Simultaneous Human-Robot Adaptation for Effective Skill Transfer*

Mohammad Ali Zamani
Computer Science Department
Ozyegin University
Istanbul, Turkey
ali.zamani@ozu.edu.tr

Erhan Oztop
Computer Science Department
Ozyegin University
Istanbul, Turkey
erhan.oztop@ozyegin.edu.tr

***Abstract*— In this paper, we propose and implement a human-in-the loop robot skill synthesis framework that involves simultaneous adaptation of the human and the robot. In this framework, the human demonstrator learns to control the robot in real-time to make it perform a given task. At the same time, the robot learns from the human guided control creating a non-trivial coupled dynamical system. The research question we address is how this system can be tuned to facilitate faster skill transfer or improve the performance level of the transferred skill. In the current paper we report our initial work for the latter. At the beginning of the skill transfer session, the human demonstrator controls the robot exclusively as in teleoperation. As the task performance improves the robot takes increasingly more share in control, eventually reaching full autonomy. The proposed framework is implemented and shown to work on a physical cart-pole setup. To assess whether simultaneous learning has advantage over the standard sequential learning (where the robot learns from the human observation but does not interfere with the control) experiments with two groups of subjects were performed. The results indicate that the final autonomous controller obtained via simultaneous learning has a higher performance measured as the average deviation from the upright posture of the pole.**

Keywords—Human-Robot Interaction; Skill Transfer; Human-in-the-loop

I. INTRODUCTION

It is generally expected that robots and autonomous agents will become a part of our daily lives in the coming decades. Yet, it is not possible to move robots directly from factories to our homes [1-2]. One obvious reason is safety [3]; the second reason is that the robots we require in our daily lives will be in charge of a variety of multiple tasks. It is not feasible to program robots in advance for all possible tasks using classical robot programming [2] as they are aimed at making robots specialized to predefined tasks [4]. Moreover, in a naturalistic setting, robots must be able to automatically self-calibrate, and adapt their –even fixed- behaviors according to the stochastic and dynamic environments they live in [2-3],[5]. Another important characteristic that sets apart today's industrial robots from the future daily-life robots is the definite need for the ability to interact with humans in the environments designed for humans [4],[6]. Therefore, intuitive and easy robot programming is one of the active research areas in robotics.

There has been considerable effort in the recent decades to make robot programming intuitive and easy. The two mainstream methods are: robot learning (self-improvement) [2-3] and teaching by demonstration (or so called imitation learning) approaches [6-9]. One of the self-improvement approaches is naturally based on reinforcement learning (RL) [10]. In RL the robot is taken as an agent that explores the state space (which is often the vector of joint angle and angular velocities) to maximize a predefined total (future discounted) reward function such as minimum total torque change to reach a given robot configuration. Straightforward RL is effective for problems degree of freedom but slow for higher number of degree problems (e.g. >6). With careful selection of hierarchical organization significant improvements can be achieved [11]. Furthermore, there are recent developments that hold promise for fast approximate RL [12-14]. In learning by demonstration, the skill of a demonstrator is transferred to the robot [15]. These two approaches are not exclusive and can be used together: the demonstration can be used as an initial rough solution after which RL improves the solution [16].

In this paper we focus on relying on the transfer of the skill to the robot without additional self-improvement. So, in this setting the simplest form of skill transfer is to record the demonstrated motion e.g. by motion capture techniques, and play back on the robot to achieve autonomous task execution [8]. Certain hand-crafted transformations, and/or inverse kinematics may be needed for this to work as the kinematics of the demonstrator and the robot may be different. An effective way to teach robots by human guidance is so called Direct Teaching [17-19], where the human demonstrator physically moves the robot's joints to accomplish the given task [20-21]. Direct Teaching is very intuitive; but unfortunately, it is not suitable for complex tasks that include non-negligible dynamics [20]. The reason for this is the fact the human is not really placed in the control loop of the robot. In contrast, in the Human Learning for Robot Skill Synthesis paradigm of Babic et al. and Oztop et al. [22-23] the human operator is placed in the real-time control of the robot as in teleoperation with the ultimate goal of obtaining an autonomous controller based on the performance of the operator. In this framework, the human operator/demonstrator must first learn how to do a given task by 'using' the robot as a 'tool' and become skilled at it. This is akin to a beginner's learning to drive a car. After the human becomes an expert at executing the task through the robot, the robot states and corresponding motor commands generated by the human operator are utilized to obtain a policy that drives the robot autonomously [22-24]. This paradigm has been

*Research supported by European Community's Seventh Framework Programme FP7/2007-2013 under the grant agreement no. 321700, Converge.

successfully applied to obtain skills such as ball manipulation [20] with an anthropomorphic robot hand, and balanced inverse kinematics for a humanoid robot [22], and more recently tasks that involve force based policies [21],[25]. It will be fair to say that robotic researchers are becoming increasingly more interested in human-in-the-loop robot control and learning, which provides a platform for both human and robot to learn actively [25-28].

