alternative methods based on global regression, such as Gaussian Process Regression (GPR), LWR is a local regression and in combination with the recursive least-squares update rule it excels in real-time online learning and is thus commonly applied in DMPs [22], [23]. Once learning is complete, the learned weights can be saved such that the DMPs may be used at a different time, for example, during the next skill propagation step. We investigated and defined DMP and LWR parameters through preliminary experiments. The DMP parameters were set as: the number of kernels to 50, the width of kernels to 2.5, gains $\alpha = 8$ and $\beta = 2$. The forgetting factor for LWR was set to 0.995.

III. Experiments & Simulations

A. Setup & Protocol

The setup included KUKA LBR iiwa7 and iiwa14 robots (right and left, respectively, in Fig. 1). Initially, the task was implemented in real-world experiments (see the multimedia

TABLE I
COMPARISON OF THE FACTORS

Factor	Symbol	Baseline value	Value	Same as baseline	Repeatable[1]	Converges
Maximum stiffness	K_{max}	1100 N	500 N	No	Yes	Yes[2]
			5000 N	No	Yes	Yes[2]
Period	τ	2 seconds	1 seconds	No	No	Yes[2]
			3 seconds	No	No	No
Base position offset	-	Both at y = 0.4m	Both at y = 0.0 m	No	Yes	No
			y = 0.4 m / y = -0.4 m	Yes	Yes	Yes[2]
			y = 0.4 m / y = -0.5 m	Yes	Yes	Yes[2]
Friction coef.	μ	0.0	0.05	No	Yes	Yes
			0.01	No	Yes	Yes[2]
			0.005	No	Yes	Yes[2]

[1] Repeatable is set to 'Yes' if the main trends of the different trials are similar. As will be discussed in Section IV-C, the mutations are never identical throughout the different trials, but often their main trends are repeatable.
[2] This mutation converges due to approaching some limit, i.e., an environmental boundary, a joint limit, or a torque limit.

material for a video). However, when considering a large number of trials (that include many propagation steps), different combinations of parameter settings, and a number of environmental conditions to be explored, it became clear that real-world experiments would be infeasible for a study of this scope. Furthermore, the amount of wood needed for the sawing would be enormous and unsustainable. Finally, since the mutations during robot-to-robot skill propagation have not yet been explored, the resulting behaviour can be potentially dangerous. Therefore, we resorted to simulations where many trials and combinations necessary for this study could be obtained in a reasonable time frame and sustainable and safe manner. An additional advantage of using a simulation is that many confounding factors, such as environmental conditions, can be fixed to make trials reproducible. This approach allows for a more controlled, repeatable and systematic investigation of the influence of these factors on the mutations. We used *Gazebo* to simulate the robots interacting with their environment. The robot control and communication were implemented via *ROS*.

The sawing axis was defined to be along the x-axis of the robot base frame. The base frame's x-axis of each robot was pointing toward the other robot. The object that was to be cut had a size of $0.565 \times 1.0 \times 0.1$ m, while the baseline saw length was 0.45 m and mass was 0.5 kg. The movement of the saw along the y-axis was constrained by the groove of the incision on both sides. The gravity vector was along the z-axis of the robot base frame.

We conducted numerous experiments to investigate the mutation during the robot-to-robot skill propagation. One three-stage skill transfer using the learning scheme described in Section II-C is referred to as a *run*. Each stage of the learning took 10 seconds, resulting in a total time of 30 seconds per run. The entire skill propagation process between 30 robots (robot 1 teaches robot 2,... robot 29 teaches robot 30) that results in 30 runs is called a *trial*. In this research, we conducted 5 trials per experimental setting defined by factors presented in Table I. The experiments are split into two groups: the group using the baseline setting of the factors, and the group with changed values for the factors. To examine them systematically and in an isolated manner, each factor was changed individually to investigate whether changing these values would influence the mutation, and if so, how. This meant that each factor would be set to its baseline value during this group of experiments, except the one being investigated. The baseline setting is used for two purposes: as a baseline, which is used to compare the influence of the changed factors, and to check whether the mutation is repeatable.

