Title: Investigating Strategies Enabling Novice Users to Teach Plannable Hierarchical Tasks to Robots

Authors: Nina Moorman[1], Aman Singh[1], Manisha Natarajan[1], Erin Hedlund-Botti[1], Mariah Schrum[2], Chuxuan Yang[1], Lakshmi Seelam[1], Matthew C. Gombolay[1], and Nakul Gopalan[3]

Funding

- Konica Minolta, Inc.: Gift
- Lincoln Laboratory, Massachusetts Institute of Technology: 7000437192
- NASA Early Career Fellowship: 80HQTR19NOA01-19ECF-B1
- National Science Foundation: 20-604, IIS-2112633, IIS-2340177
- Air Force Office of Scientific Research: FA9550-24-1-0239

Conflicts of Interest:
gatech.edu, tri.global, berkeley.edu, jhu.edu, mit.edu, harvard.edu, asu.edu

Corresponding Author:
Nina Moorman,
Department of Interactive Computing,
Georgia Institute of Technology
Atlanta, GA, USA
nmoorman3@gatech.edu

[1]Georgia Institute of Technology, Atlanta, USA
[2]University of California, Berkeley, USA
[3]Arizona State University, Tempe, USA

Investigating Strategies Enabling Novice Users to Teach Plannable Hierarchical Tasks to Robots

International Journal of Robotics Research
XX(X):2–??
©The Author(s) 2023
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Nina Moorman[1], Aman Singh[1], Manisha Natarajan[1], Erin Hedlund-Botti[1], Mariah Schrum[2], Chuxuan Yang[1], Lakshmi Seelam[1], Matthew C. Gombolay[1], and Nakul Gopalan[3]

Abstract

Learning from demonstration (LfD) seeks to democratize robotics by enabling non-experts to intuitively program robots to perform novel skills through human task demonstration. Yet, LfD is challenging under a task and motion planning (TAMP) setting, as solving long-horizon manipulation tasks requires the use of hierarchical abstractions. Prior work has studied mechanisms for eliciting demonstrations that include hierarchical specifications for robotics applications but has not examined whether non-roboticist end-users are capable of providing such hierarchical demonstrations without explicit training from a roboticist for each task. We characterize whether, how, and which users can do so. Finding that the result is negative, we develop a series of training domains that successfully enable users to provide demonstrations that exhibit hierarchical abstractions. Our first experiment shows that fewer than half (35.71%) of our subjects provide demonstrations with hierarchical abstractions when not primed. Our second experiment demonstrates that users fail to teach the robot with adequately detailed TAMP abstractions, when not shown a video demonstration of an expert's teaching strategy. Our experiments reveal the need for fundamentally different approaches in LfD to enable end-users to teach robots generalizable long-horizon tasks without being coached by experts at every step. Toward this goal, we developed and evaluated a set of TAMP domains for LfD in a third study. Positively, we find that experience obtained in different, training domains enables users to provide demonstrations with useful, plannable abstractions on new, test domains just as well as providing a video prescribing an expert's teaching strategy in the new domain.

Keywords

Learning from Demonstration, Task and Motion Planning, Robot Learning

Introduction

People exhibit the ability to learn to solve long-horizon tasks – from learning to knit a sweater to learning to cook a new recipe. These tasks can take from hours to days, but humans still learn to sequence from smaller details, like movements of the fingers, to higher level details, like completing sections of a sweater to creating a sauce for a new recipe. Additionally, these tasks are multi-task, humans can learn to cook different types of soups or knit different types of sweaters. The field of robot learning from demonstration (LfD) seeks to enable robots to exhibit the human ability to learn from end-user demonstration and scale the power of robotics. Yet, to this day, robots do not have a general-purpose ability to learn novel multi-task long-horizon tasks from demonstrations, despite significant prior work (Argall, Chernova, Veloso and Browning 2009). In this work, instead of proposing algorithmic solutions for robot learning we investigate different strategies with which people can *teach* long-horizon, multi-task chores to robots. The goal here is to identify strategies that help users teach tasks to robots, which can inform future LfD algorithms that expect human interactions when teaching robots.

Substantial emphasis in LfD has been placed on teaching robots single, short-horizon *skills*, such as picking up or making contact with an object (Haldar, Mathur, Yarats and Pinto 2022; Ijspeert, Nakanishi, Hoffmann, Pastor and Schaal 2013; Konidaris and Barto 2009; Levine, Finn, Darrell and Abbeel 2016). However, there is a lack of work enabling robots to learn long-horizon *tasks*, such as learning in-home assistive tasks or manufacturing process assembly operations, from human demonstrations. Such tasks can be considered multi-task problems. For example, setting a dinner table would require a robot to set multiple place settings, dependent on the number of guests, where each table setting consists of multiple objects that each require a different manipulation procedure. A demonstrator cannot be expected to provide demonstrations for each task specification of these multi-tasks, such as for each possible number of plate settings. These multi-task scenarios might require the users to teach tasks with multiple skills or multiple subtasks which can be repeated in different sequence ordering. An open question here is what abstractions should a teacher use to teach an agent? For example, there are different types of abstractions an agent can use sub-tasks, skills, options, sub-goals, plan hierarchies, etc. In this work, we restrict ourselves to a robotics scenario and

[1]Georgia Institute of Technology, Atlanta, USA
[2]University of California, Berkeley, USA
[3]Arizona State University, Tempe, USA

Corresponding author:
Nina Moorman, Georgia Institute of Technology Atlanta, USA.
Email: nmoorman3@gatech.edu

choose Task and Motion Planning (TAMP) based abstractions as they are a popular choice of abstractions for robots to solve long-horizon, multi-task problem scenarios (Garrett, Chitnis, Holladay, Kim, Silver, Kaelbling and Lozano-Pérez 2021; Srivastava, Fang, Riano, Chitnis, Russell and Abbeel 2014; Hauser and Latombe 2010).

Prior work in LfD has shown that users can provide demonstrations at some level of abstractions, such as keyframes (Akgun, Cakmak, Jiang and Thomaz 2012) or sub-task specifications for a hierarchical task network (Mohseni-Kabir, Rich, Chernova, Sidner and Miller 2015). However, in these studies, participants were given *explicit* instructions on how to teach the robot *each and every* task. Such an approach is untenable for scaling up to a vision of ubiquitous robotics, as it is impractical for experts to teach every end-user how to program robots each and every desired task. Instead, we examine how people teach novel tasks to robots in the absence of a roboticist's explicit tutelage.

Our major contributions are the following. We first develop a human-subjects experiment to investigate whether people naturally teach using abstractions without explicit priming*. Next, we test the efficacy of different modes of end-user instructions and feedback in soliciting robot demonstrations exhibiting hierarchical abstractions. Finally, we measure the effect of accumulating robot teaching experience in different, training domains on the degree of hierarchical abstractions exhibited in user demonstrations in a novel domain. We develop design guidelines to enable users to generalize their training towards teaching novel tasks without needing an expert to show them how. In other words, instead of training demonstrators from scratch in each domain users encounter, we train users such that the knowledge the demonstrator learns about providing demonstrations in one domain is transferable to novel domains. This is a positive result for the democratization of robot LfD.

Our work's contributions include the following:

- We first conduct a $n = 28$ user study to demonstrate that the majority of the participants (64.29%) do not naturally teach tasks to robots with any type of abstractions. They can be induced with different strategies to provide abstractions ($p < 0.05$).

- In a second experiment ($n = 24$), we find that - relative to participants primed with text-based instructions on an analog task, an expert demonstration on an analog task, and a debugging demonstration in the current task - only a video presenting an optimal teaching strategy in the current task allows 100% of the participants to create necessary and sufficient abstractions. This study demonstrates the challenge of eliciting task-specific abstraction-based demonstrations from users without showing them the exact expert solution.

- Our third multi-domain study, $n = 28$, we develop and evaluate novel LfD training domains to test the role of experience in teaching robots to perform new, test tasks. As novice users teach tasks across multiple domains and gain teaching experience, we find that participants are better able to generalize knowledge about teaching TAMP abstraction from one domain to the next in their training sequence zero shot ($p < .001$). Further, teaching efficiency ($p < .001$) increases and redundancy ($p < .05$) decreases in novel domains.

- Our post-analysis demonstrates that after obtaining experience teaching robots in different domains the quality of abstractions provided by users is indistinguishable from that of users who are prescribed how to provide demonstrations in the same domain from an expert video ($p < .001$). This finding suggests that, instead of prescribing how users should provide the robot demonstrators in all possible domains from experts, users can learn how to provide abstracted demonstrations by practicing providing demonstrations in a few training domains.

- Finally, we obtain feedback from the robotics community at large to inform our work's limitations and avenues of future work. We believe such feedback can inform future research questions and collaborations between the human-robot interaction and task and motion planning communities within robotics.

Background

In this section, we define terms pertaining to our work. We will then introduce our studies and discuss them one after another in the following sections.

Multi-task problems – In multi-task problems, the objects that the robot interacts with remain the same. However, the number of objects, their locations, or the order in which the robot interacts with the objects changes between sub-tasks (Caruana 1998). This is often accomplished by leveraging similarities between the sub-tasks (Zhang and Yang 2018).

Multi-modal tasks – A mode is a sub-manifold of robot motion within which the robot's contact specification, with respect to different objects in the world, remains constant (Alami, Siméon and Laumond 1991; Alami, Laumond and Siméon 1995; Hauser and Latombe 2010; Hauser and Ng-Thow-Hing 2011). A multi-modal task is one where the robot transitions between at least two modes to solve a task. For example, to pick up a block, the robot first is restricted to a mode where all its motion is confined to a sub-manifold within which the robot's gripper is not in contact with any object. After picking up the block, the mode of the robot is the sub-manifold within which it is in continuous contact with the block. Similarly, a *Long Horizon Task* is a task where the robot needs to perform multiple mode switches to solve the task. Thus, per our definition, multi-modal tasks have at least one mode switch (≥ 1), and long-horizon tasks have several (> 2) mode switches. In our work, the robot is solving multi-modal tasks.

Task and Motion Planning (TAMP) – Robotics problems require an interplay between symbolic and continuous domains. For example, to pass medicines to a patient, the robot needs to make a high-level symbolic plan to know which boxes of medicines to pick up and pass to a patient. This plan and its corresponding state are symbolic and discrete over the type and quantities of medicine box objects required.

*We employ the following definition of priming: "to tell someone something that will prepare them for a particular situation" (Prime. 2024).

However, to pick up a box the robot needs to create a continuous motion plan without collisions such that the box is in the robot's hand. This motion planning problem occurs over the continuous state of the robot's joints. Such problems, that exhibit an interplay between symbolic and continuous plans, are TAMP problems. We choose to define our domains as TAMP problems, as they require an interplay between symbolic goal states and continuous motion from the robot.

Sub-task based abstraction – Transitions between the symbolic states of a TAMP problem are called sub-tasks[†] (Garrett, Chitnis, Holladay, Kim, Silver, Kaelbling and Lozano-Pérez 2021). For example, when a robot moves to pick up a cup, the state of the world transitions symbolically, such that the cup is in the robot's hand. In TAMP formulations, the sub-tasks are described by preconditions (`pre`) and effects (`eff`), as well as constraints (`con`) that must hold for all continuous actions for the duration of time the action is being taken. We provide a sample mathematical TAMP formulation for the sub-tasks of the medicine dispensing domain in the Appendix.

Sufficient sub-tasks – In our domains, a sub-task is deemed *sufficient* if the sub-task changes the symbolic state of the world and results in at most one mode change. For a sub-task to change the symbolic state of the world, the change must go beyond a negligible change in the robot's pose. Moreover, limiting the sub-task to at most one mode change ensures that the robot can change its interaction with only one object within the sub-task. Such design of sub-tasks ensures that a sub-task transition affects only a small set of symbolic state variables at a time. These sub-tasks can then be sequenced by a task planner to reach a larger set of the symbolic state space, allowing maximal generalizability in the tasks that can be solved within the domain.

Redundant sub-tasks – A sub-task is deemed redundant if its goal can be met by another sufficient sub-task or a combination of sufficient sub-tasks previously taught. Sub-task redundancy is defined with respect to a given set of demonstrations being taught.

Necessary sub-tasks – Similarly, a sub-task is deemed necessary if its goal can *not* be met by another sufficient sub-task or a combination of sufficient sub-tasks previously taught. Sub-task necessity is defined with respect to a given set of demonstrations being taught. A sub-task is deemed necessary if it is not a redundant sub-task.

Domain experience – We define domain experience, a metric for demonstrator training, as the number of domains experienced thus far in the user study. Note that for each domain, this experience entails participants first providing demonstrations to the robot, and then observing the optimal teaching sub-task breakdown in the form of a video.

Negative Results on the Use of Abstractions with Training Robots

In this section, we discuss the first two human-subjects experiments together as they are both conducted on the same domain. Our first study investigates whether participants teach tasks to robots using abstractions naturally. This is an important question as we expect users to provide abstract, hierarchical task demonstrations to robots but we do not know if users innately want to provide demonstrations with abstractions. We call this study "Investigating Abstraction Use in Robot Teaching" (AUT). We further investigate what types of priming can enable participants to provide abstractions for a single task with a second study. We call the second human-subjects experiment, "Investigating Priming Strategies for Abstractions in Teaching" (PSA). Additionally, we define the metrics used in these experiments and justify the metrics we created to measure the performance of our participants.

Experiment design

We conduct two human-subjects experiments: (1) a 1×4 within-subjects experiment to test if users can be primed to provide sufficient sub-tasks and (2) a 1×4 mixed within-between-subjects experiment with different paradigms to teach users to provide sufficient sub-tasks. We will first describe our research questions, our experiment domain, and the user interface, and we then provide additional details to set up the experiment.

Research Questions We will first establish our research questions and then state our experimental design and study procedures. We formally state the following Research Questions (RQs):

- **RQ 1:** *Do people naturally provide abstractions for learning and planning?* Given that robots need people to provide sub-task-based abstractions, we want to know whether people are already naturally primed to provide such demonstrations.
- **RQ 2:** *Can external factors/ inducements elicit abstraction-based teaching (e.g., ad nauseum repetition, or variation in task composition)?* We also sought to determine whether people naturally chose to use sub-task-based abstractions to teach robots when faced with teaching tasks with numerous, repetitive components or a multi-task scenario where the robot has to solve different tasks in different instances for which the tasks share common sub-tasks.
- **RQ 3:** *Can participants be explicitly taught using textual descriptions of an analog task to provide (more helpful) abstractions?* Requests to provide demonstrations using textual descriptions with figures for an analog task are the simplest teaching guide. We wanted to see if these descriptions are enough to provide correct sub-task-based demonstrations.
- **RQ 4:** *What demographic subject or objective factors and covariates influence how well people used abstractions for teaching?* We sought to examine if demographic covariates help people teach sub-task-based abstractions to robots.
- **RQ 5:** *What explicit teaching guides if any might help the subjects learn to provide sufficient abstractions? Is this teaching guide generalizable to novel scenarios?* We created our second experiment specifically to answer this research question. We know that certain teaching guides work better than others when people are

[†] Sub-tasks are referred to as actions in Garrett, Chitnis, Holladay, Kim, Silver, Kaelbling and Lozano-Pérez (2021); however, we refer to these actions as sub-tasks to prevent confusion between low-level robot actions and TAMP level actions.

given direct instructions to teach robots with specific abstractions (Cakmak and Takayama 2014*a*). In this work, we seek to determine how little information about the current task needs to be provided to induce participants to provide correct sub-task-based abstractions

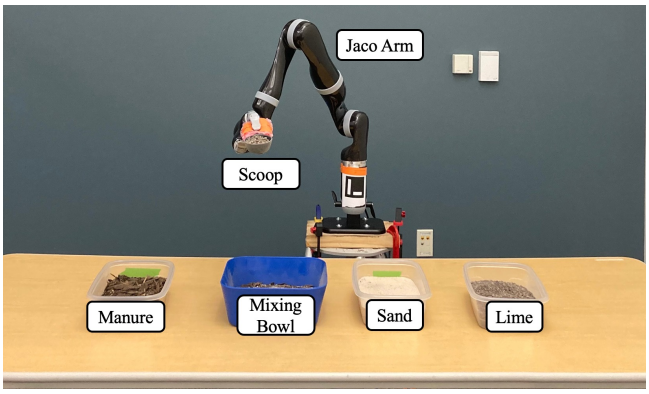


Figure 1. Jaco robot setup: Subjects were required to teach the robot to create different types of soil mixtures in the mixing bowl using the sand, lime, and manure available.