We see two major variants of the human-in-the-loop robot teaching paradigm: In the first variant (sequential learning), the robot is considered as a stationary tool [20],[22] which does not change during the course of human control. This is the type of tasks we experience most in our daily lives, e.g. one's favorite computer mouse behaves the same every time it is used. In the second variant (simultaneous learning), the robot 'learns' together with the human, and this learning is incorporated into the control. So for the human the task is not stationary anymore. The latter begs to question whether this is really a good idea. This approach was somewhat used in an ad-hoc way in Ref. [21],[25] however no study addressed this question directly. In this paper, we make a contribution in this direction and show the acquisition of autonomous 'swing up and pole balance' skill via simultaneous learning. In addition, we present our initial work on comparing sequential and simultaneous learning.

In the proposed simultaneous learning framework we introduce the novel ideas of (1) state based control sharing, and (2) well defined dynamics for the mixing coefficients based on overall performance, and implement these on a physical cart-pole system. The human-in-the loop learning setup employed is similar to that of Ref. [25] where also simultaneous human-robot learning was implemented for balance control. There, however the control was weighted between human and robot simply based on the prediction error of the machine learning[1] algorithm that learns to replicate human actions. In general, weight sharing based on prediction error is not desirable as this may cause the transfer of a 'bad skill' when the demonstrator is consistently performing badly (e.g. doing nothing). This was not an issue in Ref. [25] because the feedback relayed to the demonstrator forced the subject to perform well (otherwise it would fall). Although the work reported in Ref. [25] is in the same spirit of our simultaneous learning, we introduce the novel improvements of 1, 2 mentioned above.

In summary, we define a dynamical system for a weight variable that determines the amount of mixing of robot and human controls. The dynamics gradually change the mixing weight based on the overall success during the progress of the task, which does not directly depend on prediction error of the machine learning module. The dynamics starts off with fully manual (operator control) and tends toward fully autonomous control as overall success level increases. As such, for example, at some point, the system may shift the control towards robot autonomy; but if this results in a reduced success level, then the control will be shifted back towards manual control. Eventually this tug-of-war will enable the robot to gain full control with continued high success in task execution. We hold that the weight dynamics must be state dependent, as some parts of the state space can be learned faster than the others. In the current implementation, we split the state space in discrete regions, with individual weight sharing dynamics for each region. This split can also be viewed as dividing the overall task into manageable sub-tasks. For experimental evaluation, the developed simultaneous human-in-the-loop learning framework was implemented on a cart-pole system for the task of 'swing-up and balance'. The effectiveness of the framework was validated by an expert subject. Good autonomous policies could be generated in very short times (10-30 minutes) compared to sequential learning (see http://youtu.be/S3NW0hr72mU). To further assess the effectiveness of the simultaneous learning with naïve subjects we made a preliminary experiment with 8 subjects. The subjects were randomly divided into two, and assigned to one of the learning setups: the first group engaged in simultaneous learning setup (the robot learned with the subjects and engaged in control, so the robot behavior was not stationary); the second group engaged in standard sequential learning setup (robot behavior was stationary, i.e. the learning did not affect robot behavior). The results indicate that the performance of the final autonomous controller of the simultaneous learning group on average is higher than that of the sequential group.

[1] Machine learning here refers to any proper learning method can learn the task; However, current framework using supervised learning methods.