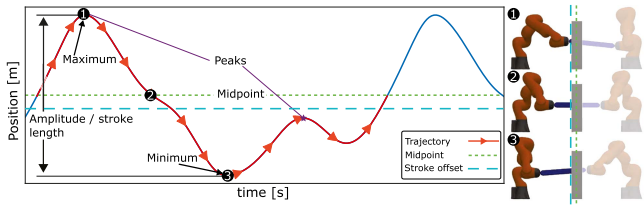

Fig. 2. Graphical representation of the used metrics. The red line represents one period of end-effector motion trajectory in the x-axis as measured by the novice robot (i.e., the highlighted robot on the left). The metrics are defined as follows: amplitude/stroke length (difference between the global maximum and minimum value), travelled distance (total distance of trajectory), the midpoint of movement (between maximum and minimum value), average value/stroke offset, and the number of peaks (local maximums).

B. Metrics

To analyse the properties of mutations in the propagated robot skill (motion and stiffness) and its resulting behaviour, we used several metrics, which are graphically depicted in Fig. 2. As we are interested in the evolution of the skill throughout the different runs, we defined these metrics as a function of the different runs. To do so, we computed the value per metric over the span of one period. Since each experiment consists of 5 trials, the results are shown as the average of the five trials (per run) and their standard deviation. This additionally allows for the investigation of the repeatability of the skill, with a low standard deviation indicating that the main trends occurring in the different mutations are similar.

Peak-to-peak amplitude: The difference between the global maximum and minimum of a signal over one period is called the peak-to-peak amplitude: $|\max(x(\phi)) - \min(x(\phi))|$. For motion, this metric is referred to as *stroke length*.

Travelled distance: The distance travelled by the end-effector of the robot during one period $\int_0^{2\pi} |\delta x(\phi) d\phi|$. This metric is only applicable to motion and not stiffness.

Midpoint of the movement: The point halfway between the global maximum and minimum: $\frac{\max(x(\phi))+\min(x(\phi))}{2}$. This metric is only applicable to motion and not stiffness. In the case of the baseline setting, the bases of the iiwa7 and iiwa14 (right and left, respectively, in Fig. 1) are located at $x = -0.7$ m and