Experimental Setup **Task Domain** We designed a robot task domain setup in which we could create multi-task settings relatively easily with readily available raw materials. Hence, we consider a gardening task in which participants are required to teach the robot to create soil mixtures for different plants as shown in Figure 1. The setup consists of a robot arm, a pot of sand, a pot of manure, a pot of lime (calcium), and a mixing bowl. The subjects teach the agent to create soil mixtures required for three similar or different plants. The action space for this domain is continuous, and the locations of the objects are assumed to be known. The soil mixture domain allows for a multi-task setting and creates opportunities for constrained TAMP problems as described next.

Feasible Task and Motion Planning sub-tasks for the Soil Mixture Domain Here we pick an example problem within our soil mixture domain: "create a soil mixture with one scoop of sand, and one scoop of manure." We then describe feasible abstractions that a user can provide to solve this task in our domain at different levels of granularity, going from coarser to finer-grained TAMP abstractions. These abstractions span the breadth of sub-tasks that our users could demonstrate. We will also describe the relative merits of these abstractions for solving *all possible* tasks in the soil mixture domain.

Coarsest sub-task can be to create the complete soil mixture, one scoop of sand, and one scoop of manure, as a single sub-task, that is, use no abstractions at all when teaching as shown in Fig 2(a). The pre-condition for this sub-task would be that the scoop is empty. The constraints would be that the agent never collides and the goal condition would be to deposit one scoop of sand and one scoop of manure to the bin. However, given this sub-task, the robot can only solve tasks that are multiples of the base level task, e.g., four scoops of sand and four scoops of manure, but not all possible tasks.

A finer sub-task-based abstraction would be to teach the robot to pick and pour one scoop of sand, and one scoop of manure as shown in Fig 2(b). The pre-condition for each

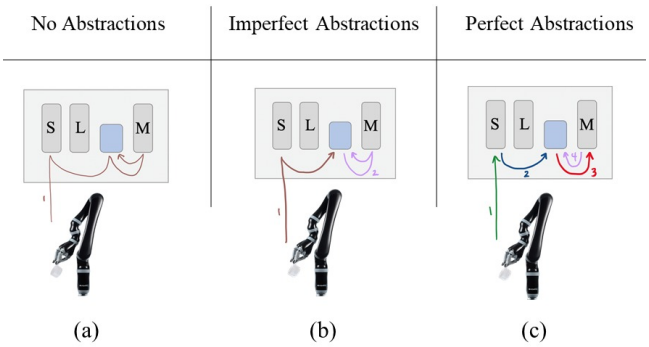


Figure 2. An example of three different types of demonstration strategies to complete the task of "create a soil mixture with one scoop of sand, and one scoop of manure.". (a) If the subject gives a complete end-to-end demonstration as in the case of no abstraction, there is very little generalization to other novel tasks, e.g., creating a soil mixture with two scoops of sand and three scoops of manure. (b) If the participant breaks the task into sub-tasks where pick and place are a single sub-task unit, there is no constraint on picking materials, so if for example, the sand's location changes, the the agent cannot generalize solve tasks. (c) Breaking down the sub-tasks such that picking a material is a different sub-task and pouring a material is another sub-task. Such a TAMP abstraction is sufficient and can solve an un-demonstrated, novel task within the soil domain, without requiring the object locations to be held consistent.

task would be to have an empty scoop, the constraint would be to avoid collisions, and the goal condition would be to deposit the scoop of sand or manure into the bowl. This abstraction is a more generalizable sub-task abstraction as it allows the agent to solve novel tasks that are not present in the demonstrations given by the user. For example, once the robot learns how to pick and pour one scoop of sand and one scoop of manure, it can easily repeat these sub-tasks with the help of a *task planner* to solve the novel task of one scoop of sand, and four scoops of manure. The task planner is used to plan for the right sequence of sub-tasks that will complete the goal of a novel task. However, there is no constraint or condition in this demonstration that the robot picks up the scoop of sand or manure from the correct location, as the location of the sand or manure is not represented in the pre-condition, or the constraints, or the goal condition. If the location of the sand or manure were changed, the planner will ask the robot to perform the scoop gesture where the sand or manure was present during the demonstration and pour an empty scoop of sand or manure in the bowl, never satisfying the sub-tasks goal condition.

The finest feasible sub-task abstraction would be to teach the robot to pick sand and manure, and then teach the robot to pour the sand and manure as shown in Fig 2(c). To teach the pick sand sub-task, the pre-condition would be an empty scoop, the constraint would be to avoid collisions, and the goal would be to have sand in the scoop. To teach the pour task, the pre-condition would be to have a scoop with sand, the constraint would be to avoid collisions and the goal would be to drop the scoop of sand in the bowl. Such a sub-task demarcation allows the robot to understand the right pre- and post-conditions for each sub-task. Specifically, the robot learns to pick up the sand from any location as a sub-goal allowing the agent to pick up and pour objects to and from any

location on the table. Moreover, the robot can again combine multiple pick-and-pour actions to deliver any required ratio of sand, and manure. This is a sufficient sub-task partition allowing the robot to solve the entire multi-task soil mixture domain with changing locations.

Robot Platform For our first experiment, we used Sawyer, a seven degrees-of-freedom (DoF) arm from Rethink Robotics. We switched to using a Kinova JACO seven (DoF) arm for the second experiment (Figure 1) because of mechanical failures on the Sawyer robot. Both robots can play back different demonstrated trajectories with high precision enabling non-expert users to teach the robot. We record the 2D location of objects (sand, manure, lime and the mixing bowl) on the table before we start the task and provide it as input to the TAMP problem. The only other variables are robot's joint positions. We use our TAMP planner with DMPs as specified in the appendix to plan on the robot given an input state space.

User Interface We designed a user interface that allows participants to save demonstration trajectories and reuse them to solve tasks. The design of the interface was fine-tuned using iterative design methods during the pilot studies. The image of the interface is shared in the supplementary appendix A.2 [‡]. The interface enables subjects to create and name sub-tasks, and then give a fixed number of demonstrations per sub-task. The sub-tasks can be reused by participants as many times as needed. Moreover, there is a procedure column for each occasion where subjects can create tasks by adding the demonstrated sub-tasks to the column sequentially to satisfy the occasion's task. The interface allows subjects to use abstractions, thereby creating shorter, repeatable, sub-tasks if so chosen.

All study participants were given equivalent training using the interface via a training video. Modulo latent confounders we have done our best to reduce confounds created by the interface itself during the experiment using iterative design, keeping the interface common across the conditions, and using training videos to provide equal training. Our results in the Section titled "Results for AUT and PSA Experiments" show that our interface and training were *sufficient* for participants to create abstractions. All our instruction videos and documents are provided in the supplementary website [*].

AUT Experiment: Investigating Abstraction Use in Robot Teaching In the AUT experiment, we investigate whether subjects are intrinsically motivated to provide abstractions when giving demonstrations to the robot or can be primed to do so. We conducted a 1×4 within-subjects experiment with 28 participants (39.3% Female, Mean age = 21.42, Standard Deviation = 2.61) where the independent variable is the phase of the experiment. We also vary the order in which the phases are introduced to control learning effects. Each study phase corresponds to the type of task or amount of training the subject receives when teaching the agent. In each phase, the subject has to create three soil mixtures for different plants; we call these "occasions" in the study so the subject treats them as three distinct occasions of creating plant soil mixtures. The four phases are the *Baseline* phase, the *Multi-task* phase, the *Large Number of Repeats* phase, and the *Multi-task via Written Instructions* phase.

- **Baseline (B):** Participants teach the agent a single task on three occasions with a few repetitions within this phase. The demonstration task for this phase involves creating a mixture of two cups of sand and one cup of manure.
- **Multi-task (MT):** The subject has to teach the agent different tasks for each of the three occasions in the MT phase. These tasks range from creating a soil mixture with the following number of scoops of objects for each of the three occasions: two of sand and one of lime, one of manure and one of lime, and one of sand and one of manure, respectively.
- **Large Number of Repeats (LR):** The subject has to teach the same task for each of the three occasions in the LR phase, but the task itself has a lot of repetitions within it. The task for this phase involved creating a soil mixture of ten scoops of sand and three scoops of lime.
- **Multi-task with Explicit Teaching via Written Instructions (MT+W):** The trainer gets explicit written instructions to use abstractions when training the agent. In the instructions, we describe abstractions in an unrelated task of cooking eggs. We also attempt to solicit correct abstractions by describing the robot's learning constraint in text.

As this is a within-subjects study with predictable learning effects across conditions, the ordering of the phases plays an important role in understanding subjects' ability to provide abstractions. Since we want to study whether the participants naturally tend to provide abstractions or not, the subjects always begin with the baseline (B) phase. To establish which phase induces abstractions faster we introduced the study phases to the participants in one of two possible orders. **Order 1**: B, MT, LR, MT+W. **Order 2**: B, LR, MT, MT+W. We only change the order with the MT and LR phases as giving instructions upfront, i.e., MT+W will bias the subjects to provide abstractions in the prior phases. We study the effect of introducing the two MT and LR in the results of this experiment in the section titled "Results for AUT and PSA Experiments", where we address RQ 2.

PSA Experiment: Investigating Priming Strategies for Abstractions in Teaching From our AUT experiment, we found that although participants learned to provide some form of demonstrations the participants generally failed to provide sufficient abstractions in those demonstrations to solve multi-task domains with constraints. Thus, we conduct a follow-up experiment where we consider direct teaching modes: (1) a robot's debugging demonstration, (2) a video of an analog task, and (3) an expert demonstration video of the same task the participants are teaching. The PSA experiment is a 1×4 mixed within-between-subjects study with 24 subjects (45.83% Female, Mean age = 20.875, Standard deviation = 2.69). The experiment has three different teaching modes along with a baseline condition of no teaching. All participants experience the baseline phase and the phase which shows an expert demonstration video of the *same task* that the participants are teaching the robot. Moreover, half the participants observe the video demonstration of an

[‡] https://sites.google.com/view/investigating-strategies-1/home

analog task, and the other half of the participants observe a debug demonstration that shows the consequence of their demonstration strategy. The experiment was constructed in this way to avoid learning effects between the teaching mode and to avoid fatigue by keeping the length of the experiment to less than 2.5 hours. In all modes, the subject attempts to teach the robot in a multi-task scenario where each occasion has a different task. Moreover, the sand, lime, and manure pots' locations are changed between demonstrations. The object locations are changed to emphasize the need to teach constraint-based sub-tasks with their demonstrations. The four conditions are as follows:

- **No teaching (NT):** Here no instructions are provided to the subjects. The subjects are free to use any strategy to teach.
- **Debug demonstration (DD):** The subjects are first provided with written instructions with diagrams showing sufficient abstractions in a similar task of touching two blocks. Further, the users are shown the consequence of the abstractions they provided in the "No Teaching" phase using a trajectory demonstration on the robot. Additionally, we also move the mixing bowl and the pot of sand to a new location. We used the demonstration strategies we observed in the first experiment to create these Wizard of Oz, debugging demonstrations. They have been designed to be informative about every sub-task a participant could have taught to successfully complete the overall task.
- **Video of analog task (VA):** We provided the users with a video that demonstrated using our interface to teach the robot a related constraint-based task of touching different blocks in a specific sequence.
- **Expert demonstration video of the Soil Mixture Task (EV):** We also wanted to see if providing a video demonstrating sufficient abstractions for parts of the soil mixing task would aid the subjects to extrapolate and provide sufficient abstractions for the entire task.

All participants started off with a phase of no teaching mode. They then either completed a debug demonstration or a video for an analog task for their second phase in the experiment. Then all subjects finished with a final phase where they were shown a video showing parts of the soil mixture task. More details about the PSA Experiment conditions are provided in Appendix A.3. We know from prior work that showing a video tutorial for a task is sufficient to teach abstractions (Cakmak and Takayama 2014*a*). Hence, we chose to show videos of partial task solving towards the end to establish that people can provide sufficient abstractions with a little help from an expert in the problem domain without the complete solution. We conducted a between-subjects study, comparing a debugging demonstration and the video of an analog task to prevent learning effects between the two modes and to keep the study duration fatigue-free for participants.

Study Procedure Prior to the start of both the first and second experiments, we obtained approval for human-subjects experimentation from the Institutional Review Board (IRB), protocol #H21036 at our affiliated institution. We recruited all participants through university mailing lists for both of our studies. Due to the COVID-19 pandemic, we were unable to conduct large-scale user studies with off-campus participants. Nonetheless, we were able to recruit 28 and 24 participants for the first and second studies, respectively. All participants were compensated with a $25 and a $35 Amazon gift card for the first and second studies, respectively. No participant from the AUT experiment was allowed to take part in the PSA experiment. The procedure for both the user studies was quite similar and took a maximum of 2.5 hours to complete.

Upon arrival, the participants were asked to complete a pre-experiment questionnaire assessing demographic information, and pre-surveys that include the Big-5 personality test and their previous experience in teaching children or students. Subjects then participated in a practice round to get familiarized with the user interface while providing kinesthetic demonstrations to the robot. The experimenter then explained the soil-mixture task and the user interface to the participants. In the AUT experiment, the participants begin with the baseline condition and follow either **Order 1** or **Order 2**. The order for each trial is chosen at random. After each phase, the participants also filled out a questionnaire to measure their workload using the NASA-TLX (Hart and Staveland 1988*b*). At the end of teaching tasks for all the phases in the first experiment, the participants took an online IQ test (FSIQ 2019).

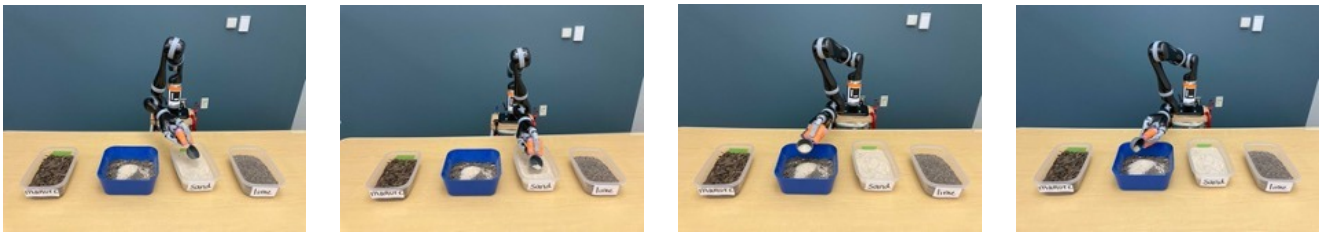
In the second experiment, the participants follow the same pre-study procedures. Participants perform three soil mixture tasks with different teaching modes to help them provide sufficient abstractions. The subjects will begin with the NT condition, followed by either DD or VA (the between-subjects component). They will conclude with the EV condition. Participants also filled out the NASA-TLX workload questionnaire after each condition. The between-subjects variable (DD or VA) for each trial was randomized at the start of the trial.

Metrics

We used the following metrics to measure the performance of the users teaching our robots within the two experiments in this study.