II. METHOD

A. Human-Robot Simultaneous Learning System

We propose a framework that aims to engage human and robot learning simultaneously to improve the effectiveness of human to robot skill transfer. This may be seen analogous to the learning process between a human teacher and a human learner [25]. The learner tries to mimic the teacher's action gradually by getting help from the teacher when needed. As the learner becomes better at satisfying the task's objectives, the teacher becomes less involved. However, when this deteriorates performance the teacher intervenes back and issues corrective actions. In this framework, likewise, the demonstrator has the full control of the task in the very beginning, and gradually the control is shifted to the robot based on the overall performance, so if the performance degrades the human demonstrator's share in control increases. When the success objectives are completely satisfied, the robot becomes the only controller. At this point, the learned policy can be preserved as it is and deployed later for full autonomous execution of the task without human guidance. Fig.1 presents our human-in-the-loop simultaneous learning framework. Human demonstrator controls the robot in real-time via fixed feed-forward interface, and observes the robot state through a feedback interface (which could be simply direct vision). At the same time, the robot starts to mimic the human policy and injects its own control gated through a weight parameter. The final action (u) is simply the linear combination of the human and machine action (u_h, u_m respectively):

$$u = wu_h + (1 - w)u_m \quad (1)$$

Where $0 \leq w \leq 1$ indicates weight or share of the human action on the overall control. The key design issue is to define a w dynamics to facilitate effective skill transfer avoiding oscillations between human and machine control. For this we propose to use a time-windowed success indicator to determine

the dynamics of w. The proposed framework does not specify the details of the involved feedback or feed-forward interfaces. In the current implementation direct vision is used as the feedback. For the feed-forward interface we used a standard computer mouse: the horizontal displacements were multiplied with an experimentally tuned constant gain to obtain the voltage that drives the cart of the cart-pole system.

B. Cart-Pole swing up and balance task

We chose a cart-pole system to test the proposed simultaneous human-in-the loop learning framework (see Fig 2) . This task is simple for RL to solve. We used it to check two possible interfaces (i.e. Simultaneous vs Sequential). The desired controller would move the cart left and right several times to accumulate sufficient energy to swing up the pole and then keep it at the vertical upright position. So, the desired state manifold is given by $\theta = \dot{\theta} = 0$. The control policy is captured by a time independent function of the cart-pole state $u = g(X)$. The goal of the human-in-the-loop robot learning is to have the human produce control data points that can be learned by a machine learning algorithm so that g is obtained and can be used as an autonomous controller for the task. Here u is the controller output (voltage). X is the cart-pole state which is normally defined as $X = [x, \dot{x}, \theta, \dot{\theta}]$ where x is the position of the cart, $\dot{x}$ is the velocity of the cart, θ is the angle of the pole, $\dot{\theta}$ is the angular velocity of the pole. Alternatively the state definition can be chosen $X = [x, \dot{x}, \cos(\theta), \sin(\theta), \dot{\theta}]$ (The latter one is applied for this paper) which avoids the discontinuities due the periodic nature of θ, and so make the task of machine learning module easier.

C. Machine Learning

An important notion of human-in-the-loop skill synthesis is that 'data' is the key not the underlying machine learning algorithm, as the framework allows generation of large data sets as the human is placed in the control loop [23-24]. So we do not focus on the specifics of the machine learning algorithm to represent *g()*, but instead underline the requirements for a generic machine learning algorithm for simultaneous learning: (1) support for online incremental learning, (2) robustness against over-fitting, and (3) low computational load. A good match for these requirements is the Receptive Field Weighted Regression (RFWR) algorithm [29-30]. .

D. Weight Dynamics

One of the critical part of the learning is how fast to pass the control from the demonstrator to the robot. If the robot takes the control too early it may not learn the task properly, and it may also hinder the corrective actions that the demonstrator may take. Obviously, if the non-stationarity introduced is too severe (e.g. by a fast change of the weight), it may prohibitively increase the task complexity for the human demonstrator. On the other hand, slow transition of the control can be frustrating for the demonstrator. The aforementioned factor can be reflected as a time constant which presents how fast the machine can take the control during the demonstration.

Another factor is the acceptable level of task performance that should indicate a shift towards autonomy (e.g. shall we

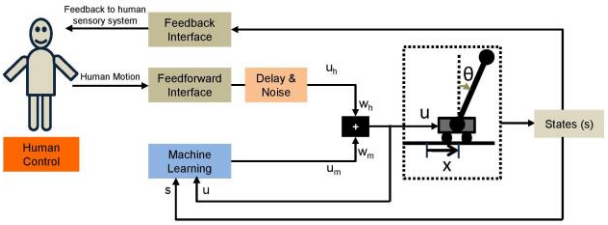

Figure 1. Simultaneous human-in-the-loop robot learning framework is illustrated. Human demonstrator controls the robot in real-time to achieve a desired goal through the robot . Simultanously, the robot learns to imitate the human policy. The net control the robot receives is obtained by the weighted summation of the human and machine generated controls. The weighting is dynamically adjusted so as to pass the control to the robot when successful learning and task completion is achieved. At this point the task skill has been transferred to the robot and it can complete the task autonomously