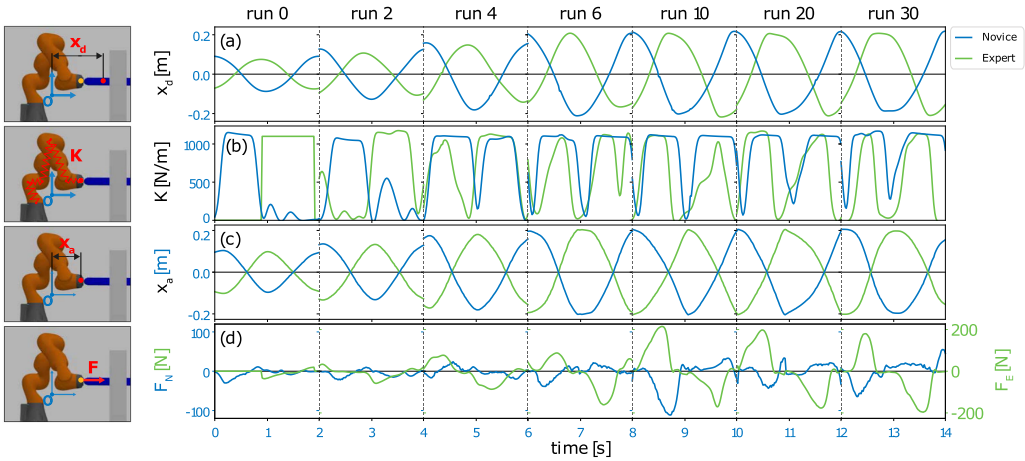

Fig. 3. Skill evolution of the expert and novice robots showed using one period of stage 3 of the learning scheme for different runs throughout one trial. On the left, four images describing the different metrics have been visualised. In the graph, from left to right, different runs are shown from one trial. Note here that the runs are not linearly spaced from 0–30. Each of the graphs (a)–(d) show both the results of the novice (blue) and expert (green). From top to bottom: (a) Visualises the desired motion x_d, here means a high value that the end-effector of the robot is further away from the robot's origin. (b) Shows the stiffness. (c) Illustrates the actual motion x_a of the robots. (d) Shows the forces sent to the robot, with the left axis showing the forces of the novice robot and the right axis showing the forces of the expert robot.

$x = 0.7$ m, respectively, in the global frame located midway between the base frames. A value lower and higher than 0.0 m indicates that the overall movement shifted closer to the iiwa7 and iiwa14, respectively.

Average value of the data: $\frac{\int_0^{2\pi} x(\phi)d\phi}{t_{2\pi}-t_0}$. For motion, this metric is referred to as the *stroke offset*.

Number of peaks: Determined by counting the local maximums in the signal and gives an indication of the data complexity. Note that for stiffness, a flattened peak is often found where the maximum stiffness was reached. These "flattened" peaks are counted as only one.

IV. Results

We tested the variation of ten factors and their effect on skill mutations, which were initially identified through a combination of a literature survey and preliminary experiments. Four factors showed visible differences compared to the baseline setting and are thus closely examined in the results. These factors are the maximum stiffness, the length of the period, the robot base position offset, and the friction coefficient (see Table I for the baseline and varied settings). These factors can be generalised to other tasks that involve periodic co-manipulation with physical interaction. The other factors were: initial stroke length, robot base position distance, saw length, saw mass, robot hardware (alternating iiwa7 and iiwa14 between runs or keeping them fixed), and error threshold (from (3)). Some are task-specific (e.g., initial stroke length), while others can be generalised to other co-manipulation tasks, objects and scenarios (e.g., length and mass are relevant in other co-manipulated objects). The error threshold is learning method-specific but may be generalised to other position-tracking tasks. In terms of results structure, we first examine mutations for the baseline setting, using the previously discussed metrics, to give an intuitive starting point. The analysis of variations in settings is structured based on mutation patterns rather than individual factors. This is because

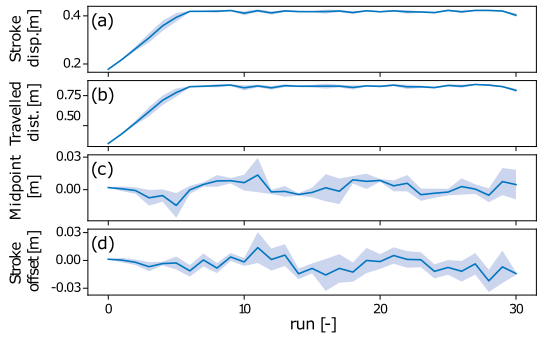

Fig. 4. Analysis of the desired motion learned by the novice robot in stage 1 of the learning scheme. Different metrics are used for this analysis. (a)–(d) Illustrate the stroke length, the travelled distance, the midpoint of the movement, and the stroke offset, respectively.

often different tested factors resulted in similar patterns. The patterns discussed are the influence of the phase lag, overshoot on the desired motion, effect of the joints states, and approaching torque limits. In the analysis, we look at both the mutations in the encoded skills (desired motion and stiffness) as well as the behaviour that results from it (measured actual motion).

A. Baseline Setting

We will first examine the mutation of the skill for the baseline setting as defined in Table I. A quantitative analysis of the mutation in the skill of both the expert and novice robot is visualised in Fig. 3. This graph shows one period of the learned skill (stage 3) for multiple runs of one trial.