Pre-study Questionnaire

- **Demographic Information:** We collect participants' age, gender, education, and race/ethnicity.
- **Personality** At the start of the study, participants filled out the Big-5 personality questionnaire (John and Srivastava 1999) on a five-point Likert scale.
- **Prior Robotics Experience** We ask a hand-crafted single-item question rated on a scale from 0 to 10+ years.
- **Prior Teaching Experience** We obtain participants' prior teaching experience through a hand-crafted, 5-question survey.
- **Negative Attitude towards Robotics** We employ the Negative Attitudes Towards Robotics (NARS) Scale (Syrdal, Dautenhahn, Koay and Walters 2009), composed of 14-questions rated on a seven-point scale (Strongly Disagree=1 to Strongly Agree=7). We report results on the three sub-scales: negative situations, negative social influence, and negative emotions.



(a) Reaching for sand. **(b)** Picking up sand. **(c)** Move to bowl. **(d)** Dumping sand.

Figure 3. Series of images demonstrating the plan created from learned trajectories on the robot to pick and drop sand from the set of 10 tasks created to test the learned policies from different demonstrations. For more details refer Appendix B.1.

Objective Metrics

- **Task Completion Time:** The duration of each phase was measured and was known to the participants.
- **Abstraction Score:** We also created an abstraction rubric to measure the performance of the participant in providing useful abstractions. The rubric provided a point for every valid TAMP abstraction provided by the user as described in the Section titled "Experiment design." Moreover, a point is also provided for every valid TAMP abstraction that can be created from the abstractions provided by the user, i.e., if the user provided finer-grained TAMP abstractions, such as *picking sand* and *pouring sand*, the rubric also provided points for other coarser TAMP abstractions that can be satisfied by the finer abstractions, such as *pick and pour one cup of sand*.

  This scoring strategy is important, as if valid and generalizable low-level abstractions are provided to the robot it can solve more tasks. However, we do not award points for extremely low-level abstractions, e.g. *move* 1 cm *to the left*, which would not be efficient in solving the task. All the valid abstractions that a user can provide to the robot in the soil mixture domain have been described in the Section titled "Experiment design."

  We show in the section titled "Validation of the Abstraction Metric," and with Fig. 4 that this is a valid scoring strategy to measure the generalizability of a given demonstration to solve a wide variety of tasks.

  For example, in the task of making a soil mixture with two scoops of sand and one scoop of manure, if the participant gave a demo of the complete task, the demo would get 1 point. However, if the participant broke the task into creating abstractions of scooping one cup of sand and another for scooping one cup of manure and used these constraint-based sub-tasks to complete the overall task, then the demo would receive one point for a scoop of sand another for manure and one additional point to complete the overall task. Abstractions earn more points as breaking up the tasks into sub-tasks helps solve other tasks. The rubric does not award points for just taking a low-level action or teaching an unnecessary sub-task. To gain a point the created abstraction needs to create a sub-task based on a valid constraint.
- **Binary Abstraction Score**: We created a *binary score* where a participant's demo scored 1 if the demo had *any* abstraction in the phase and 0 if the demo did not.
- **Perfect Abstraction Score**: Finally, we checked whether the demonstrations given by the participants created valid sub-tasks with valid constraints. The participant's demonstrations were scored 1 if *sufficient* abstraction was provided in the phase, else 0. The participants were unaware of this rubric and were told to complete the phases as efficiently as possible.

Post Study Questions

- **Workload:** Participants filled the NASA-TLX questionnaire (Hart and Staveland 1988*b*) to assess perceived workload for each condition.
- **IQ Metric:** We gave the participants in the first experiment an approximate open-source Intelligence Quotient (IQ) test (FSIQ 2019) to test whether IQ has any relation to the ability to teach with abstractions. This test took each participant approximately 30 minutes to perform. We note that we do not employ an official IQ test in our work and that the test we use is not a replacement for a real IQ test. The IQ test we employ simply demonstrates the types of questions and skills necessary to perform well on an IQ test. We also note that this measure of IQ is imperfect as it was performed with an open-source, online test rather than a trained, in-person examiner. As prior work has found that IQ was associated with better problem-solving accuracy (Lee, Ng, Ng and Lim 2004) and that IQ is a strong predictor of academic achievement (Mayes, Calhoun, Bixler and Zimmerman 2009), in our study, the IQ metric is used as a proxy for problem-solving capabilities/academic achievement. We did not conduct the IQ test for the participants in the second experiment because the results from the first experiment answered the relevant research question.
- *Verbal Interviews:* We used these interviews to understand the participants' training strategies and explain the purpose of the experiment.

Validation of the Abstraction Metric

We first demonstrate that the abstraction score we created as described in the previous section is valid. We then present our investigations into the research questions posed in the Section titled "Research Questions."

Planning with the learned Policies on the Robot We justify the creation of the abstraction score by comparing the task-solving potential of trajectories demonstrated by our participants on the real robot. For this, we created

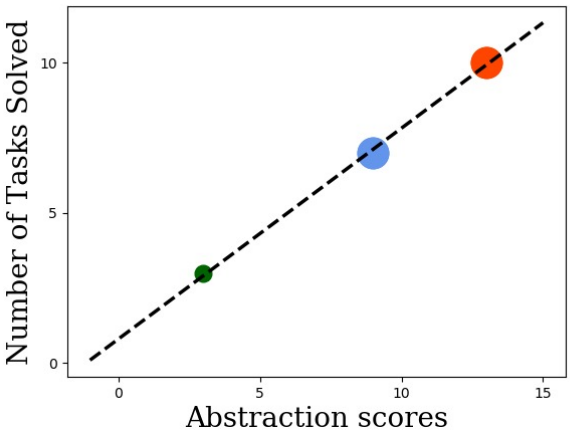


Figure 4. Here we have plotted the abstraction score and the corresponding number of tasks *five* users' demonstrations were able to solve on the robot. There are a total of ten tasks and seven of these ten tasks are unseen to the robot previously. The larger circles indicate two users' demonstrations for the given score and tasks solved. The colors indicate the type of abstraction taught, green for no abstraction, blue for imperfect abstractions, and red for sufficient TAMP abstractions. For a demonstration to be good, it should be scored higher, as the robot can solve all the tasks. As it can be seen from the plot these good demonstrations also have a large abstraction score. The dotted line is the linear regression of the scores vs the tasks solved, and it demonstrates that the abstraction scores that we created correlate well with the number of tasks that a given demonstration can solve.

a set of 10 tasks to be solved by 5 demonstration sets chosen to represent different ranges of the abstraction score. This comparison shows that demonstrations that are given without sub-task abstractions can solve fewer than half the tasks. Specifically, tasks where the locations of the objects are changed arbitrarily can only be solved by the demonstrations in the highest quartile of the abstraction scores as observed in the multi-task phase with clear instructions in the first experiment. To measure this we create a set of ten tasks, in which seven tasks are completely novel, i.e., users did not provide any demonstrations for these seven tasks. Demonstration sets that do not use abstractions to train the robot can solve only the exact task that was taught to the agent, i.e., three out of ten tasks. With a slightly higher quality abstraction, where the users break apart tasks into picking and pouring individual scoops of sand, manure and lime, the robot can plan arbitrary combinations of these sub-goals, allowing the robot to solve seven out of ten tasks. The demonstration sets were given keeping in mind that the robot learns using goals and constraints for sub-tasks, and separates the picking and pouring for scoops of objects to solve all ten tasks even when object locations change. This experiment primarily demonstrates that our abstraction scoring system was practical, and higher abstraction scores for demonstrations indicate the ability to solve a larger number of possible tasks. Figure 4 shows that demonstrations that solve more novel tasks on the robot, also have high abstraction scores. Hence, our abstraction scores are a valid measure of a demonstration's quality in solving novel tasks using the TAMP formalism. The complete experimental details are provided in Appendix B with the whole set of tasks and their outcomes in Table 1 of the appendix. An example of the trajectory is shown in Figure 3 along with a video supplement showing multiple trajectories[§]. These empirical results validate that our abstractions score quantifies the capability of a given demonstration to generalize to novel tasks. Next, we will discuss our Research Questions and their implications.

Results for AUT and PSA Experiments

We present our findings for the individual research questions across the AUT and PSA experiments.

RQ1: *Do people naturally provide sufficient abstractions for learning and planning?* Results from our AUT experiment using the Binary Score indicate that only 35.71% of the participants used any abstraction in the baseline phase, implying that the majority of the participants do not provide abstractions naturally.

Takeaway: We posit that the majority of subjects have difficulty in knowing where to break a task in a continuous robot domain, as there is no natural indication of what a sub-task for a robot could be.

RQ 2: *Can external factors or inducements elicit abstraction-based teaching?* In our AUT experiment, we examine the effectiveness of using different priming methods to help subjects use abstractions while providing robot demonstrations. A Wilcoxon-signed rank test with abstraction score as the dependent variable and study phase as the independent variable shows that there exists no statistical difference in abstraction scores between the **LR** phase and the **MT** phase. Further, we conducted a Cox-Regression Hazard analysis to verify if task order might be critical in determining the number of abstractions a user provides, but did not find any significance.

Takeaway: Our results imply that seeing a large number of repetitions and a multi-task setup both encourage people to use abstractions at similar rates.

RQ 3: *Can participants be explicitly taught to provide (more helpful) abstractions?* In the robot study 24 out of the 28 (85.71%), participants learned to teach abstractions to the agent after **MT+W** phase (measured with binary abstraction score). The remaining four participants could not learn to break tasks apart to teach the robot. The most common form of abstraction chosen was to "pick and pour sand," "pick and pour manure," and "pick and pour lime." When tested against the **MT+W** phase where explicit instructions were given to break down tasks into repeatable sub-goal-based abstractions, participants succeeded in providing abstractions and performed significantly better. We ran multiple Wilcoxon-signed rank tests with Bonferroni correction ($\alpha = 0.05/6$) to compute pairwise comparisons for abstraction scores across the different study phases in the AUT experiment. Results from the Wilcoxon-signed rank tests indicate that abstraction scores from the **MT+W** were significantly better than the baseline ($Z = 165$, $p < 0.0001$), **LR** ($Z = 598$, $p < 0.001$), and **MT** ($Z = 560$, $p < 0.001$) with

[§]https://sites.google.com/view/experience-impact-abstraction/home

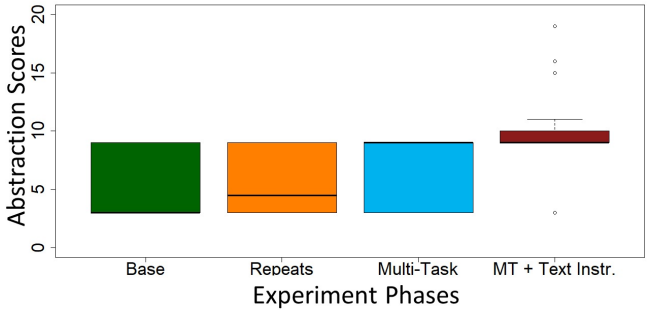


Figure 5. Box plot indicating the abstraction score distributions for the phases of baseline, large number of repeats, multi-task, multi-task with instructions, respectively for the AUT experiment. As soon as participants are directly asked, using textual instructions, to teach using sub-task-based abstractions, in the multi-task with instruction phase, the majority of the participants choose to do so, but they still fail to provide optimal sub-task-based abstractions.

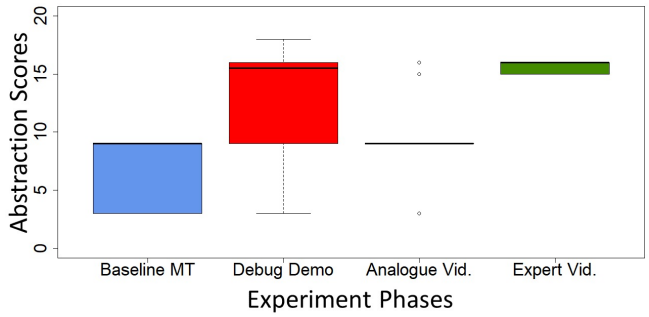


Figure 6. Box plot indicating the abstraction score distributions for the phases of Baseline Multi-task, Debug Demo, Analogue Video, and Expert training demonstration, respectively for the PSA experiment. Note that the expert training demonstration video that shows a partial solution performs much better than other modalities to train subjects.

effect sizes $0.779, 0.740, 0.697$; respectively. The box-plot of all the abstraction scores are in Figures 5 and 6. Although 24 participants were able to provide abstractions in the MT+W phase, only 7 out of the 28 participants provided demonstrations for sufficient sub-tasks according to the perfect abstraction score, despite being given clear instructions that the robot cannot touch two objects in the same trajectory as these are different goal constraints.

Takeaway: From the first experiment, we note that a majority $(24/28)$ of the participants were able to provide task abstractions after being primed with the MT+W phase. However, only a small fraction $(7/24)$ of the participants were able to provide sufficient abstractions. These results show that teaching long horizon and multi-task problems to robots is not trivial, and correspondence problems between the robot and its teacher can be an issue in robot teaching.

RQ 4: *What demographic subjective or objective factors and covariates influence how well people use abstractions for teaching?* To analyze which subjective and objective factors play a significant role in influencing a user's ability to provide abstractions, we created a linear mixed effects model with abstraction score as the dependent variable, and the independent variables being study phases, conditions, with covariates of age, IQ, and personality score. We pick the model with the lowest Akaike information criterion (AIC) by pruning variables, and covariates from the largest possible model. All of the models were tested for normality and homoscedasticity for which the details are in the supplementary Appendix A.1. For the first experiment, we found the abstraction score was significantly dependent on the phase of the study ($F(3, 112) = 25.05, p < 0.001$) and the IQ of the participants with ($F(1, 112) = 6.81, p = 0.01$).

Takeaway: Our analyses indicate that the user's ability to provide task abstractions is significantly dependent on the study phase and IQ.

Condition 1	Condition 2	p value	Effect of size
EV	VA	< 0.005	0.863
EV	NT	< 0.005	0.829
VA	NT	< 0.01	0.757
DD	NT	< 0.005	0.829

Table 1. p-values for pairwise comparison of teaching modes on abstraction scores of subjects, after the Bonferroni correction.

RQ 5: *What explicit teaching strategies, if any, might help the subjects learn to provide sufficient abstractions? Is this teaching generalizeable to novel scenarios?* We computed six Wilcoxon-signed rank with Bonferrroni Correction ($\alpha = 0.05/6$) tests for pairwise comparisons of perfect abstraction scores across all combinations of the teaching modes used in the second experiment. The significant results from our pairwise comparisons are listed in Table 1.

We also compare the ability of participants to provide the right sub-task decomposition (or abstractions) after the final phases of the first experiment (**MT+W**) and the second experiment **EV**, with a Wilcoxon-signed rank test and find that the abstraction scores of participants in the **EV** condition are significantly better ($Z = 568$, $p < 0.001$, effect size=0.616).

Takeaway: Our results indicate that **EV** is the most effective technique in eliciting sufficient abstractions from non-experts for teaching a robot in a multi-task, long-horizon setting. However, this approach does not scale well to novel tasks that an end-user might want to teach a robot. These results imply that showing the participants a video of the expert demonstrating the training to teach the same task that the participant is teaching is better than other teaching modalities to help the robot learn to solve novel tasks. However, providing such videos for a household-hold robot would not be possible in all cases.

Note on Perceived Workload and Abstractions: To analyze how providing abstractions can affect the perceived workload of a user, we employ a linear mixed effects model (LMER) with workload as the dependent variable. Our results show that workload was significantly dependent on the interaction effect between abstraction scores ($F(3, 112) = 11.48, p < 0.001$), and the phase of the study, with a linear effect from the IQ of the participant ($F(1, 112) = 5.29, p = 0.02$) for the first experiment. However, the perceived workload was not dependent on any of the independent variables with significance in the second experiment. We hypothesize that this is because all the phases had multi-task scenarios, reducing the significance of variables such as the study phase or ordering, to predict workloads. Finally, we believe there are important mediating effects between

measured IQ, workload, and score, which are difficult to isolate due to assumptions of available statistical procedures for mediation analysis for the AUT experiment.