expect the demonstrator to find the theoretical optimal policy?). So it is reasonable to introduce a success measure that (time-wise) locally captures the performance of the human-robot system, and use it to guide the weight dynamics. As the weight parameter $0 \leq w \leq 1$ determines the level of autonomy, we arbitrarily define that $w = 1$ to mean full human (manual) control, and $w = 0$ to mean full robot (autonomous) control. One simple alternative can be linear decay of w which may not be good because the performance of subject does not decrease monotonically. As it is motivated above, we wish to have the control passed to the robot when the task performance is high. This intuition can be captured with the following w dynamics:

$$\tau \dot{w}(t) = -w(t) + f\big(success(t)\big), \quad w(0) = 1 \qquad (3)$$

Here τ is the time constant, $0 \leq success \leq 1$ is the measure for the temporally local task performance, and f is fixed monotonic function of $success$ bounded by 0 and 1. If $f = 0$ w will settle to zero, meaning that the control has been completely passed to robot. So, intuitively $f(success)$ should indicate the need for human guidance. There are infinitely many possible f choices; in the current implementation, we selected the simplest without further investigating the effect of this choice.

$$f(success) = 1 - success \qquad (4)$$

With this the dynamics of w becomes simply

$$\tau \dot{w}(t) = -w(t) + 1 - success(t) \qquad (5)$$

E. Success Measure and State Space Partitioning

In the cart-pole system an intuitive success measure is the (average) height of the tip of the pole. The goal in our cart-pole task is to bring the pole from downward hanging posture to the upright vertical posture and keep it there as long as possible, where the height achieves its maximum. Representing the angle of the pole from the vertical upright posture with θ we can take $cos\theta$ as the indicator for the height of the pole tip as a value in [-1, 1] irrespective of the pole length. From now on, we will use the term pseudo-height to refer this value.

It is reasonable to think that the success should be defined based on state or state regions. For example, one may be good at driving at low speeds; but, may be terrible at high speeds. This naturally brings the idea that a task can be divided into subtasks via partitioning of its state space. This is in fact a

common practice in (hierarchical) reinforcement learning [11]. It is not uncommon to see swing-up and pole balancing as different benchmarking tasks in the literature. By generalizing from this, we defined three regions in the pole angle space: *bottom* region (start), *middle* region (swing-up) and *top* region (balance). And, similarly we defined two regions for angular velocity: *slow* and *fast*. With this, we partitioned the state space into 2x3=6 regions, for which we defined individual success measures as:

$$success_i(t) = h\left(\frac{\overline{cos\theta_i(t)}\,{}^{t}_{t-\alpha} - cos\theta_i^{min}}{cos\theta_i^{threshold} - cos\theta_i^{min}}\right) \quad (6)$$

where h is given by

$$h(x) = \begin{cases} 0 & if\ x < 0 \\ x & if\ 0 \le x \le 1 \\ 1 & if\ x > 1 \end{cases}$$

and $cos\theta_i^{min}$ indicates the minimum pseudo-height of the pole tip in region i, $cos\theta_i^{threshold}$ is a constant threshold set for region i, and finally $\overline{cos\theta_i(t)}\,{}^{t}_{t-\alpha}$ is the moving average of the pseduo-height within the past α time unit. For convenience the state regions are named as down-slow, down-fast, middle-slow, middle -fast, top-slow and top-fast. Since we partitioned the state space based on the height and angular velocity of the pole, each state region is determined by four parameters of $cos\theta_i^{min}$, $cos\theta_i^{max}$, $|\dot{\theta}|_i^{min}$, $|\dot{\theta}|_i^{max}$. The parameter $cos\theta_i^{threshold}$ determines the performance level required for that region to be considered succesful. We selected the threshold relatively lower for the swing up task and higher for the balance task. These settings helps the demonstrator to clear the swing-up task faster and have it handled by the robot for focusing more on the balance task. The aforementioned parameters are given in Table I.