By observing Fig. 3(a), we can see that the stroke length of the desired motion x_d, as learned in stage 1, increased from run 0 to run 6, after which it plateaued. This plateau is confirmed when looking at the analysis of the stroke length in Fig. 4(a). The plateau of the stroke length resulted from reaching the environment's boundaries, i.e., the inner parts of the saw handle on both sides began to touch the object it is sawing. This limit

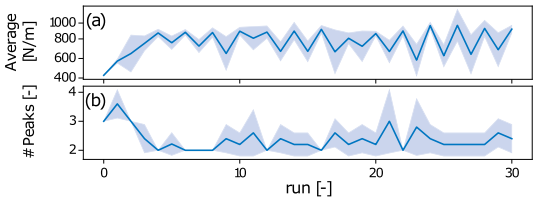

Fig. 5. Analysis of the stiffness learned by the novice robot in stage 2. Different metrics are used for this analysis. (a) Shows the average value of the stiffness. (b) Shows the number of peaks per period.

was at 0.45 m, which is equal to the length of the saw. There is a similar pattern visible for the distance travelled (Fig. 4(b)). Based on these metrics, we could assume that the desired motion is smooth with only one peak. This is substantiated by looking at Fig. 3(a). The results of the midpoint and the stroke offset of the desired motion (Fig. 4(c) and (d)) show similar patterns. Based on this and the results in Fig. 3, it can be concluded that the general motion characteristics in one direction are similar to the other direction. If these were different, the stroke offset would be shifted compared to the midpoint of the movement.

By analysing the stiffness learned by the novice in stage 2 (Fig. 5), we see that the stiffness's average value increases throughout the different runs. The raw data showed that in the later runs, the error between the desired and actual position of the robot increased. As the computed stiffness depends on this error (3), the stiffness was often set to its maximum value. Therefore, peaks only occurred when the desired motion crossed the actual motion, which for most cases was only twice. The complexity of the stiffness will be further examined in the following sections that present mutation patterns occurring when settings were varied from the baseline.

B. Factor Variations

While Table I shows a more general overview of skill mutations based on different settings and factors, we will also more closely examine several concrete cases. Fig. 6(a) shows learned desired motion and actual motion for different parameter settings. Here we define "phase lag" property used in the analysis, which is a shift in the desired motion pattern compared to the actual motion pattern along the phase. For example, the peak motion when the saw reaches one end might be shifted along the phase during the runs. The phase lag is larger for the low maximum stiffness setting $K_{\max} = 500$ N (Fig. 6(a.2)) compared to the baseline setting (Fig. 6(a.1)), which is natural since lower stiffness produces less force and thus desired motion following is less strict. It is interesting that both show an overall increase in the stroke length throughout the different runs (Fig. 6(b.1)) and as a result of the phase lag difference, the increase over runs happens faster in the low stiffness case compared to the baseline case.

While the low stiffness case exhibits some differences in mutation compared to the baseline case, the faster movement speed/frequency case ($\tau = 1$ s) has much bigger effects on the mutation. By observing Fig. 6(a.3), we can see a larger phase lag even when stiffness is the same as in the baseline case. This phase lag resulted in a drift in the stroke offset from run 4 to 30 (Fig. 6(b)). This indicates that the expert robot always