Confound of Domain Experience

In the previous section, we described our results on whether users teach robots with abstractions and whether any priming strategies help users in teaching a single task to the robots. However, we identify a confound over experience teaching the robot: as the users get familiar with teaching robots, they get better at specifying the right abstraction. In the next section, we describe our results where we test this confound of participants experiencing multiple domains and describe a human-subjects experiment where we observe the learning effect between tasks and domains for users teaching robots.

Investigating the Impact of Experience on a User's Ability to Perform Hierarchical Abstraction

In this section, we discuss the third human-subjects experiment, where we investigate the learning effect between tasks and domains for users teaching robots. Furthermore, we develop and validate a robot-teaching experience-based approach to enable end-users to teach robots generalizable abstractions for novel long-horizon tasks. While the previous two studies (AUT and PSA) were conducted only on one (Soil-Mixture) domain this experiment is conducted across multiple domains to ensure teaching compatibility across different types of robot tasks.

EPA Experiment: Investigating the role of Experience in providing abstractions

We conducted a 1×4 within-subjects experiment with twenty-eight participants, seven per ordering condition (see Appendix for the domain ordering of each condition). Participants experience five domains in this study, a practice domain that all participants experience first, and the four ordered domains. We control for the ordering of the remaining four domains using a Latin square, ensuring that the participant count per condition is balanced. The independent variable in this study is the number of domains encountered thus far.

The robot employed in this study is the JACO arm (Gen2, three fingers for a total of seven degrees of freedom) (Campeau-Lecours, Maheu, Lepage, Lamontagne, Latour, Paquet and Hardie 2016) attached to a hand-crafted base located next to the experiment's table, as seen in Figure 7. We additionally designed a user interface that allows users to record and save sub-tasks they demonstrate to the robot; interface design decisions can be found in the Appendix. Participants can then use the interface to combine different sub-tasks to accomplish a task. We name the group of sub-tasks assigned to a particular task a *recipe*. We require the participant to record three demonstrations for each sub-task in order to capture variability in the way the participant moves the robot for robustness to noise.

Research Questions

- ***RQ 6:*** *What is the impact of domain experience on the quality of demonstrations?* We investigate whether participants can perform zero-shot transfer to novel domains of any acquired knowledge as measured by sub-task abstraction score, teaching efficiency, and sub-task redundancy.
- ***RQ 7:*** *What is the effect of demonstration abstraction on participants' perceived workload?* We hypothesize that higher abstraction scores will reduce the repetitiveness of participant demonstrations, thereby reducing perceived workload.
- ***RQ 8:*** *Do participant demographics impact the quality of demonstrations?* We investigate whether participant demographics, such as prior robotics experience and prior teaching experience, impact the quality of their demonstrations. We posit that participants with robotics or teaching experience will teach the robot, via demonstration, more effectively and efficiently.
- ***RQ 9:*** *Does domain type impact the quality of demonstrations?* We hypothesize that the domain type will impact the sub-task count and redundancy, abstraction score, and teaching duration.

Metrics

We collected the following metrics as part of this user study:

- **Abstraction Scores:** We used the abstraction scores as justified in the previous section.
- **Redundancy Score:** We count the number of redundant sub-tasks taught, i.e., sub-tasks whose function can be fulfilled by another existing sub-task or a combination of existing sub-tasks. This metric allows us to evaluate the necessity, independently from the sufficiency, of demonstration sub-tasks.
- **Sub-task Count:** We count the total number of sub-tasks taught to the robot in each domain, that are employed to accomplish a task.
- **Teaching Duration:** We measure the total time the participant taught the robot similar to the previous studies.

The subjective metrics in our user study are as follows. The details of hand-crafted surveys, Cronbach's alpha, and qualitative results and quotes from interview questions are in the Appendix.

Pre-study Questionnaire

- **Demographic Information:** We collect participants' age, gender, education, and race/ethnicity.
- **Personality** Prior work has found a statistically significant relationship between teachers' personality type and the degree to which they are effective teachers (Fatemi, Ganjali and Kafi 2016), and that extroversion is correlated with overall teaching efficacy (Roberts, Harlin and Briers 2007). As such, we included the Big 5 Personality Score as a covariate as we hypothesized that if a participant is an extrovert, they may be an effective robot teacher. The Big Five Personality survey (Goldberg 1992) consists of fifty questions rated on a seven-point scale (Very Strongly Disagree=1 to Very Strongly Agree=7).

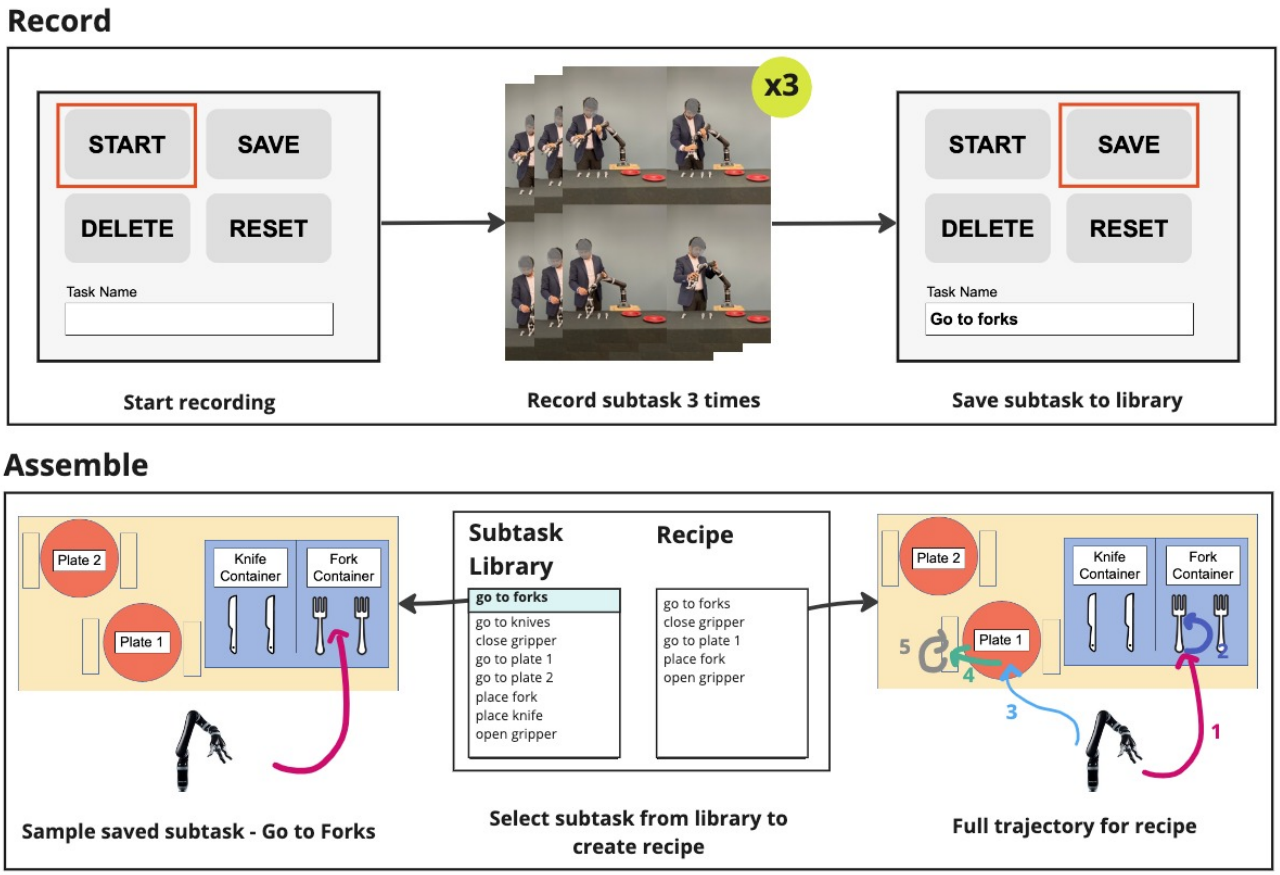
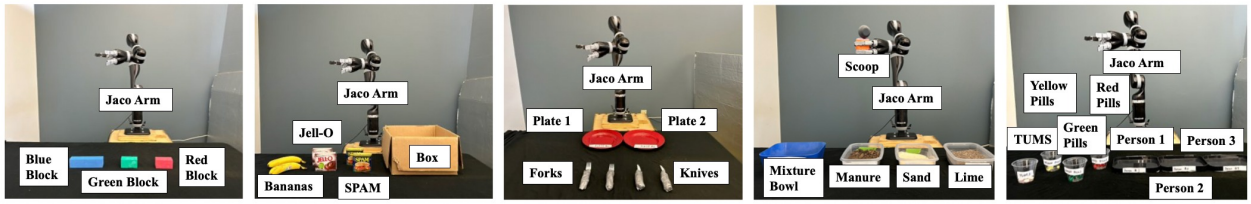


Figure 7. Via the interface and kinesthetic teaching, participants record three demonstrations to save a sub-task. Saved sub-tasks are then available in the interface library. To execute a task, participants assemble a recipe from the set of recorded sub-tasks.



Block Touching | Box Packing | Table Setting | Soil Mixing | Medicine Dispensing

Figure 8. The five domains that the participants trained the robot in the third human-subjects experiment.

- **Prior Robotics Experience** We obtain participants' prior robotics experience through a hand-crafted single-item question rated on a scale from 0 to 10+ years.

- **Prior Teaching Experience** We obtain participants' prior teaching experience through a hand-crafted, 5-question survey rated on a five-point scale (Strongly Disagree=1 to Strongly Agree=5).

- **Negative Attitude towards Robotics** We employ the Negative Attitudes Towards Robotics (NARS) Scale (Syrdal, Dautenhahn, Koay and Walters 2009), composed of 14-questions rated on a seven-point scale (Strongly Disagree=1 to Strongly Agree=7). We report results on the three sub-scales: negative situations, negative social influence, and negative emotions.

Post-domain Questionnaire

- **Teaching Strategy** After completing each domain, we ask participants to "please explain your strategy and thought process when teaching the robot in this domain."

Post-study Questionnaire

- **Workload** We use the NASA Task Load Index (NASA TLX) (Hart and Staveland 1988*a*) to obtain perceived workload.

- **Impression of Agent** We use the Perceived Intelligence and Likeability sub-scales of the Godspeed Questionnaire Series, rated on a 5-point scale (Bartneck, Kulić, Croft and Zoghbi 2009).

- **Post-Interview** We ask participants five post-interview questions. The question list and qualitative results can be found in the Appendix.

Domains

We employ five domains in this study, each comprised of three tasks that the participant must teach the robot, as seen in Figure 8. The set of tasks in each domain was designed to be repetitive and time-consuming to encourage participants to use sub-task abstractions in order to avoid recording repetitive sub-tasks. Additionally, we chose these domains because they are representative of common household chores that humans could reasonably be asked to teach a robot: setting the table, packing lunch, gardening, and dispensing medication. Each task has a distinct objective with differing numbers of objects and goals but requires similar types of abstractions from the participant.

Furthermore, when teaching a robot a task in a residential or "unsanitized" setting, there will likely be objects in the environment that are unrelated to the task being taught. We thus employ distractor items in this study, which are present but not relevant to the list of tasks the participant must teach the robot in the domain, to realistically represent such settings.

Participants were given unlimited time to record sub-task demonstrations and build recipes using these recorded sub-tasks in the interface. The optimal sub-task list for each task in each domain can be found in the Appendix. We now describe the domains in this study (Fig. 8).

Block Touching: A blue, green, and red block are placed in front of the robot. Participants are asked to teach the robot to touch the blocks in a particular order using the robot gripper.

Box Packing: Two plastic bananas, two Jell-O boxes, and two Spam cans, along with a cardboard box are laid out in front of the robot. Participants are asked to teach the robot to pack (pick up and place) a combination of these food items into the cardboard box.

Table Setting: Two forks, two knives, and two plates are placed in front of the robot. Participants are asked to teach the robot to set the table by picking up the utensils and placing them in designated locations around the two plate settings.

Soil Mixing: A bucket of manure, a bucket of sand, and a bucket of lime are placed in front of the robot, along with a mixing bowl into which scoops of each of these materials are to be poured. Participants are asked to teach the robot to create different soil mixtures for different plants. In this domain, the scoop is placed in the robot's gripper by the experimenter.

Medicine Dispensing: Four kinds of medicine (red pill cup, green pill cup, yellow pill cup, and TUMS pill cup) along with three trays labeled persons 1, 2, and 3 are placed in front of the robot. Participants are asked to dispense the proper medication to each person by picking and placing medicine cups into the appropriate person's tray.

Study Procedure

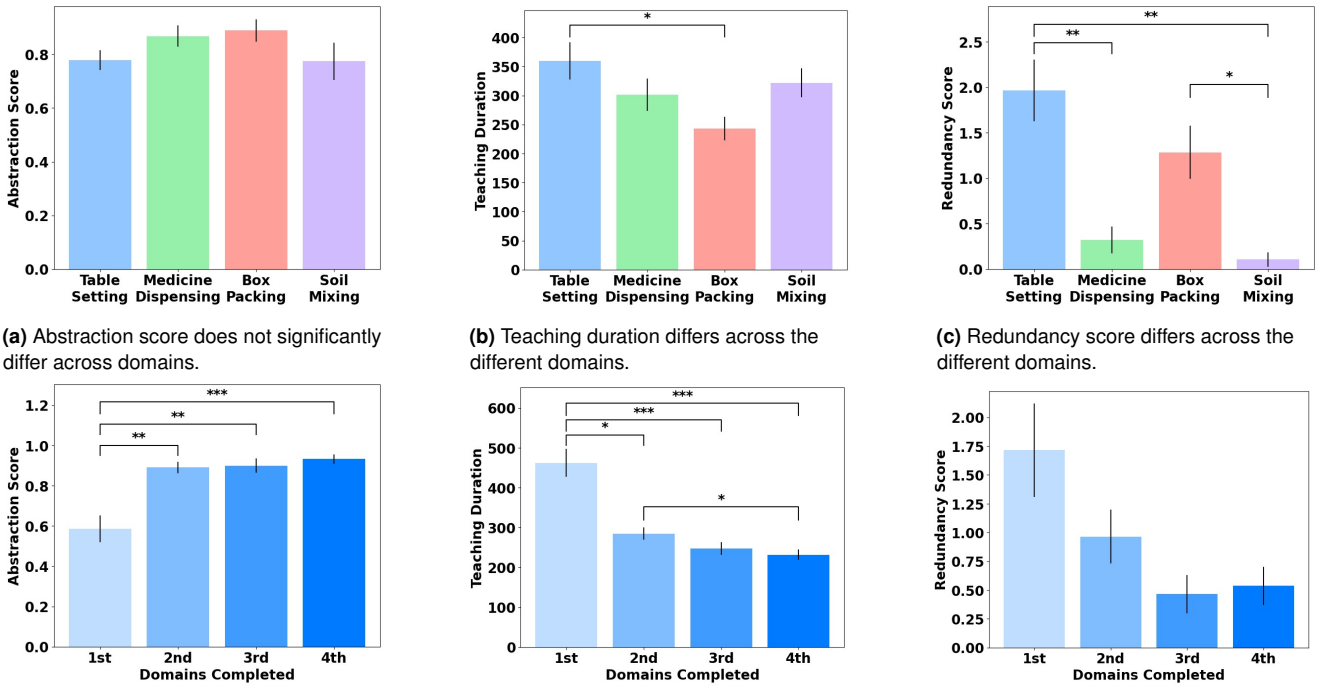
This study was approved by our university's IRB, protocol #H22450. We recruited all participants through advertisements on campus. The study took three hours, and participants were compensated with a $50 Amazon gift card, given the long duration of the study. Participants were not explicitly given breaks during the study. However, the consent form told participants they can take breaks (for instance, to go to the bathroom). The procedure of the study is as follows.