TABLE I. THE CART POLE STATE REGIONS PARAMETERS

	Average Achieved Height by Subjects (cos θ)				
	$cos\theta_i^{min}$	$cos\theta_i^{max}$	$\|\dot{\theta}\|_i^{min}$	$\|\dot{\theta}\|_i^{max}$	$cos\theta_i^{thresh}$
Down Slow	$cos(180^\circ)$	$cos(120^\circ)$	*0 rad/s*	*8 rad/s*	$cos(160^\circ)$
Down Fast	$cos(180^\circ)$	$cos(120^\circ)$	*8 rad/s*	*20 rad/s*	$cos(160^\circ)$
Middle Slow	$cos(120^\circ)$	$cos(30^\circ)$	*0 rad/s*	*3 rad/s*	$cos(70^\circ)$
Middle Fast	$cos(120^\circ)$	$cos(40^\circ)$	*3 rad/s*	*20 rad/s*	$cos(60^\circ)$
Top Slow	cos(30°)	$cos(0^\circ)$	*0 rad/s*	*1 rad/s*	$cos(4^\circ)$
Top Fast	cos(40°)	$cos(0^\circ)$	*1 rad/s*	*20 rad/s*	$cos(20^\circ)$

Since the success is defined to be state dependent, the control sharing also becomes state dependent. Hence the dynamical equations governing the weights for each region is given by:

$$\tau \dot{w}_i(t) = -w_i(t) + 1 - success_i(t),\ \ w_i(0) = 1 \quad (7)$$

F. Obtaining the Autonomous Controller

Since each state region has independent learning dynamics, the (sub)task completion (i.e. *w*=0 and *success*=1 for the particular region) may be attained at different times for each region. The overall skill is already transferred to the robot, when the weights for each region approximately reach 0, as at this time the human contribution to control would be minimal. At this point the parameters of the machine learning system is retrieved and saved as the autonomous policy that can be deployed at a later time. There is a practical issue that must be handled to make the life of the demonstrator easier: if the demonstrator fails to continue to provide appropriate action (e.g. due to tiredness or lack of attention) for the state regions that has been taught to the robot, *success* will decrease and the control weight will be shifted back to the human. In other words, thinking that a state region is learned by the robot and thus can be conveniently taken care of the robot will not really work in the standard form. However, it is easy to gain this convenience by *freezing* the learning for regions that attain high level autonomy during the overall training period. In the experiments reported this strategy was adopted.

Figure 2. The experimental setup with cart-pole system is shown

III. EXPERIMENTS AND RESULTS

The proposed simultaneous human-robot learning framework for robot skill transfer was evaluated on a physical cart-pole system. The experimental sessions started with the pole positioned downward with zero velocity and the cart placed at the center of the cart-track. The subjects were asked to swing up the pole and keep it there in balance. There were no instructions for the position of the cart. The system could be operated in manual (*sequential mode*) where the sole control is given by the human, and data collection was performed for subsequent machine learning; or in *simultaneous mode*, where the proposed learning and control sharing was employed. The open parameters for the proposed framework are the time constant weight dynamics (τ) and the size of the moving average window (a) that is using computing the task success (*success* parameter). Both parameters were chosen as 40 for the experiments reported in this paper.

The initial experiments were done with a subject who had extensive experience with the experimental setup, and thus could do *swing-up* quite easily but could not keep the pole in upright position for more than a few second via manual control. When the expert subject was asked to perform the task in simultaneous mode he was able to perform better in pole balancing[2]. In fact, the autonomous controller obtained as the result of 30 minutes of simultaneous learning could swing up and hold the pole in the upright position. During the experiment, the expert first completed the regions which are likely to be involved in the swing-up (down-slow, down-fast and mid-slow) which were easy to complete. Hence the weights of these regions converge to fully automatic (w~0) much faster (see Fig. 3, 4 dashed lines). On the other hand the state regions that were more involved in pole balancing (top-slow, top-fast, and mid-fast regions) took longer to reach w~0 level (see Fig. 3, 4 solid lines). A set of additional experiments

[2] The video of a learning session and autonomous performance can be viewed at http://youtu.be/S3NW0hr72mU

with 8 naïve subjects were carried out to assess the validity of the proposed method for naïve subjects, and compare it against the sequential learning scheme. The subjects were randomly divided into two groups of 4. All subjects were allowed to work with cart-pole system for 3 minutes to familiarize themselves with the human-in-the loop cart-pole control setup. Then, after a short break, they were given the robot control task for 7 minutes. The reason for this choice is that we expected to see initial learning effect with 7 minutes experiments. First group learned simultaneously with the robot, and the second group performed the task based on the sequential learning scheme, where robot did not change its behavior during human learning. Average pseudo-height during the first thirty seconds is used to assess the performance of the autonomous policies