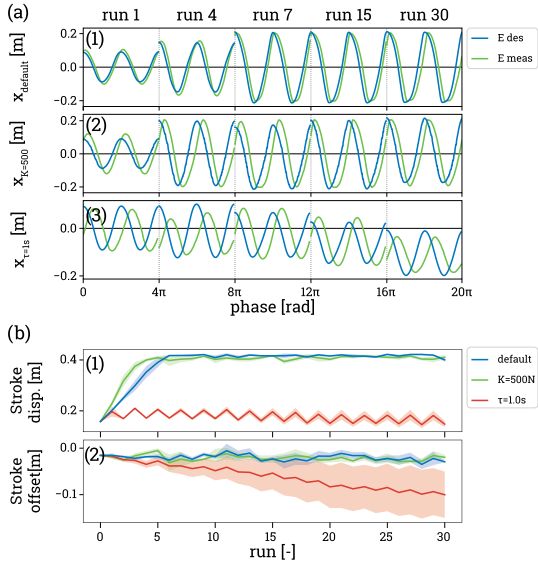

Fig. 6. Influence of the phase lag on the mutation of the learned motion. (a) Shows from left to right the desired and actual motion of the expert during the second and third periods of stage 1 of different runs. From top to bottom, the baseline setup (1), a setup where the maximum stiffness was changed to $K_{\max} = 500$ N (2), and a setup where the period was set to $\tau = 1$ s is visualised (3). (b) Shows the evolution of the stroke length (1) and the stroke offset of the learned trajectory of the novice robot during the different runs (2).

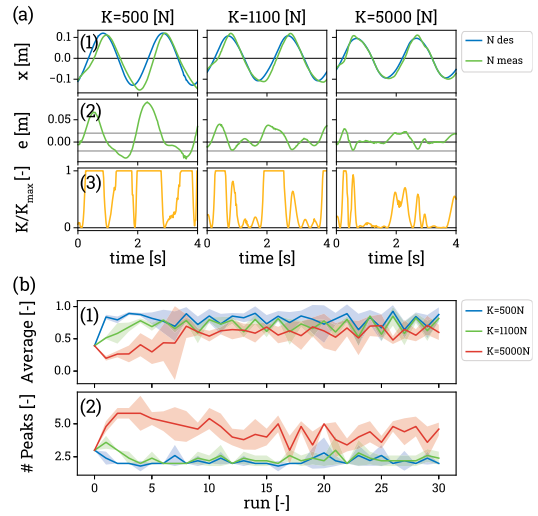

Fig. 7. Influence of the phase lag on the mutation of the learned stiffness. (a) Shows two periods of the actual and desired motion of the novice robot in stage 2 and its normalized desired stiffness (stiffness divided by maximum stiffness). This is shown for different maximum stiffness values during run 8. (b) Shows the evolution, throughout the different runs, of the normalized average value and the number of peaks of the learned stiffness of the novice robot during one period.

moves the entire movement slightly closer to its base position. Furthermore, when the iiwa7 robot is the expert, the entire movement shifts more towards its base than when the iiwa14 is the expert. However, unlike for the baseline and low stiffness cases, the stroke length remained relatively constant and did not mutate for the high-frequency case.

Besides the trajectory, the phase lag also influences the learned stiffness in stage 2. Fig. 7 shows results for three different values of stiffness where the different stiffness values were normalised

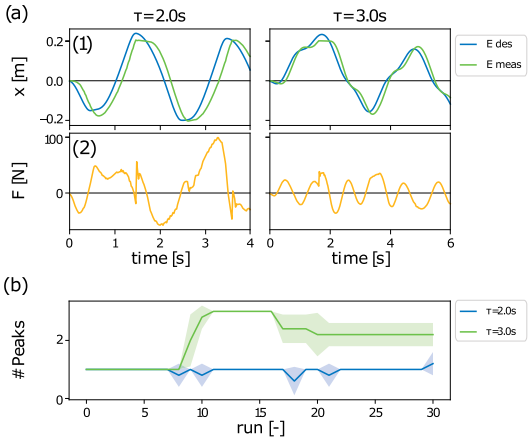

Fig. 8. Influence of overshoot on the desired motion on the actual motion. (a) Shows an example of the results of two phases of stage 1 of the desired and actual motion (1) and its corresponding force along the x-axis (2) for the expert robot using two different periods ($\tau = 2$ s and $\tau = 3$ s). (b) Shows the complexity of the trajectories learned by the novice in different runs.

to 1 for easier comparison. When the stiffness was low ($K_{\max}$ = 500 N), the phase lag increased, resulting in a larger error between the desired and actual motion than for the cases with higher stiffness. This meant that the transitions between stiff and compliant behaviour according to the stiffness learning rule in (3) were smoother in the lower stiffness case than in the higher stiffness case. To confirm this, Fig. 7 shows fewer peaks in the stiffness trajectory for the lower stiffness settings, where the complexity of the trajectory increased with increased peaks for the higher stiffness setting.