Participants first take the pre-study questionnaire, comprised of surveys to collect demographic information, personality measures, prior robotics experience, prior teaching experience, and negative attitude towards robots. After the pre-study questionnaire, participants start the training portion of the study. To begin, they observe the introduction video. The introduction video[¶] introduces the study, the robot, and the interface used to teach the robot sub-tasks. The video then consists of a conceptual description of how to optimally teach the robot to make an omelet. The optimal sub-tasks described for this example included (1) going to the egg carton, (2) picking up an egg, (3) going to the pan, and (4) breaking an egg into the pan. This portion of the video motivates breaking up the task into sub-tasks that can be called many times, to generalize to an omelet of any quantity of eggs. It also suggests recording the "go to the egg container" sub-task separately from the "pick up the egg" sub-task, allowing the robot to generalize going to an egg carton whose location has been moved. Finally, this video communicates that the sub-tasks can be called from generalized starting positions so multiple sub-tasks can be chained together without going back to a home position first. We note that this initial omelet domain is experienced entirely virtually, and we do not show the participant how it would be taught on the physical robot.

Next, participants teach the robot to complete the three block touching tasks in the demo (i.e., practice) domain. After this demo domain, the participant explains their teaching strategy, recorded via a voice recording. We note that after this demo domain, we do not show the participant a video of the optimal way to teach the robot. The demo domain and task are not the same as the ones participants encounter later in the study. Additionally, since a video of the optimal way to teach is not shown for the demon domain, the demo domain is absent of a learning signal for the participant, contrary to the following domains. As such,this domain does not contribute to the teaching experience of the participant. This block-touching domain serves to familiarize the participant with moving the robot and using the interface.

Then, for the testing portion of the study, participants teach the robot how to accomplish tasks in four different domains: box packing, table setting, soil mixing, and medicine dispensing. Each participant experiences one ordering condition, which defines the order in which the domains are encountered. All participants experience each of these domains (within subjects). The four domain ordering conditions are listed in the Appendix. For each of these four domains, participants are introduced to the domain verbally, then asked to teach the robot how to do three tasks in that domain using the interface, as seen in Figure 7. To teach the sub-tasks, participants provide kinesthetic demonstrations in which participants physically manipulate the robot. After teaching the robot, the participants answer the post-domain interview question and then observe a video showing the optimal way of teaching the robot in that domain (communicating the proper sub-task breakdown) prior to experiencing the next novel domain. The optimal teaching strategy video for each domain was designed to communicate how to optimally teach the robot, listing the optimal sub-tasks for the domain, along with how to teach and record those

[¶]The videos employed in this study can be found at https://sites.google.com/view/experience-impact-abstraction/home.



(a) Abstraction score does not significantly differ across domains.

(b) Teaching duration differs across the different domains.

(c) Redundancy score differs across the different domains.

(d) Abstraction score increases with more domain experience.

(e) Teaching duration decreases with more domain experience.

(f) Redundancy score decreases overall with more domain experience.

Figure 9. We depict results with respect to domain type (top row) and domain experience (bottom row). We find that abstraction score increases with domain experience and teaching duration decreases with domain experience, and that teaching duration and redundancy score differ across the different domains.

sub-tasks on the robot using the interface. Next, the videos show how to use the sub-tasks to build the recipe for one task in the domain.

Between each domain, the experimenter reset the environment, placing the proper domain's items on the table. After experiencing all four domains, participants take the post-survey questionnaire, comprised of surveys to collect perceived workload and impressions of the robotic agent. Finally, participants answered the post-interview questions.

For each demonstration saved via the interface, we record the robot trajectories along with a third-person perspective video of the participant moving the robot, collected using a Kinect camera. While participants record their sub-tasks, the experimenter takes detailed notes on participant behavior, recording which sub-tasks they record. These notes, along with the interface's saved recipes (i.e., an ordered list of sub-tasks applied to each task in each domain), were used to obtain the abstraction score and redundancy score for each domain. Three coders scored participant abstraction and redundancy scores, resulting in an intra-class correlation coefficient of 0.998 for abstraction scores and 0.755 for redundancy scores.

EPA Experiment Results

We conducted our third user study with 28 participants (39.26% female, mean age = 22.89, standard deviation = 1.63). Before running statistical tests, we first checked that our data met parametric assumptions via Shapiro-Wilk's test and Levene's test. Due to our statistical models not passing tests for normality, we employ non-parametric tests throughout our analysis. We employ Bonferroni correction when applying multiple tests for the same hypothesis to reduce the risk of Type I errors (Schrum, Ghuy, Hedlund-Botti, Natarajan, Johnson and Gombolay 2022). To test RQ1 and RQ4 we employ the Friedman rank sum test, where we report χ^2(degree of freedom) and p-value. For follow-up pairwise comparisons, we employ the Nemenyi Wilcoxon-Wilcox all-pairs test, for which we report the p-value. To test RQ2 and RQ3 we employ Spearman's rank correlation test, where we report ρ and p-value.

RQ 6: In the AUT experiment, we investigate the impact of domain experience on the quality of demonstrations. This hypothesis investigates whether participants can perform zero-shot transfer of knowledge regarding sub-task abstraction, teaching efficiency, and sub-task redundancy to novel domains.

We note that the block touching domain was the demo task, intended to familiarize the participant with the robot and the interface, to isolate the effect of learning in the actual test rounds. As the participants do not observe the optimal demonstration after this demo task, we do not include the block-touching domain in our domain experience.

Abstraction Score – Through a Friedman test, we find a main effect of the participant's domain experience on the participant's domain abstraction score ($\chi^2(3) = 28.056, p < .001$). We conduct pairwise comparisons using a Nemenyi Wilcoxon-Wilcox all-pairs test, visualized in Figure 9d, and find significance between the first and second domain ($p = .006$), the first and third domain ($p = .001$), and the first and fourth domain experienced ($p < .001$).

We first observe in Figure 9d that abstraction scores improve between the first and second domains. This finding points to participants' ability to transfer knowledge about sub-task abstraction zero-shot to a novel domain. We further

observe that abstraction scores improve between the first domain and all subsequent domains. This finding supports our hypothesis that participants improve the level of abstraction of their demonstrations as they gain domain experience.

Our results show that abstraction scores, on average, are monotonically increasing. While the statistically significant improvement in abstraction score occurs after the first domain, the results show a positive trend in subsequent rounds. The diminishing but positive improvement is consistent with prior work finding that human task performance improves logarithmically with practice (Ritter and Schooler 2001).

Teaching Duration – We find significance with respect to teaching duration and domain experience ($\chi^2(3) = 41.796, p < .001$). We find the significant pairs (Figure 9e) to be between the first and second domain ($p = .014$), the first and third domain ($p < .001$), and the first and fourth domain ($p < .001$), as well as between the second and fourth domain ($p = .041$). This finding indicates that participants provide demonstrations more efficiently over time.

Sub-task Redundancy– We find a main effect with respect to learning experience and sub-task redundancy ($\chi^2(3) = 8.018, p = .046$), but find no pairwise significance, (Figure 9f). This finding suggests that there may be a trend between domain experience and sub-task redundancy, but more data are needed.

RQ 7: We investigate the effect of demonstration sub-task abstraction on participants' perceived workload.

Sub-task Count – We perform a Spearman's correlation test and find significance between sub-task count and perceived workload ($\rho = -.519, p = .005$). These findings imply that sub-task count is negatively correlated with perceived workload. High sub-task count means breaking up the task into many smaller sub-tasks, each of which can be reused to avoid redundant demonstrations. One possible explanation of this finding is that fewer sub-tasks for a task indicate more repetitive demonstrations.

RQ 8: We investigate whether participant demographics impact the quality of demonstrations.

Teaching Experience – We find significance between prior teaching experience and sub-task count ($\rho = -.473, p = .011$). This finding is evidence that increased prior teaching experience is negatively correlated with sub-task count. This gained understanding of the impact of prior teaching experience on sub-task count could be used to improve the existing methods employed to teach demonstrators how to provide sufficient demonstrations.

Likeability – We find significance between sub-task count and robot likeability ($\rho = -.501, p = .007$). This finding is evidence that increased robot likeability is negatively correlated with sub-task count.

Agreeableness – Next, we find significance between teaching duration and the agreeableness sub-scale of the Big Five Personality survey ($\rho = .503, p = .006$). This finding is evidence that participant agreeableness is negatively correlated with the efficiency with which they provide demonstrations, namely that more agreeable participants utilize more time to provide demonstrations.

Negative Social Influence – Finally, we find significance between teaching duration and the negative social influence sub-scale of the Negative Attitude towards Robotics survey ($\rho = .577, p = .001$). This finding is evidence that higher teaching duration is correlated to perceptions of negative robot social influence, i.e., participants who are warier of robots take more time to provide demonstrations.

RQ 9: We now investigate whether domain type impacts the quality of demonstrations.

Teaching Duration – Through a Friedman rank sum test, we find significance in the teaching duration among domains ($\chi^2(3) = 8.656, p = .034$). We find one significant pair between table setting and box packing domains ($p = .036$). We plot domain type against teaching time, as seen in Figure 9b.

Sub-task Redundancy – Through a Friedman rank sum test, we find significance in the redundancy score among domains ($\chi^2(3) = 33.836, p < .001$). We find the significant pairs to be between table setting and medicine dispensing ($p = .006$), table setting and soil mixing ($p < .001$), and box packing and soil mixing ($p = .023$) (Figure 9c).

Sub-task Count – Through a Friedman rank sum test, we find significance in the unique sub-task count among domains ($\chi(3)^2 = 45.87, p < .001$). A Nemenyi-Wilcoxon-Wilcox all-pairs test yields significant pairs for table setting and box packing ($p < .001$), table setting and medicine dispensing ($p = .006$), and table setting and soil mixing ($p < .001$).

Abstraction Score – Finally, we note that we find no significance between the abstraction score and domain, as seen in Figure 9a.

EPA Experiment Takeaways

Impact of domain experience on demonstration sufficiency, necessity, and efficiency (RQ7). We find that participant abstraction score is positively impacted by the number of domains experienced ($p < .001$), meaning that over time participants provide demonstrations that manifest higher levels of abstraction. We further find that teaching duration is negatively impacted by the number of domains experienced ($p < .001$). This indicates that over time participants take less time to provide demonstrations.

These findings suggest that participants can generalize knowledge gained about providing demonstrations efficiently, using more sub-task abstraction, from previously experienced domains to a novel domain. **These findings indicate that demonstrators can be trained to efficiently provide sufficient demonstrations to new domains, zero-shot.**

Impact of prior teaching experience on sub-task count (RQ9). We find that prior teaching experience is negatively correlated with sub-task count ($p = .011$), indicating that participants with more teaching experience record fewer sub-tasks. We note that we don't find significance between prior teaching experience and abstraction score or redundancy score. **This finding indicates that increasing teaching experience will increase sub-task efficiency, though not at the expense of sub-task sufficiency or necessity.**

Since general teaching experience does not appear to translate to demonstration quality, our findings highlight the need for a way to teach demonstrators how to provide sufficient and necessary sub-tasks. Our results show that we contribute a scalable and generalizable method for training

LfD demonstrators, by exposing demonstrators to multiple domains in which they practice and observe the optimal teaching method.

Impact of sub-task count on perceived workload (RQ8). We find that participant workload is negatively correlated with their sub-task count ($p = .005$). This indicates that a lower sub-task count correlated with a higher perceived workload. We hypothesize that this is due to the lengthier process of demonstrating and recording under-abstracted sub-tasks. When a demonstrator provides demonstrations with more sub-tasks, while there are more sub-tasks to teach, each is less complex and shorter to demonstrate. On the other hand, when a demonstrator provides demonstrations with fewer sub-tasks, the demonstrations provided are more complex in content and therefore take longer to demonstrate. **In addition to abstractions being useful for robust robot learning, this finding suggests that participants find correct abstractions less effort to teach, as observed via lower perceived workload.**

Impact of robot likeability, participant agreeableness, and negative attitudes on demonstrations (RQ9). We find that robot likeability is negatively correlated with sub-task count ($p = .007$). **This suggests that people rated the robot as more likeable when the teaching was less involved. We hypothesize that this is because when a demonstrator provides demonstrations with more sub-tasks, the processing and recording of this increased number of demonstrations (i.e. which involves pausing the robot, editing the UI takes, etc.) makes teaching more sub-tasks a more involved process. When a demonstrator provides demonstrations with fewer sub-tasks, fewer demonstrations are needed and teaching involvement is lower.**

On the other hand, we find that participant agreeableness is positively correlated with teaching duration ($p = .006$). This finding suggests that demonstrators with higher agreeableness take longer when providing demonstrations, though not at the expense of sub-task count, abstraction score, or redundancy. **This finding indicates that more agreeable demonstrators take their time when recording demonstrations.** We posit this is because these participants either wanted to please the experimenter or because they wanted to be thorough in order to be helpful.

Participants who perceived robots as more socially negative additionally took longer to teach the robot ($p = .001$). **Participants that are warier of robots take more time to provide demonstrations**, therefore we posit that addressing negative robot perceptions will reduce the time people take to teach robots.

Comparing Impact of Prior Experience to Explicit Teaching

In this section we perform post-analysis of the data collected from our second and third user studies. After the PSA experiment, we established that showing an expert video of a task before they teach the robot is sufficient for participants to teach the robot with perfect abstractions. We also noticed that just showing the video of an Analog task, and not letting users teach the Analog task itself, was insufficient in helping users teach the robot perfectly. With the third user study, we tested the confound of the learning effect to demonstrate that participants can teach perfect abstractions to robots as they gain experience in teaching multiple domains. It remains to be established if showing expert videos is better than gaining experience. Moreover, we also wanted to see if there is a trend across studies for participants' performance with respect to the abstraction scores.

In this section we do not compare to the AUT experiment as the Expert Video (**EV**) phase from the PSA experiment performed significantly better than all phases of the AUT experiment. Moreover, only the soil mixture domain is consistent across the PSA and EPA experiments so we restrict analysis to this domain. We also use normalized abstraction scores to avoid any task based scaling issues. Hence, our goal in this section is to determine if domain experience in different, training domains can catch up to the golden standard approach of showing the optimal teaching strategy in the same test domain before a user trains the robot. To compare these two strategies we perform two types of post-analysis –

Performance of Experience in Comparison to Expert Video Tutorial

We aim to see if the performance of the participants across phases in the EPA experiment is equivalent to that of the EV phase of the PSA experiment. For determining equivalence we use Two One-Sided Tests (TOST) between different phases and the EV phase as discussed below. We used non-parametric TOST tests as the scoring data is not normally distributed. Additionally, determining equivalence requires us to *pick* a boundary or threshold for equivalence of the distribution to accept or reject the hypothesis that the observed distribution has shifted by a value smaller than an allowed boundary while maintaining equivalence. By noticing the abstraction scores across different domains we determined that the generalization ability of an abstraction will not change drastically within a $\pm 15\%$ change of the abstraction score. In other words, we stipulate that if two abstractions have a score that is different by $\pm 15\%$ their generalization ability is pretty much equivalent. Given this observation, we picked the boundary to establish the equivalence of distribution of abstraction scores across any two different phases *conservatively* to be 10% of the abstraction scores.

In the following sub-sections, we compare each of the phases (phase 1-3) of the EPA study with the EV phase of the PSA study. We use a Wilcoxon rank sum test with continuity correction, a null hypothesis significance test (NHST), and an equivalence test via two one-sided tests (TOST) with a significance level of $\alpha = 0.05$. These tested the null hypotheses that true location shift is equal to 0 (NHST), and true location shift is more extreme than -0.1 and 0.1 (TOST). The results are also summarized in Figure 11.

Phase 1 EPA vs EV PSA – The equivalence test was not significant (p = 1). The NHST was significant, as observed via the Wilcox Mann Whitney test ($W = 199, p < 0.001$) location shift = 0.786 90% C.I.[0.723, 0.83]; Rank-Biserial Correlation = 0.843 90% C.I.[0.693, 0.923]). We cannot reject the null hypothesis of TOST.