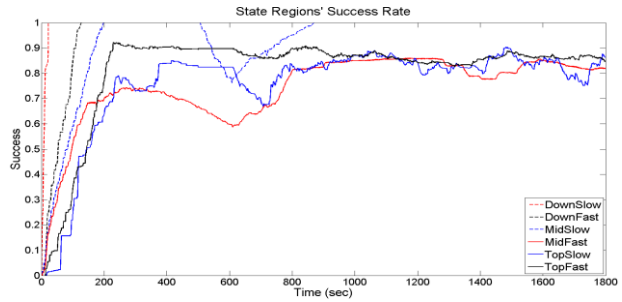

Figure 3. Success level of a skilled demonstrator in each state regions. 100% Success level reached in early time of easy-to-learn state regions. In the rest of the graph,the level oscillates between 70% and 100% for Middle Fast, Top Slow, and Top Fast state regions. (see http://youtu.be/S3NW0hr72mU)

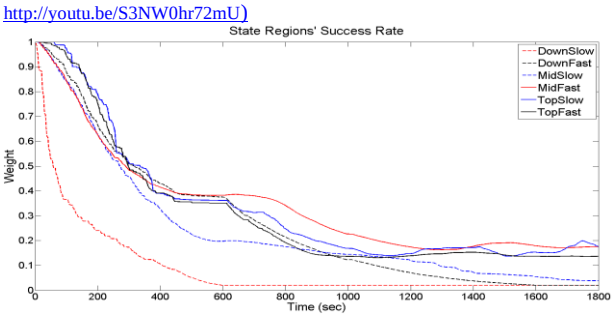

Figure 4. Control sharing weight in each state region for a skilled demonstrator is shown during simulatnaous learning. Weight starts from 1 (fully manual) and approaches to 0 (autonumous), gradually transferring the skill to the robot. This indicates different state regions have different difficulties.

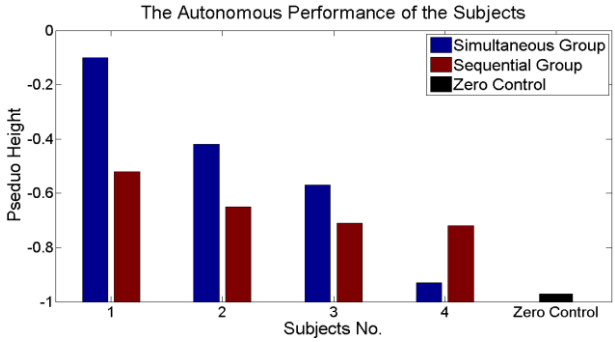

Figure 5. Average pseudo-height obtained from the autonmous policies obtained from the subjects in simultaneous (red) and sequential (blue) experimental conditions. The pole was initialized to downward vertical and a small perturbation was applied. The horizontal dashed line indicates the zero control (i.e. the voltage to the cart pole system was 0 V) performance of the perturbation.

obtained via the sequential and simultaneous learning[3]. In addition, the performance during the human control of the robot is also calculated. This corresponded to simple teleoperation in sequential mode and 'combined control' (human and machine) in the simultaneous mode. For testing the autonomous performance the pole was initialized to the downward vertical position and given a small fixed perturbation (this was necessary to get some 'poor[4]' policies started, otherwise we would not be able to make within 'poor policy' comparisons). The simultaneous learning based autonomous policies performed better in average (-0.50) than the sequential mode based policies (-0.65) (see Table-II). However the difference was not statistically significant as can be understood from the standard deviation given in the table. To gain an understanding at the individual level, we sorted the subjects in each experimental group according to their performance and generated the performance vs. subject order plots (Fig. 5). The worst subject of simultaneous group generated a policy worse than all sequential subjects[5]. Except this, simultaneous learning seems to generate better autonomous policies quantified as average pseudo-height as can be seen from Fig 5. An interesting observation is that the difference between the best and worst sequential policies is so small, which can be seen from the standard deviation in Table-II. This is very different in the simultaneous learning based policies where the best subject performance is really good. The generated policy from him could do swing up, and also pole balancing to some level. The performance and performance variation seen in autonomous policies could be also seen during learning (see Table-II, Human Control vs. Human + Robot Control).