Fig. 8(a) shows the variation of sawing speed: a shorter (τ = 2 seconds) and a longer period (τ = 3 seconds). For the shorter period (baseline setting) the resulting robot force is quite natural: negative while pulling (moving in the negative direction) and positive while pushing (trajectory moving in the positive direction). This is not the case for the case with a longer period (lower sawing speed). The small error between the desired and actual motion often results in the expert robot overshooting the desired motion. The robot compensates for this overshoot by applying a counterforce, resulting in a rougher force signal, i.e., more peaks. As the novice learns from this measured actual motion, a trajectory including these irregularities will be learned. The extent of these irregularities increases throughout the different runs, which is shown by analysing the peaks of the learned trajectory (Fig. 8(b)). This indicates that the trajectory becomes rougher throughout the different runs. A benefit of this mutation can be found when looking at energy efficiency. This efficiency is computed using the work $W = \int K(x_d - x_a)dx$. As the examples provided in Fig. 8(a) have different periods, we decided to compare them by computing the work per period. The results for τ = 2 seconds is 33.3 kJ/period, whereas it was 11.3 kJ/period for τ = 3 seconds. These results indicate a trade-off between the smoothness (low complexity) of the trajectory and its energy efficiency.

C. Repeatability

We also investigated the repeatability of the mutation by comparing the same runs of different trials (baseline setting).

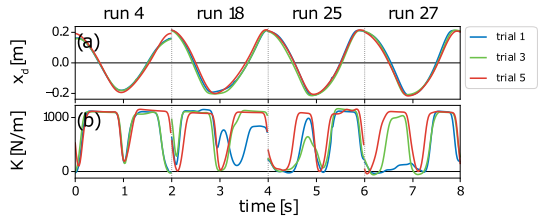

Fig. 9. One period of the skill learned by the novice robot during multiple (random) runs of different trials is shown. From left to right, four different runs are visualized. The upper graphs (a) show the learned desired motion and the bottom graphs and (b) show the learned stiffness.

The results are shown in Fig. 9. The repeatability of the mutation is defined as the ability of the robot to reproduce the same skill (desired motion and stiffness), during each trial. The desired motion of the different trials exhibits minimal variability and is thus very repeatable. On the other hand, the stiffness mutations are much less repeatable. An example of this is provided in trial 1 of run 18 (Fig. 9(b)), where the characteristics of the learned stiffness were completely different compared to the other trials. However, for most of the other cases, we often only identified two peaks. The analysis of the stiffness, in Fig. 5, does show similar trends for the stiffness, as the standard deviations are relatively small, again indicating a similar trend for the different trials.

V. Discussion

The testing of different settings revealed both benefits and risks associated with the mutations. An example of potential risk can be observed in Fig. 3 where in some cases, the mutated skill produced relatively higher forces. Though higher force may be needed in different conditions (e.g., heavier tool, stronger material, etc.), and thus can also be beneficial. An example of potential benefit can be observed in Figs. 3 and 4(a), where the movement trajectory significantly increased its stroke length after propagation compared to the original run. The increase in stroke length could be beneficial when using a longer saw or when cutting a larger object. The increased stroke length also benefited in terms of force manipulability (i.e., a better capacity to transform joint torques into endpoint forces). As the range of the movement increased, the point where the saw switched direction, from now on referred to as the switch-over point, moved further away from the base of the robot that had to start pulling. Therefore, the force manipulability of the robot at this switch-over point was much higher, producing more endpoint force in the sawing axis for the same joint torques. Having high force manipulability at the switch-over point is helpful since the inertial and stiction forces needed to overcome due to the switching direction are the highest. Therefore, this mutation optimised the actions in terms of the force manipulability of both robots.