We further confirm this significant difference with a Wilcoxon two-sided rank sum test ($W = 199, p < 0.001$).

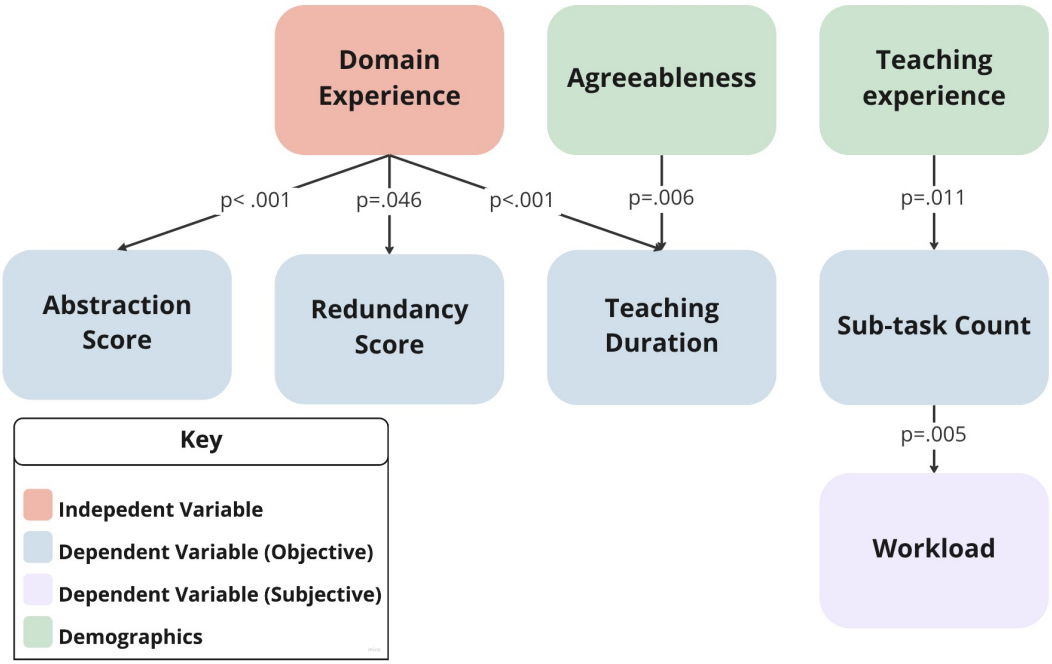


Figure 10. Depicted is a summary of the significant results of the EPA experiment. We find a main effect of the participant's domain experience on the participant's domain abstraction score, redundancy score, and teaching duration. Regarding demographic factors, we find a main effect of agreeableness on teaching duration, and of teaching experience on sub-task count. Finally, we find a main effect of sub-task count on workload.

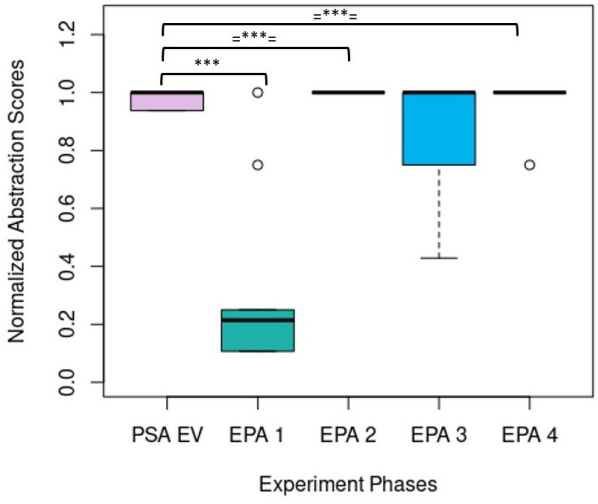


Figure 11. Box plot indicating the abstraction score distributions across phases of the second and third experiments. We limit our comparison to the gold-standard of our PSA experiment, which was expert video based tutoring of participants to teach perfect abstractions. We notice that in our third study, participants start of performing significantly worse, but with experience catch up to the performance of an expert video based tutoring. With "=***=" we indicate statistically significant equivalence results with the two one-sided test results between the PSA experiment's EV phase and Round 2 and Round 4 of the EPA experiment. With "***" we indicate that the PSA experiment's EV phase and Round 1 of the EPA experiment are statistically significantly different. Which means that participants in the EPA study start with teaching poor abstractions and get better over the study as they teach the robot different domains. Our third experiment demonstrates that with teaching experience across domains participants can teach novel usable planning abstractions to robots.

We find that if participants encounter the the soil mixture domain as the first round of robot teaching task within the EPA study, the performance of the participants in providing demonstrations is not similar or equal to the performance of users who have seen experts teach the robot in the same domain.

Phase 2 EPA vs EV PSA – The equivalence test was significant, as observed via the Wilcox Mann Whitney test ($W = 192, p < 0.001$) with location shift = -7.87e-05 90% C.I.[-0.0624, 0] and Rank-Biserial Correlation = -0.417 90% C.I.[-0.684, -0.0499]). We reject the null hypothesis of TOST with significance ($p < 0.001$) that the difference between the means and the 95% confidence intervals of the distributions are greater than 0.1 normalized abstraction scores; this result supports the hypothesis that the difference is relatively small.

We find that if participants encounter the soil mixture domain in the second round of teaching tasks to robots within the EPA study the performance of the participants in providing demonstrations is equivalent or statistically indistinguishable from the EV domain within the PSA study. This is a noteworthy observation as given the teaching experience of a different task and its corresponding expert solution presented after, the participants can learn to provide demonstrations as well as if they had learned to provide demonstrations within the same task from an expert.

Phase 3 EPA vs EV PSA – Both the equivalence test (p = 0.064), and the NHST (p = 0.797) were not significant (location shift = 9.18e-06 90% C.I.[-2.38e-05, 0.187]; Rank-Biserial Correlation = 0.056 90% C.I.[-0.311, 0.407]). We also checked with a regular Wilcoxon two-sided rank sum-test, and found that no significant difference can be established between the two distributions ($W = 114, p > 0.05$).

The results are inconclusive: neither null hypothesis can be rejected. We find that if participants encounter the soil mixture domain in the third round of teaching tasks to robots within the EPA study the performance of the participants in providing demonstrations cannot with certainty be distinguished as being equivalent or different from the EV domain within the PSA study. The variance of the abstraction scores in third round of the EPA study is quite high making the equivalence hard to establish. The variance is not high enough to establish non-equivalence either. We believe more samples in the EPA study are required to solve this question one way or another.

Phase 4 EPA vs EV PSA – These tested the null hypotheses that the true location shift is equal to 0 (NHST), and true location shift is more extreme than -0.1 and 0.1 (TOST). The equivalence test was significant, as observed via the Wilcox Mann Whitney test ($W = 24, p = 0.001$) with location shift = -4.05e-06 90% C.I.[-0.0624, 5.76e-05]; Rank-Biserial Correlation = -0.214 90% C.I.[-0.56, 0.194]). We reject the null hypothesis of TOST with significance ($p < 0.001$) that the difference between the means and the 95% confidence intervals of the distributions are greater than 0.1 normalized abstraction scores of each other, which supports the hypothesis that these distributions are similar.

We find that if participants encounter the soil mixture domain in the fourth round of teaching tasks to robots within the EPA study the performance of the participants in providing demonstrations is equivalent or statistically indistinguishable from the EV domain within the PSA study. This is again an important observation as given a few teaching experiences of a different tasks and their corresponding expert solutions presented after, the participants on their own can learn to provide demonstrations as well as if they had learned to provide demonstrations within the same task from an expert.

These results bolster our claim that non-expert end-users can teach robots novel tasks using plannable TAMP abstractions *without* an expert's guidance in a *novel* domain (i.e., zero shot) just as well as an end-users who have access to a video demonstration of an expert teaching in that novel domain. This result is a tremendous boon for democratizing robot LfD as it shows that end-users can accrue and transfer experience teaching to new domains without the need for ad hoc tutoring from expert teachers.

Impact of Prior Experience and Explicit Teaching on Abstraction Scores

Next, we compare the impact of domain experience on abstraction score with the impact of explicit teaching on abstraction scores in the Soil Mixture domain across the two experiment of PSA and EPA. We ran a gamma-distributed generalized linear mixed effects regression model with an identity link function to model abstraction scores given teaching round, and type of study as independent variables with mixed effects while using subject ID as an independent noise variable. We find that the performance of the participants improves significantly as they experience more rounds across studies as ($F(1, 129) = 147.58, p < 0.001$). We found that subject's abstraction scores were correlated with significance based the study (PSA vs EPA) they were performing ($F(1, 129) = 5.63, p < 0.05$). The implication here being subjects in general performed better in the EPA study where they have been shown to score equivalent to the highest scoring phase of the PSA study in multiple rounds. Moreover, there is a mixed effect where the round of the study and the study itself correlate with the performance of the participants in providing good abstractions to robots ($F(1, 129) = 17.95, p < 0.001$).

The key takeaway here is that across two studies we notice that experience in teaching the robot is a significant predictor of improved performance in providing good planning abstractions to the robot and thereby scoring higher abstraction scores. Moreover, the EPA study which presents multiple domains to users, without showing expert training strategies at first, allows users to train the robot better than the PSA domain in general. These results demonstrate that we can teach novice users to train robots given training videos for *some* tasks and allowing users to train these tasks robots. The users will generalize to novel unseen tasks and provide useful, generalizable and plannable demonstrations to the robot. This provides novel opportunities in end-user house-hold robotics where users might have to teach novel tasks to the robot safely in real time.

Related Work

In this section we cover a broad literature review that covers the ability of people to create and teach abstractions to other people or robots.

Cognitive Studies About Abstractions Created by Humans

Cognitive science has investigated the problem of abstraction to understand how and when people create abstractions. Pioneering work by Lashley (1951) argued with examples for order and hierarchy in the performance of complex behaviors. Lashley argued that humans are not performing complex behaviors by moving from one skill to the next based on the sensory inputs alone, but that they have a "primed" pre-determined order or a plan in which they want to execute the skills. Moreover, Lashley argued that such primed sequencing of skills can also be altered if the environment requires it. There has been follow-up work demonstrating such planning and organization of skills with experiments in different domains (Klapp 1977; Rosenbaum, Cohen, Jax, Weiss and van der Wel 2007). Apart from sequencing of skills, Aarts and Dijksterhuis (2000) demonstrated that just specifying a goal condition to people activates skills related to that goal. For example, thinking about bicycles, when the task of going to university is specified, pointing again to organization of skills given a goal condition. These skills can be activated in novel contexts as well, such as the act of cleaning when people smell cleaning products while doing unrelated and novel chores (Holland, Hendriks and Aarts 2005).

There is also the related question of how these skills and hierarchies are learned, which is a much harder question to answer. Multiple physiological studies have demonstrated changes in the brain's structure and morphology during skill learning (Lövdén, Garzón and Lindenberger 2020; Wenger, Brozzoli, Lindenberger and Lövdén 2017; Kolasinski, Hinson, Zand, Rizov, Emir and Stagg 2018). A clear predictive model of these changes across different skills and periods of learning still remains elusive (Wenger, Brozzoli, Lindenberger and

Lövdén 2017). From a cognitive perspective, Solway, Diuk, Córdova, Yee, Barto, Niv and Botvinick (2014) have shown that people can learn an *optimal behavior hierarchy* given a novel task based on observed data. The optimality here is being defined with respect to the goals of a task which in the case of Solway, Diuk, Córdova, Yee, Barto, Niv and Botvinick is a routing task. Other studies have demonstrated that humans have an innate ability to identify "breakpoints" (Newtson and Engquist 1976), or "chunks" within videos or lexical tasks (Orbán, Fiser, Aslin and Lengyel 2008; Brady, Konkle and Alvarez 2009). Such chunking or breaking of events in videos or demonstrations in the physical world can explain how humans can create skill segments by observing others or while performing the tasks themselves. These skills can then allow humans or other animals to have a primed high-level plan as pointed out by Lashley (1951). Our work is inspired by cognitive studies testing for abstractions humans create such as those in Solway, Diuk, Córdova, Yee, Barto, Niv and Botvinick (2014); Newtson and Engquist (1976); however, our goals are to see if users can be taught to create abstractions that are useful for robots rather than themselves. The previous studies do not demonstrate if humans can create abstractions useful for robots in task solving. This is an important question to consider as household robots might need to be taught novel tasks as technology or a user's requirements change. This might have implications for the performance of teachers when teaching agents using abstractions, as they might spend too much time thinking about what type of abstractions to use.

Abstractions in Robotics and Reinforcement Learning

Learning hierarchical abstraction is a large challenge in both reinforcement learning and robot learning. Hierarchical abstractions have multiple properties that make them ideal for the robot learning problem. They can allow efficient exploration by allowing the agent to quickly reach hard-to-reach states using previously learned abstractions (Konidaris 2019). There are also computational savings to be considered as the agent can reason over hierarchies rather than consider every possible state-action pair in the environment (Gopalan, desJardins, Littman, MacGlashan, Squire, Tellex, Winder and Wong 2017; Jong, Hester and Stone 2008).

In reinforcement learning, multiple approaches have been proposed to learn abstractions and to solve novel tasks with learned abstractions. There are two major types of abstractions that an agent can learn: State abstractions or Action abstractions (Konidaris 2019). An agent abstracts over states when it can combine multiple states into one large *abstract* state and then reason over the set of abstract states. State abstractions have been classically used in navigation problems where an agent can abstract all locations within a room, building, or city to belong to a single abstract state (Abel, Arumugam, Lehnert and Littman 2018; Li, Walsh and Littman 2006; Van Roy 2006). Action abstraction, on the other hand, is over sequences of low-level actions (Lioutikov, Neumann, Maeda and Peters 2015; Muelling, Kober and Peters 2010; Konidaris and Barto 2009). Consider a sequence of torques that move a robot's arm so it can pick an object. These sequences of torques are harder to reason over, and can be abstracted into a single temporally extended action to "pick can." Some approaches combine both State and Action abstractions to demonstrate sample efficiency of computational savings in task solving (Dietterich 2000; Gopalan, desJardins, Littman, MacGlashan, Squire, Tellex, Winder and Wong 2017; Gopalan, Rosen, Konidaris and Tellex 2020; Konidaris, Kaelbling and Lozano-Perez 2014).

A formalism that stands out in reinforcement learning settings is the Options formalism (Sutton, Precup and Singh 1999). An Option is a triple: $\{\mathcal{I}, \mathcal{T}, \pi\}$. Where initiation condition $\mathcal{I} : \mathcal{S} \rightarrow [0, 1]$ defines the set of states where the skill can begin; termination condition $\mathcal{T} : \mathcal{S} \rightarrow [0, 1]$, defines the set of states where the option terminates; and $\pi : \mathcal{S} \longrightarrow Pr(\mathcal{A})$ defines the actions an agent takes from each state. The options formalism provides the ability to learn the state and/or action abstraction. By learning the initiation of termination sets the agent will learn a state abstraction, and by learning a policy between them the agent will learn an action abstraction. Robot skill learning problems have been previously posed as options learning problems (Konidaris and Barto 2009; Lioutikov, Neumann, Maeda and Peters 2015; Muelling, Kober and Peters 2010). Such skills are especially useful for dynamic tasks such as playing table tennis (Muelling, Kober and Peters 2010).