TABLE II. THE COMPARISON OF SEQUENTIAL AND SIMULATANOUS HUMAN-IN-THE-LOOP ROBOT SKILL TRASNFER

Expr. Cond.	Average pseudo-height (cos θ) over subjects			
	Sequential Learning Group		*Simultaneous Learning Group*	
	Human Control	*Autonomous Robot Control*	*Human+Robot Control*	*Autonomous Robot Control*
Mean	-0.42	-0.65	-0.37	-0.5
STD	0.1	0.09	0.23	0.35

IV. CONCLUSIONS

In this paper, we proposed a simultaneous human-robot learning system for robot skill transfer. It is based on human-in-the-loop robot learning framework, and thus relies on a human operator to actively learn to complete a task through a robotic device. However, we addressed specifically the less studied topic of simultaneous control and learning of human and the robot. The novelty we introduced over the existing methods are (1) state based control sharing, and (2) well defined dynamics of the mixing coefficient based on the

[3] This duration of 30 seconds was found to capture the full behavior of the autonomous controllers obtained; however it is not critical and could have been chosen longer.

[4] By poor policy we refer to those autonomous policies which could not start swing-up task.

[5] This poor performance might be an outlier as the zero-control performance level (Fig 5, dashed horizontal line) is very close to his performance.

overall performance, rather than the prediction error.

The proposed system was realized on a physical cart-pole system and shown to be effective in generating pole swing-up and balance controller. Additional experiments with naïve subjects gave encouraging results that simultaneous learning can be more beneficial when the control mixing and learning regime are designed suitably. Intuitively, simultaneous learning can be understood as obtaining help from the robot which reflects the demonstrator's own policy of sometime in the past. In other words, the robot control helps to 'correct' inappropriate actions in a given state based on the average of the correct actions that were issued in that state leading to good performance. So in this sense, we were expecting better autonomous policies from the simulations learning framework. Our results, although not fully conclusive are in this direction. A counter argument to this is the fact that the learning task for the human becomes more complex, as the controlled robot is not stationary in the simultaneous learning case. In fact when we asked the subjects to do both sequential and simultaneous trials (this is after the experiments reported in this paper were finished) they stated that simultaneous robot control was harder.

In the naïve subject experiments, both groups were given the same amount of time to work with the cart-pole setup. However, the subjects in the simultaneous group had to deal with a harder task. It is likely that most subjects did not have enough time to reach their best during the experiments. If the goal is to obtain better policies the experiments should be continued until the success level of the subjects plateau.

There are several lines of future work that emerges as promising. Firstly, the experiments should be conducted with more subjects, and the subjects must be given longer time to work. Secondly, the state dependent weight dynamics must be based on an automatic state partitioning, or the dependence should be kept continuous. Thirdly, the open parameters of the framework, i.e. time constant for the weight dynamics and temporal window size for local success computation must be automatically adapted for each subject. Lastly, the framework should be validated on other robots and tasks, and a range of feedback interfaces must be tested for relaying back the level of competence of the machine control to the human operator.

REFERENCES

[1] S. Schaal, "The new robotics—Towards human-centered machines," HFSP J. Frontiers Interdisciplinary Res. Life Sci., vol.1, no. 2, pp. 115–126, 2007.

[2] Learning Control in Robotics, S. Schaal and C. G. Atkeson, IEEE Robotics & Automation Magazine, 17, 20-29, 2010.

[3] J. Peters and S. Schaal, "Learning to control in operational space," Int. J. Robot. Res., vol. 27, no. 2, pp. 197–212, 2008.

[4] Peternel, L., & Babiˇc, J. "Learning of compliant human–robot interaction using full-body haptic interface. *Adv. Robotics*, vol.27, no.13, pp. 1003–1012, 2013.

[5] J. Kober, E. Oztop, and J. Peters, "Reinforcement Learning to adjust Robot Movements to New Situations," in Proceedings of Robotics: Science and Systems (R:SS), 2010.

[6] Billard, A., Calinon, S., Dillmann, R., & Schaal, S. (2008). Robot programming by demonstration. In B. Siciliano & O. Khatib (Eds.), Springer handbook of robotics (pp. 1371–1394). Berlin: Springer.

[7] Schaal, S. (1999). Is imitation learning the route to humanoid robots? Trends in Cognitive Sciences, 3(6), 233–242.

[8] Atkeson, C. G., Hale, J. G., Pollick, F., Riley, M., Kotosaka, S., Schaal, S., et al. (2000). Using humanoid robots to study human behavior. IEEE Intelligent Systems, 15(4), 46–56.

[9] Argall, B. D., Chernova, S., Veloso,M., & Browning, B. (2009). A survey of robot learning from demonstration. Robotics and Autonomous Systems, 57(5), 469–483

[10] Sutton, R., & Barto, A. (1998). Reinforcement learning. MIT Press

[11] Morimoto J, Doya K (2001) Acquisition of stand-up behavior by a real robot using hierarchical reinforcement learning. Robotics and Autonomous Systems 36: 37-51.