The result in Fig. 6(b) showed a case when mutation caused movement of the offset to drift toward one of the robots. In this case, one robot is benefiting from an improved force manipulability (the one the offset drifts away from) and the other is being disadvantaged (the one the offset drifts toward). While this can be unwanted when two similar robots are used, it can be beneficial when we have one strong and one weak robot and the

disproportional manipulability may help them balance out their strengths and prevent the weak one from being overpowered.

In some cases, the stiffness mutated from a reciprocal exchange pattern to almost constantly stiff behaviour (e.g., Fig. 6(a) left graph). While this is not very useful for collaborative tasks where reciprocally exchanging effort is required, it may be a useful starting skill when executing collaborative tasks requiring mirrored behaviour [17], or in adapting the task from collaborative to single-agent where the robot must be stiff all the time.

We also observed cases where the alternations of the characteristics of the skill were much larger leading to more complex behaviours (e.g., see Fig. 8). While the mutated behaviour was less smooth than the original in terms of measured forces, the task execution was not significantly affected. However, the rougher behaviour was more energy efficient as the work exerted by the robot was less: 33.3 kJ/period for smooth compared to 11.3 kJ/period for rough, as was already mentioned in the results. Besides energy efficiency, there is also a task-related tradeoff between smoothness or roughness of behaviour. For some delicate tasks, rough interaction is unwanted or even dangerous, and thus such skill would not be beneficial. However, other tasks may benefit from rough interaction, such as polishing off hard stains from a surface.

Finally, other variables such as forces may be affected by mutations of skills. In the particular example shown in Fig. 3, the force peaks gradually increased on both robots roughly equally in the initial runs and stabilised in run 10. However, in subsequent runs, novice skill further mutated and its forces decreased with respect to expert. This may be due to mutation exhibiting inclination of the saw angle towards the novice robot and thus reducing its pulling efforts.

In summary, this study identified several key aspects of skill mutation such as movement length, movement offset, and trajectory characteristics. Based on the results we identified possible benefits (skill useful for different settings or different tasks, energy efficiency, improvement of manipulability) and dangers (high forces and skill becoming useless for the original task) of the mutation. However, there are some limitations of this study that open possibilities for future work. While the sawing task is an excellent case study as it provides many aspects like motion, stiffness, coordination, and physical interaction with both environment and another agent, exploring other tasks would be interesting as well. Furthermore, if many robots and experimental resources are available over a prolonged time, it would be interesting to repeat the study fully in a real setup.

REFERENCES

[1] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1238–1274, 2013.

[2] A. S. Polydoros and L. Nalpantidis, "Survey of model-based reinforcement learning: Applications on robotics," *J. Intell. Robot. Syst.*, vol. 86, no. 2, pp. 153–173, 2017.

[3] K. Chatzilygeroudis, V. Vassiliades, F. Stulp, S. Calinon, and J.-B. Mouret, "A survey on policy search algorithms for learning robot controllers in a handful of trials," *IEEE Trans. Robot.*, vol. 36, no. 2, pp. 328–347, Apr. 2020.

[4] F. Berkenkamp, M. Turchetta, A. Schoellig, and A. Krause, "Safe model-based reinforcement learning with stability guarantees," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 909–919.

[5] A. G. Billard, S. Calinon, and R. Dillmann, "Learning from humans," in *Springer Handbook of Robotics*, Berlin, Germany: Springer, 2016, pp. 1995–2014.

[6] G. J. Maeda, G. Neumann, M. Ewerton, R. Lioutikov, O. Kroemer, and J. Peters, "Probabilistic movement primitives for coordination of multiple human–robot collaborative tasks," *Auton. Robots*, vol. 41, pp. 593–612, 2017.