The focus of this paper however is specifically on Task and Motion Planning (TAMP) abstractions for robotics (Garrett, Chitnis, Holladay, Kim, Silver, Kaelbling and Lozano-Pérez 2021). A TAMP problem is defined over both continuous and discrete variables, which are common in the real world and robotics domains. Here a robot might have to reason over discrete objects and goal conditions, such as stacking plates. However, the reasoning for these discrete goals needs to be done over continuous variables of robot poses and continuous collision constraints. Alami, Simeon and Laumond (1990) and Siméon, Laumond, Cortés and Sahbani (2004) presented some of the earliest works describing this interplay of discrete and continuous reasoning a robot needs to perform. A large challenge in TAMP settings is the computational efficiency of optimal planning in domains with continuous and discrete state variables (Alami, Simeon and Laumond 1990). TAMP provides a robust framework to solve long-horizon tasks and has been demonstrated as the framework of choice across several mobile manipulation domains (Lozano-Perez and Kaelbling 2014; Srivastava, Fang, Riano, Chitnis, Russell and Abbeel 2014; Kaelbling and Lozano-Pérez 2010; Stilman 2007; Dantam, Kingston, Chaudhuri and Kavraki 2018). Its robustness is one of the biggest reasons why we chose TAMP as the formalism of choice to learn within this work.

There have been algorithmic approaches to learning tasks from user demonstrations without requiring the demonstrations to specify task abstractions (Konidaris, Kuindersma, Grupen and Barto 2012; Kim, Lee and Kim 2018; Cobo, Isbell and Thomaz 2012). However, these methods usually extract tasks from low-dimensional torque and environment data with sophisticated statistical techniques that are computationally expensive and might not scale to novel environments and tasks found in different households and factory floors. Instead, we seek to empower end-users to train these robots using sub-tasks. People can remove this computational bottleneck and provide correct sub-tasks as they have better generalization capabilities than modern robotics techniques. Hence, we test whether the users are

equipped to provide such demonstrations, and what type of priming or tutoring would elicit demonstrations using sub-tasks that help the robot to generalize to novel task specifications while keeping in mind that the robot is going to solve a TAMP (Garrett, Chitnis, Holladay, Kim, Silver, Kaelbling and Lozano-Pérez 2021) problem, and that the sub-tasks specified should be usable by a TAMP formalism.

Human-Interactive Robot Teaching Strategies

Learning from demonstration (LfD) is a ubiquitous approach for enabling humans to program robots to perform new skills via human task demonstrations (Ploeger, Lutter and Peters 2020; Chen, Paleja and Gombolay 2020; Ho and Ermon 2016; Argall, Chernova, Veloso and Browning 2009). Prior work in LfD has learned impressive dynamic skills on the robot (Ploeger, Lutter and Peters 2020; Chen, Paleja and Gombolay 2020), and the ability to play high-dimensional games (Samvelyan, Rashid, Witt, Farquhar, Nardelli, Rudner, Hung, Torr, Foerster and Whiteson 2019). These approaches generally attempt to either directly model the robot's unknown policy (Ho and Ermon 2016) or infer the robot's latent reward function (Abbeel and Ng 2004; Fu, Luo and Levine 2018; Ziebart, Maas, Bagnell and Dey 2008). Some LfD approaches attempt to acknowledge the way humans teach tasks by modeling feedback more accurately (MacGlashan, Ho, Loftin, Peng, Wang, Roberts, Taylor and Littman 2017; Knox and Stone 2009). However, these works have not addressed the question of whether people teach agents tasks using an abstraction hierarchy.

In the human robot interaction (HRI) community, significant research has shown that people can teach abstractions, sub-tasks or otherwise, when tutored to teach the exact same task (Cakmak and Takayama 2014*a*; Mohseni-Kabir, Rich, Chernova, Sidner and Miller 2015). Cakmak and Takayama (2014*a*) attempt to teach keyframe-based abstractions when subjects are shown a video tutorial of a task. Mohseni-Kabir, Rich, Chernova, Sidner and Miller (2015) attempt to teach hierarchical task networks from human feedback with a well-designed interface and training to use the interface. Other works have attempted to teach task relevant features to a robot with strategies such as active learning to query demonstrators (Bajcsy, Losey, O'Malley and Dragan 2018; Bullard, Chernova and Thomaz 2018). Multiple works have shown that novice users can learn to use their interface and teaching paradigms effectively to train the agent with novel algorithms (Paxton, Hundt, Jonathan, Guerin and Hager 2017; Orendt, Fichtner and Henrich 2016; Mollard, Munzer, Baisero, Toussaint and Lopes 2015). In all of these works establishing a human demonstrator's ability to provide usable demonstrations that contain abstractions, the participants are shown precisely how to teach the robot. They are then asked to reproduce the method of robot teaching that was prescribed.

Some work has attempted to understand strategies to provide instructions to users so they can teach tasks to robots or machine learning algorithms. Cakmak and Takayama (2014*b*) compare written and video demonstrator instruction, and find that trial and error plays a large role in the learning process; we note that these demonstrators learn and are evaluated on the same task such as classifying animals into different categories. Teaching a robot using abstractions without this guidance is not intuitive to non-experts (Knaust and Koert 2021). In this work, we investigate whether demonstrators' ability to provide sufficient sub-task abstractions improves over time, as they practice providing demonstrations in multiple different domains.

Teaching tasks to robots is akin to programming a computer that can manipulate the physical world. Humans have been teaching each other to code and think computationally for a while (Hsu, Chang and Hung 2018; Lu and Fletcher 2009). Just as coding requires a programmer to think computationally, and design the code using functions, robot programming requires the user to think about the physical consequences of the skills they teach robots. Moreover, similar to functional reuse, a robot can have skills that it can repeatedly use to solve multiple tasks. This similarity has been noticed by works that refer to robot learning from demonstration as Programming by Demonstration (PbD) (Billard, Calinon, Dillmann and Schaal 2008; Calinon 2009; Alexandrova, Cakmak, Hsiao and Takayama 2014; Fischer, Kirstein, Jensen, Krüger, Kukliński, aus der Wieschen and Savarimuthu 2016). While these approaches have attempted to program tasks on robots, they have not tested the ability of users to create task and motion planning abstractions for novel task-solving on robots from scratch. Moreover, it is not clear what type of interactions allow users to program robots efficiently.

Other techniques orthogonal to our approach are those that use Large Language Models to specify task plans for robots to solve TAMP problems (Huang, Xia, Xiao, Chan, Liang, Florence, Zeng, Tompson, Mordatch, Chebotar, Sermanet, Brown, Jackson, Luu, Levine, Hausman and Ichter 2022; Ahn, Brohan, Brown, Chebotar, Cortes, David, Finn, Gopalakrishnan, Hausman, Herzog, Ho, Hsu, Ibarz, Ichter, Irpan, Jang, Ruano, Jeffrey, Jesmonth, Joshi, Julian, Kalashnikov, Kuang, Lee, Levine, Lu, Luu, Parada, Pastor, Quiambao, Rao, Rettinghouse, Reyes, Sermanet, Sievers, Tan, Toshev, Vanhoucke, Xia, Xiao, Xu, Xu and Yan 2022; Shah, Equi, Osinski, Xia, Ichter and Levine 2023). This area of research is novel and it aims to split large problems into smaller skills that the robot can solve. Similar methods have been extended to large vision and language models (Brohan, Brown, Carbajal, Chebotar, Choromanski, Ding, Driess, Finn, Florence, Fu, Arenas, Gopalakrishnan, Han, Hausman, Herzog, Hsu, Ichter, Irpan, Joshi, Julian, Kalashnikov, Kuang, Leal, Levine, Michalewski, Mordatch, Pertsch, Rao, Reymann, Ryoo, Salazar, Sanketi, Sermanet, Singh, Singh, Soricut, Tran, Vanhoucke, Vuong, Wahid, Welker, Wohlhart, Xiao, Yu and Zitkovich 2023; Stone, Xiao, Lu, Gopalakrishnan, Lee, Vuong, Wohlhart, Zitkovich, Xia, Finn and Hausman 2023). These methods allow general usability robots via language-based task specifications. The large language models allow task splitting akin to action abstraction and the visual models allow state abstractions for robots. All of these methods depend on pre-trained skills whereas in our work we want to see if novice users can specify skills a robot can use to solve future tasks. Robot skill and task learning is a challenging problem and we are attempting to see if users can teach skills or sub-tasks from scratch to robots. We believe our approach can enable interactive approaches with pre-trained models where robots can learn novel tasks from novice users using language.

We note that our focus is not explicitly on user interface design unlike previous works (Cakmak and Takayama 2014*a*; Mohseni-Kabir, Rich, Chernova, Sidner and Miller 2015; Orendt, Fichtner and Henrich 2016; Paxton, Hundt, Jonathan, Guerin and Hager 2017). Rather, we are keen on investigating priming mechanisms and teaching guides to help users teach useful sub-task-based abstractions given a sufficient interface.

Limitations and Community Feedback

Ours is the first work that attempts to ground Task and Motion Planning abstractions for robot planning from novice users. We wanted to understand if this work reflects the quality of abstractions specified by the Task and Motion Planning Community. Moreover, we wanted to understand if the studies themselves reflected accurate end-user expectations as experienced by human-robot interaction researchers. We setup multiple meetings with researchers in the robotics community representing both of these fields (outside of our labs). We interviewed them about the limitations of our work and about the generalizability of our approach to enable novice users to teach TAMP abstractions. Our line of questioning was to find limitations of our work with respect to the type of abstractions learned, the generalizability of the said abstractions, and the assumptions made by our work when conducting the user studies. We have compiled the feedback of the community on the limitations of this work along with those we knew on our own. There are two different types limitations to our approach: the technical limitations and limitations in study design.

Technical Limitations Firstly, multiple expert roboticists pointed out that the approach compels participants or end-users to adopt a particular teaching philosophy. Specifically, we expect users to provide mode transition-based sub-goals to the robot to create TAMP abstractions. While this approach is suitable for the robot, it might not be the best demonstration or teaching approach for the users. We employed this technique because it allows us to scale up the teaching without requiring the users to provide a large number of demonstrations per task. However, truly intelligent machines or robots should (hypothetically) be capable of using any teaching approach a human might use to extract TAMP or other abstractions the robot needs with a few demonstrations. This is especially true about our study demonstrating participants are unable to provide any abstraction. We also note that expecting users of a robot to program them with ideas such as functional abstraction is challenging, as evidenced by our results. As such, we argue that robots need better and more intuitive methods of providing task-level demonstrations including language, gestures, and imitation from observation.

Another major technical limitation pointed out to us is that the robotic agent is only learning action abstractions or sub-task level abstractions with teaching methods examined in this work. These are not the same as *state* abstractions. We do not have the ability to explain to the robot that "this room is a kitchen" with our techniques. State and Action abstractions are both important in task solving depending on the type of problem that a robot needs to solve (Konidaris 2019). Without state-level abstractions describing low-level constraints such as collisions with the table that the robot is operating on is challenging.

In this work the age distributions of our participants is limited in scope, and is primarily composed of college students. In future work, we propose to validate our results with a larger more representative population pool. In future work we also propose to examine approaches that allow users to teach such low-level constraints and state abstractions in the future. This would include improvements not just in the learning methods but also over interfaces used to teach tasks to robots. Moreover, it will be interesting to see how we can combine human-taught abstractions with those learned by neural networks from large scale data (Brohan, Brown, Carbajal, Chebotar, Choromanski, Ding, Driess, Finn, Florence, Fu, Arenas, Gopalakrishnan, Han, Hausman, Herzog, Hsu, Ichter, Irpan, Joshi, Julian, Kalashnikov, Kuang, Leal, Levine, Michalewski, Mordatch, Pertsch, Rao, Reymann, Ryoo, Salazar, Sanketi, Sermanet, Singh, Singh, Soricut, Tran, Vanhoucke, Vuong, Wahid, Welker, Wohlhart, Xiao, Yu and Zitkovich 2023; Stone, Xiao, Lu, Gopalakrishnan, Lee, Vuong, Wohlhart, Zitkovich, Xia, Finn and Hausman 2023). Our representations are not generalizable outside of the tasks taught, but their sample complexity allows tasks being taught by humans within their homes and offices. The existing experiments would change drastically if we looked at the capability to teach and add TAMP abstractions to a large scale pre-trained representation.

Study Design Limitations We further acknowledge the limitations of our study design. A major challenge we faced was that we did not have an unlimited amount of time for users to interact with the robot to fully mitigate novelty effects. At first, users are often apprehensive when learning to control a robot. A longer duration of interaction with the robot and LfD interface would have mitigated issues relating to novelty, such as the time pressure to finish a study within a given duration and the limited number of times a user's various teaching attempts can be trialed. Ultimately, giving users more control over their acclimation experience would help users explore their preferred methods of teaching robots robustly. We partially explored these novelty issues by allowing users to teach different tasks in our EPA study, but longer-duration longitudinal user studies are required in robotics where users are away from lab settings. We also note that our measure of IQ is imperfect, as it was performed with an open-source, online test rather than a trained, in-person examiner.

We created the teaching experience questionnaire as prior work had yet to develop a questionnaire to measure teaching experience, however, we agree that ideally the development and validation of a scale should be published separately from the use of the scale. While ad hoc scales are often used in HRI (Sartori and Bocca 2023; Gottardi, Tortora, Tosello and Menegatti 2022), we agree that results based upon ad hoc scales should be taken with a grain of salt (Schrum, Johnson, Ghuy and Gombolay 2020). We propose to validate this scale in future work.

Finally, we note that we could have considered priming strategies for the detailed written or visual guide that would have clarified the goals of the demonstrations, in other words instructions that would have optimized for high-quality demonstrations specific to the domain or user's teaching style. In a significantly longer experiment, we could have designed a variety of training videos of varying durations

and pedagogical approaches – each of which could have served as its experimental condition – to find a hierarchy of training videos towards one that is most optimal. However, this would be impractical. In our experiment, we do not assume prior knowledge of robotics from our participants. Designing a video to provide participants with expert-level performance would have required more in-depth, long-form explanations of all the information needed to be an optimal demonstrator. Such a design was not feasible for our cross-sectional experiment.

Open Problems

There are many open problems that our work points towards. Firstly, we need to conduct long-horizon user studies where participants teach tasks and function with a robot over multiple interactions over extended periods of time. This is challenging in a laboratory setting as it is challenging to impose on participants to come back to robot labs over longer horizons. An approach that robotics might have to follow is sending robots home with people so they can work with them in the comfort of their homes. What tasks to teach and how to ensure that robots are functional when not in labs are other challenges that such approaches might face, however, true democratization of programmable robots in households would require such tests to be conducted.

Secondly, more work needs to be done in the development of user interfaces to control robots. We developed multiple user interfaces with rapid prototyping in our work to allow users to teach tasks to robots. However, these interfaces are not general-purpose enough. Our community is lacking a common interface to teach and interact with different types of hardware platforms. This makes the adoption of robotics challenging outside of laboratory settings.

Finally, we need to develop sample efficient learning from demonstration approaches for learning robot skills. Current approaches to learning pre-trained skill embeddings might prove promising in this regard (Brohan, Brown, Carbajal, Chebotar, Choromanski, Ding, Driess, Finn, Florence, Fu, Arenas, Gopalakrishnan, Han, Hausman, Herzog, Hsu, Ichter, Irpan, Joshi, Julian, Kalashnikov, Kuang, Leal, Levine, Michalewski, Mordatch, Pertsch, Rao, Reymann, Ryoo, Salazar, Sanketi, Sermanet, Singh, Singh, Soricut, Tran, Vanhoucke, Vuong, Wahid, Welker, Wohlhart, Xiao, Yu and Zitkovich 2023). However, the approaches need to be general enough to work on different hardware platforms. TAMP provides a bridge here where not everything needs to be learned as the skills between sub-tasks can be planned with a low-level motion planner. Nevertheless, dynamic skills such as chopping vegetables, hitting a ball, etc., require a robot to learn skills. Learning skills in a sample efficient manner would allow users to teach robots personalized skills such as cutting vegetables with a specific technique inside their homes.