[12] E. Todorov. Efficient computation of optimal actions. Proceedings National Academy of Science USA, 106(28):11478–83, 2009b.

[13] E. Theodorou, J. Buchli, and S. Schaal. A generalized path integral approach to reinforcement learning. Journal of Machine Learning Research, 11(Nov):3137–3181, 2010.

[14] M. P. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data efficient approach to policy search," in Proc. 28th Int. Conf. Mach. Learn., Bellevue, WA, 2011, pp. 465–472.

[15] Schaal S, Ijspeert A, Billard A (2003) Computational approaches to motor learning by imitation. Philos Trans R Soc Lond B Biol Sci 358: 537-547

[16] Kober, J., Wilhelm, A., Oztop, E., & Peters, J. (2012). Reinforcement learning to adjust parametrized motor primitives to new situations. Autonomous Robots, 33(4), 361–379.

[17] Kushida, D., Nakamura, M., Goto, S., & Kyura, N. (2001). Human direct teaching of industrial articulated robot arms based on forcefree control. Artificial Life and Robotics, 5(1), 26–32.

[18] J. Saunders, C.L. Nehaniv, K. Dautenhahn, A. Alissandrakis, Self-imitation and environmental scaffolding for robot teaching, International Journal of Advanced Robotics Systems 4 (1) (2007) 109–124.

[19] J. Saunders, C.L. Nehaniv, K. Dautenhahn, Teaching robots by moulding behavior and scaffolding the environment, in: 1st Annual Conference on Human–Robot Interaction HRI2006, Salt Lake City, Utah, USA, March 2–4, 2006, pp. 142–150.

[20] Moore, B., & Oztop, E. (2012). Robotic grasping and manipulation through human visuomotor learning. *Robotics and Autonomous Systems*, *60*(3), 441–451.

[21] Peternel, L., Petric, T., Oztop, E., Babic, J.. Teaching robots to cooperate with humans in dynamic manipulation tasks based on multi-modal human-in-the-loop approach. Auton. Robots, 2014, vol. 36, p. 123-136.

[22] Babic, J.,Hale, J. G.,&Oztop, E. (2011).Human sensorimotor learning for humanoid robot skill synthesis. *Adaptive Behavior—Animals, Animats, Software Agents, Robots, Adaptive Systems*, *19*, 250–263.

[23] Oztop, E., Lin, L.-H., Kawato, M., & Cheng, G. (2006). Dexterous skills transfer by extending human body schema to a robotic hand. *In 2006 6th IEEE-RAS Int. Conf. on humanoid robots (pp. 82–87).*

[24] Oztop, E.; Li-Heng Lin; Kawato, M.; Cheng, G., "Extensive Human Training for Robot Skill Synthesis: Validation on a Robotic Hand," *Robotics and Automation, 2007 IEEE International Conference on* , vol., no., pp.1788,1793, 10-14 April 2007.

[25] Peternel, L.; Babic, J., "Humanoid robot posture-control learning in real-time based on human sensorimotor learning ability," *Robotics and Automation (ICRA), 2013 IEEE International Conference on* , vol., no., pp.5329,5334, 6-10 May 2013

[26] Chipalkatty, R.; Egerstedt, M., "Human-in-the-Loop: Terminal constraint receding horizon control with human inputs," *Robotics and Automation (ICRA), 2010 IEEE International Conference on* , vol., no., pp.2712,2717, 3-7 May 2010

[27] Chipalkatty, R.; Daepp, H.; Egerstedt, M.; Book, W., "Human-in-the-loop: MPC for shared control of a quadruped rescue robot," *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ Int. Conf. on* , vol., no., pp.4556,4561, 25-30 Sept. 2011

[28] Bringes, C.; Yun Lin; Yu Sun; Alqasemi, R., "Determining the benefit of human input in human-in-the-loop robotic systems," *RO-MAN, 2013 IEEE* , vol., no., pp.210,215, 26-29 Aug. 2013

[29] Schaal, S.;Atkeson, C. G. (1998). Constructive incremental learning from only local information, *Neural Computation*, 10, 8, pp.2047-2084.

[30] Computational Learning and Motor Control Lab., software for *Receptive Field Weighted Regression (RFWR) [online]. http://www-clmc.usc.edu/Resources/Softwar*