[7] R. Caccavale, M. Saveriano, A. Finzi, and D. Lee, "Kinesthetic teaching and attentional supervision of structured tasks in human–robot interaction," *Auton. Robots*, vol. 43, pp. 1291–1307, 2019.

[8] F. Steinmetz, V. Nitsch, and F. Stulp, "Intuitive task-level programming by demonstration through semantic skill recognition," *IEEE Robot. Automat. Lett.*, vol. 4, no. 4, pp. 3742–3749, Oct. 2019.

[9] L. Peternel, T. Petrič, and J. Babič, "Robotic assembly solution by human-in-the-loop teaching method based on real-time stiffness modulation," *Auton. Robots*, vol. 42, no. 1, pp. 1–17, 2018.

[10] M. J. Zeestraten, I. Havoutis, and S. Calinon, "Programming by demonstration for shared control with an application in teleoperation," *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 1848–1855, Jul. 2018.

[11] A. Pervez, H. Latifee, J.-H. Ryu, and D. Lee, "Motion encoding with asynchronous trajectories of repetitive teleoperation tasks and its extension to human-agent shared teleoperation," *Auton. Robots*, vol. 43, pp. 2055–2069, 2019.

[12] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Ann. Rev. Control, Robot., Auton. Syst.*, vol. 3, pp. 297–330, 2020.

[13] V. Chu, T. Fitzgerald, and A. L. Thomaz, "Learning object affordances by leveraging the combination of human-guidance and self-exploration," in *Proc. IEEE/ACM Int. Conf. Hum.-Robot Interact.*, 2016, pp. 221–228.

[14] C. Celemin, G. Maeda, J. Ruiz-del Solar, J. Peters, and J. Kober, "Reinforcement learning of motor skills using policy search and human corrective advice," *Int. J. Robot. Res.*, vol. 38, no. 14, pp. 1560–1580, 2019.

[15] E. Chisari, T. Welschehold, J. Boedecker, W. Burgard, and A. Valada, "Correct me if i am wrong: Interactive learning for robotic manipulation," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 3695–3702, Apr. 2022.

[16] L. Peternel and A. Ajoudani, "Robots learning from robots: A proof of concept study for co-manipulation tasks," in *Proc. IEEE-RAS Int. Conf. Humanoid Robot.*, 2017, pp. 484–490.

[17] L. Peternel, N. Tsagarakis, and A. Ajoudani, "A human–robot co-manipulation approach based on human sensorimotor information," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 7, pp. 811–822, Jul. 2017.

[18] A. Albu-Schaffer, C. Ott, U. Frese, and G. Hirzinger, "Cartesian impedance control of redundant robots: Recent results with the DLR-light-weight-arms," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2003, pp. 3704–3709.

[19] L. Peternel, L. Rozo, D. Caldwell, and A. Ajoudani, "A method for derivation of robot task-frame control authority from repeated sensory observations," *IEEE Robot. Automat. Lett.*, vol. 2, no. 2, pp. 719–726, Apr. 2017.

[20] X. Chen, N. Wang, H. Cheng, and C. Yang, "Neural learning enhanced variable admittance control for human–robot collaboration," *IEEE Access*, vol. 8, pp. 25727–25737, 2020.

[21] E. Zheng, Y. Li, Z. Zhao, Q. Wang, and H. Qiao, "An electrical impedance tomography based interface for human–robot collaboration," *IEEE/ASME Trans. Mechatron.*, vol. 26, no. 5, pp. 2373–2384, Oct. 2021.

[22] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: Learning attractor models for motor behaviors," *Neural Comput.*, vol. 25, no. 2, pp. 328–373, 2013.

[23] M. Saveriano, F. J. Abu-Dakka, A. Kramberger, and L. Peternel, "Dynamic movement primitives in robotics: A tutorial survey," *Int. J. Robot. Res.*, 2023.