Conclusion

In this work, we investigate the performance of novice users in teaching tasks level planning abstractions to robots. While previous LfD work has attempted to teach robots novel behavior using trajectories, we investigate the ability of users to teach Task and Motion Planning (TAMP) based abstractions to the robot for novel tasks. We use TAMP abstractions here as they are the representation of choice for robots to solve long-horizon, multi-task problems. We conduct three novel human-subjects experiments to answer (1) what are the necessary conditions to teach users through hierarchy and task abstractions; (2) what instructional information or feedback is necessary to support users to learn to program robots effectively to solve novel tasks; (3) how does experience teaching the robot help users teach novel tasks with useful abstractions. Our first experiment shows that fewer than half (35.71%) of our subjects provide demonstrations with sub-task abstractions when not primed. Our second experiment demonstrates that users fail to teach the robot correctly when not shown a video demonstration of an expert's teaching strategy for the exact task that the subject is training. Not even showing an expert training video of an analog task was sufficient. As both of these previous experiments failed to study the learning effect of experience obtained through multiple domains, we created a third experiment where we find that increasing participant experience with providing demonstrations improves their demonstration's degree of sub-task abstraction ($p < .001$), teaching efficiency ($p < .001$), and sub-task redundancy ($p < .05$) in novel domains. We find that experience performs as well as providing an expert video in enabling users to provide demonstrations with useful plannable abstractions to the robot, which is backed by our post-analysis across studies ($p < .001$). These experiments together investigate if and how users provide demonstrations with useful abstractions to robots even in novel task settings. Our experiments reveal a need for fundamentally different approaches in LfD that can allow end-users to teach generalizable long-horizon tasks to robots without the need to be coached by experts at every step. We address this need with a series of training domains that enable novice users to learn to provide demonstrations with plannable abstractions in novel domains.

Acknowledgements

We want to thank Siddharth Srivastava, Tom Silver, Harish Ravichandar, Eric Rosen, and Rachel Holladay for their feedback on our work allowing us to write a thorough limitations section.

This work was sponsored by MIT Lincoln Laboratory (7000437192), NASA Early Career Fellowship (80HQTR19NOA01-19ECF-B1), the National Science Foundation (20-604, IIS-2112633, and IIS-2340177), and a gift from Konica Minolta, Inc. The content of this work is solely the responsibility of the authors and does not necessarily represent the official views of our sponsors.

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-24-1-0239. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

References

Aarts, H. and Dijksterhuis, A. (2000), 'Habits as knowledge structures: automaticity in goal-directed behavior.', *Journal of personality and social psychology* **78 1**, 53–63.

Abbeel, P. and Ng, A. (2004), 'Apprenticeship learning via inverse reinforcement learning', *Proceedings of the twenty-first international conference on Machine learning* .

Abel, D., Arumugam, D., Lehnert, L. and Littman, M. (2018), State abstractions for lifelong reinforcement learning, *in* J. Dy and A. Krause, eds, 'Proceedings of the 35th International Conference on Machine Learning', Vol. 80 of *Proceedings of Machine Learning Research*, PMLR, pp. 10–19.

Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Ho, D., Hsu, J., Ibarz, J., Ichter, B., Irpan, A., Jang, E., Ruano, R. J., Jeffrey, K., Jesmonth, S., Joshi, N. J., Julian, R. C., Kalashnikov, D., Kuang, Y., Lee, K.-H., Levine, S., Lu, Y., Luu, L., Parada, C., Pastor, P., Quiambao, J., Rao, K., Rettinghouse, J., Reyes, D. M., Sermanet, P., Sievers, N., Tan, C., Toshev, A., Vanhoucke, V., Xia, F., Xiao, T., Xu, P., Xu, S. and Yan, M. (2022), Do as i can, not as i say: Grounding language in robotic affordances, *in* 'Conference on Robot Learning'.

Akgun, B., Cakmak, M., Jiang, K. and Thomaz, A. L. (2012), 'Keyframe-based learning from demonstration', *International Journal of Social Robotics* **4**(4), 343–355.

Alami, R., Laumond, J.-P. and Siméon, T. (1995), Two manipulation planning algorithms, *in* 'Proceedings of the Workshop on Algorithmic Foundations of Robotics'.

Alami, R., Simeon, T. and Laumond, J.-P. (1990), A geometrical approach to planning manipulation tasks. the case of discrete placements and grasps, *in* 'The fifth international symposium on Robotics research', MIT Press, pp. 453–463.

Alami, R., Siméon, T. and Laumond, J.-P. (1991), A geometrical approach to planning manipulation tasks. the case of discrete placements and grasps, *in* 'International Symposium on Robotics Research'.

Alexandrova, S., Cakmak, M., Hsiao, K. and Takayama, L. (2014), Robot programming by demonstration with interactive action visualizations., *in* 'Robotics: science and systems', pp. 1–9.

Argall, B. D., Chernova, S., Veloso, M. and Browning, B. (2009), 'A survey of robot learning from demonstration', *Robotics and autonomous systems* **57**(5), 469–483.

Bajcsy, A., Losey, D. P., O'Malley, M. K. and Dragan, A. D. (2018), Learning from physical human corrections, one feature at a time, *in* 'Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction', pp. 141–149.

Bartneck, C., Kulić, D., Croft, E. and Zoghbi, S. (2009), 'Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots', *International journal of social robotics* **1**(1), 71–81.

Billard, A., Calinon, S., Dillmann, R. and Schaal, S. (2008), Robot programming by demonstration, *in* 'Springer handbook of robotics', Springer, pp. 1371–1394.

Brady, T. F., Konkle, T. and Alvarez, G. A. (2009), 'Compression in visual working memory: using statistical regularities to form more efficient memory representations.', *Journal of experimental psychology. General* **138 4**, 487–502.

Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Choromanski, K., Ding, T., Driess, D., Finn, C., Florence, P. R., Fu, C., Arenas, M. G., Gopalakrishnan, K., Han, K., Hausman, K., Herzog, A., Hsu, J., Ichter, B., Irpan, A., Joshi, N. J., Julian, R. C., Kalashnikov, D., Kuang, Y., Leal, I., Levine, S., Michalewski, H., Mordatch, I., Pertsch, K., Rao, K., Reymann, K., Ryoo, M. S., Salazar, G., Sanketi, P. R., Sermanet, P., Singh, J., Singh, A., Soricut, R., Tran, H., Vanhoucke, V., Vuong, Q. H., Wahid, A., Welker, S., Wohlhart, P., Xiao, T., Yu, T. and Zitkovich, B. (2023), 'Rt-2: Vision-language-action models transfer web knowledge to robotic control', *ArXiv* **abs/2307.15818**.

Bullard, K., Chernova, S. and Thomaz, A. L. (2018), Human-driven feature selection for a robotic agent learning classification tasks from demonstration, *in* '2018 IEEE International Conference on Robotics and Automation (ICRA)', IEEE, pp. 6923–6930.

Cakmak, M. and Takayama, L. (2014*a*), Teaching people how to teach robots: The effect of instructional materials and dialog design, *in* 'Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction', pp. 431–438.

Cakmak, M. and Takayama, L. (2014*b*), 'Teaching people how to teach robots: The effect of instructional materials and dialog design', *2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* pp. 431–438.

Calinon, S. (2009), *Robot programming by demonstration*, EPFL Press.

Campeau-Lecours, A., Maheu, V., Lepage, S., Lamontagne, H., Latour, S., Paquet, L. and Hardie, N. (2016), Jaco assistive robotic device: Empowering people with disabilities through innovative algorithms, *in* 'Rehabilitation Engineering and Assistive Technology Society of North America (RESNA) Annual Conference'.

Caruana, R. (1998), *Multitask learning*, Springer.

Chen, L., Paleja, R. and Gombolay, M. (2020), Learning from suboptimal demonstration via self-supervised reward regression, *in* 'Proceedings of the Conference on Robot Learning'.

Cobo, L. C., Isbell, C. L. and Thomaz, A. L. (2012), Automatic task decomposition and state abstraction from demonstration, *in* 'Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1', AAMAS '12, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, p. 483–490.

Dantam, N. T., Kingston, Z. K., Chaudhuri, S. and Kavraki, L. E. (2018), 'An incremental constraint-based framework for task and motion planning', *The International Journal of Robotics Research* **37**, 1134 – 1151.

Dietterich, T. G. (2000), 'Hierarchical reinforcement learning with the maxq value function decomposition', *Journal of artificial intelligence research* **13**, 227–303.

Fatemi, M. A., Ganjali, R. and Kafi, Z. (2016), 'Efl teachers' personality type and their effectiveness in teaching: Investigating the relationship', *International Journal of Educational Investigations* **3**(1), 166–177.

Fischer, K., Kirstein, F., Jensen, L. C., Krüger, N., Kukliński, K., aus der Wieschen, M. V. and Savarimuthu, T. R. (2016), A comparison of types of robot control for programming by demonstration, *in* '2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)', IEEE, pp. 213–220.

FSIQ (2019), 'Open source psychometrics project Full Scale IQ Test', https://openpsychometrics.org/tests/FSIQ/. Accessed: 2021-05-28.

Fu, J., Luo, K. and Levine, S. (2018), 'Learning robust rewards with adversarial inverse reinforcement learning', *ArXiv* **abs/1710.11248**.

Garrett, C. R., Chitnis, R., Holladay, R., Kim, B., Silver, T., Kaelbling, L. P. and Lozano-Pérez, T. (2021), 'Integrated task and motion planning', *Annual review of control, robotics, and*

autonomous systems **4**, 265–293.

Goldberg, L. R. (1992), 'The development of markers for the big-five factor structure.', *Psychological assessment* **4**(1), 26.

Gopalan, N., desJardins, M., Littman, M., MacGlashan, J., Squire, S., Tellex, S., Winder, J. and Wong, L. (2017), Planning with abstract markov decision processes, *in* 'Proceedings of the International Conference on Automated Planning and Scheduling', Vol. 27.

Gopalan, N., Rosen, E., Konidaris, G. and Tellex, S. (2020), Simultaneously learning transferable symbols and language groundings from perceptual data for instruction following, *in* 'Proceedings of Robotics: Science and Systems.'.

Gottardi, A., Tortora, S., Tosello, E. and Menegatti, E. (2022), 'Shared control in robot teleoperation with improved potential fields', *IEEE Transactions on Human-Machine Systems* **52**(3), 410–422.

Haldar, S., Mathur, V., Yarats, D. and Pinto, L. (2022), 'Watch and match: Supercharging imitation with regularized optimal transport', *CoRL* .

Hart, S. G. and Staveland, L. E. (1988*a*), Development of nasa-tlx (task load index): Results of empirical and theoretical research, *in* 'Advances in psychology', Vol. 52, Elsevier, pp. 139–183.

Hart, S. and Staveland, L. (1988*b*), 'Development of nasa-tlx (task load index): Results of empirical and theoretical research', *Advances in psychology* **52**, 139–183.

Hauser, K. and Latombe, J.-C. (2010), 'Multi-modal motion planning in non-expansive spaces', *The International Journal of Robotics Research* **29**(7).

Hauser, K. and Ng-Thow-Hing, V. (2011), 'Randomized multi-modal motion planning for a humanoid robot manipulation task', *The International Journal of Robotics Research* **30**(6).

Ho, J. and Ermon, S. (2016), Generative adversarial imitation learning, *in* 'NIPS'.

Holland, R. W., Hendriks, M. and Aarts, H. (2005), 'Smells like clean spirit: Nonconscious effects of scent on cognition and behavior', *Psychological science* **16**(9), 689–693.

Hsu, T.-C., Chang, S.-C. and Hung, Y.-T. (2018), 'How to learn and how to teach computational thinking: Suggestions based on a review of the literature', *Computers & Education* **126**, 296–310.

Huang, W., Xia, F., Xiao, T., Chan, H., Liang, J., Florence, P. R., Zeng, A., Tompson, J., Mordatch, I., Chebotar, Y., Sermanet, P., Brown, N., Jackson, T., Luu, L., Levine, S., Hausman, K. and Ichter, B. (2022), Inner monologue: Embodied reasoning through planning with language models, *in* 'Conference on Robot Learning'.

Ijspeert, A. J., Nakanishi, J., Hoffmann, H., Pastor, P. and Schaal, S. (2013), 'Dynamical movement primitives: learning attractor models for motor behaviors', *Neural computation* **25**(2), 328–373.

John, O. P. and Srivastava, S. (1999), *The Big-Five trait taxonomy: History, measurement, and theoretical perspectives*, Vol. 2, University of California Berkeley.

Jong, N. K., Hester, T. and Stone, P. (2008), The utility of temporal abstraction in reinforcement learning, *in* 'Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 1', AAMAS '08, International Foundation for Autonomous Agents and Multiagent Systems, p. 299–306.

Kaelbling, L. P. and Lozano-Pérez, T. (2010), Hierarchical planning in the now, *in* 'Workshops at the Twenty-Fourth AAAI Conference on Artificial Intelligence'.

Kim, W., Lee, C. and Kim, H. J. (2018), Learning and generalization of dynamic movement primitives by hierarchical deep reinforcement learning from demonstration, *in* '2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)', pp. 3117–3123.

Klapp, S. T. (1977), 'Response programming, as assessed by reaction time, does not establish commands for particular muscles', *Journal of Motor Behavior* **9**(4), 301–312.

Knaust, M. and Koert, D. (2021), Guided robot skill learning: A user-study on learning probabilistic movement primitives with non-experts, *in* '2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)', IEEE, pp. 514–521.

Knox, W. B. and Stone, P. (2009), Interactively shaping agents via human reinforcement: the tamer framework, *in* 'K-CAP '09'.

Kolasinski, J., Hinson, E. L., Zand, A. P. D., Rizov, A., Emir, U. E. and Stagg, C. J. (2018), 'The dynamics of cortical gaba in human motor learning', *The Journal of Physiology* **597**, 271 – 282.

Konidaris, G. (2019), 'On the necessity of abstraction', *Current opinion in behavioral sciences* **29**, 1–7.

Konidaris, G. and Barto, A. (2009), 'Skill discovery in continuous reinforcement learning domains using skill chaining', *Advances in neural information processing systems* **22**, 1015–1023.

Konidaris, G., Kaelbling, L. P. and Lozano-Perez, T. (2014), Constructing symbolic representations for high-level planning, AAAI'14, AAAI Press, p. 1932–1940.

Konidaris, G., Kuindersma, S., Grupen, R. and Barto, A. (2012), 'Robot learning from demonstration by constructing skill trees', *The International Journal of Robotics Research* **31**(3), 360–375.

Lashley, K. S. (1951), *The problem of serial order in behavior*, Vol. 21, Bobbs-Merrill Oxford.

Lee, K., Ng, S.-F., Ng, E.-L. and Lim, Z.-Y. (2004), 'Working memory and literacy as predictors of performance on algebraic word problems', *Journal of Experimental Child Psychology* **89**(2), 140–158.

Levine, S., Finn, C., Darrell, T. and Abbeel, P. (2016), 'End-to-end training of deep visuomotor policies', *The Journal of Machine Learning Research* **17**(1), 1334–1373.

Li, L., Walsh, T. J. and Littman, M. L. (2006), 'Towards a unified theory of state abstraction for mdps.', *AI&M* **1**(2), 3.

Lioutikov, R., Neumann, G., Maeda, G. and Peters, J. (2015), Probabilistic segmentation applied to an assembly task, *in* '2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)', IEEE, pp. 533–540.

Lövdén, M., Garzón, B. and Lindenberger, U. (2020), 'Human skill learning: expansion, exploration, selection, and refinement', *Current Opinion in Behavioral Sciences* **36**, 163–168.

Lozano-Perez, T. and Kaelbling, L. P. (2014), 'A constraint-based method for solving sequential manipulation planning problems', *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* pp. 3684–3691.

Lu, J. J. and Fletcher, G. H. (2009), Thinking about computational thinking, *in* 'Proceedings of the 40th ACM technical symposium on Computer science education', pp. 260–264.

MacGlashan, J., Ho, M. K., Loftin, R., Peng, B., Wang, G., Roberts, D. L., Taylor, M. E. and Littman, M. (2017), Interactive learning from policy-dependent human feedback, *in* 'ICML'.