2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)
Nancy, France. November 22-24, 2024

2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids) | 979-8-3503-7357-8/24/$31.00 ©2024 IEEE | DOI: 10.1109/Humanoids58906.2024.10769918

Curriculum Learning Influences the Emergence of Different Learning Trends

Romina Mir, Pegah Ojaghi, Andrew Erwin, Ali Marjaninejad, Michael Wehner, Francisco J Valero-Cuevas*

***Abstract*— Reinforcement learning (RL) algorithms are traditionally evaluated and compared by their learning trends (i.e., average performance) over trials and time. However, the presence of a single learning trend in a curriculum is, in fact, an assumption. To test this assumption, we used the performance of Proximal Policy Optimization (PPO) under five different curricula aimed at learning dynamic in-hand manipulation tasks. The curricula consisted of different combinations of rewards for lifting and rotating a 5g ball with a three-finger hand with the palm facing down. Mining the performance of all 60 individual trials as time series, we find there are learning trends distinct from the average. We conclude researchers should look beyond the average learning trends when evaluating curriculum learning to fully identify, appreciate, and evaluate the progression of autonomous learning of multi-objective tasks.**

I. INTRODUCTION

Traditionally, RL algorithms are evaluated by averaging multiple trials (i.e., trajectories). Each trial represents a sequence where the RL agent interacts with its environment, makes decisions based on its policy, receives feedback, and adjusts its actions to maximize some notion of cumulative reward [1]–[4]. The practice of employing averages and measures of dispersion as the default method to evaluate learning performance is so common that it is often employed without much deliberation [5]–[9]. This assumption has been naturally extended to the evaluation of curriculum learning in RL implementations, where training progresses from one subset of a task to another. However, this assumption may not always hold, necessitating a critical examination of the underlying assumptions and potential limitations associated with relying solely on averaged performance trends in multi-objective tasks.

RL is extensively utilized in the field of robotics [10], [11]. One of the primary goals in robotics, particularly crucial for interaction with and use of objects in unstructured human environments, is *dexterous in-hand manipulation*—i.e. dynamically holding and reorienting an object with the fingertips [12]–[16]. Hence, we will use in-hand manipulation as an experiment to detect different learning trends when learning multi-objective tasks.

Mir R, Marjaninejad A, Erwin A, and Valero-Cuevas F are with the Department of Biomedical Engineering, University of Southern California, Los Angeles, California, USA

Ojaghi P is with the Computer Science and Engineering Department, University of California Santa Cruz, Santa Cruz, California, USA

Wehner M is with the Mechanical Engineering Department, University of Wisconsin-Madison, Madison, Wisconsin, USA

* Corresponding Author: valero@usc.edu

An essential step before performing any detailed data analysis is understanding the characteristics of a given data set. Evaluating underlying learning trends in time series data from learning algorithms can be challenging. However, one can visualize RL learning trajectories as time series of over all episodes of a run. This involves plotting performance metrics (such as reward, loss, or success rate) against the number of episodes as time goes by.

In this study, we propose finding clusters of performance time series to reveal learning trends. As a task, we use the example of dynamical manipulation involving lifting and rotating a ball with a simulated 3-finger robotic hand using 3D-force tactile sensory information. We employ end-to-end Proximal Policy Optimization (PPO) algorithm as a benchmark state-of-the-art reinforcement learning algorithm in simulation.

Our contributions are as follows:

- We present a novel unsupervised clustering algorithm to detect learning trends from multiple RL runs.
- We investigate the impact of multi-objective curriculum learning trends in in-hand dynamic manipulation tasks.

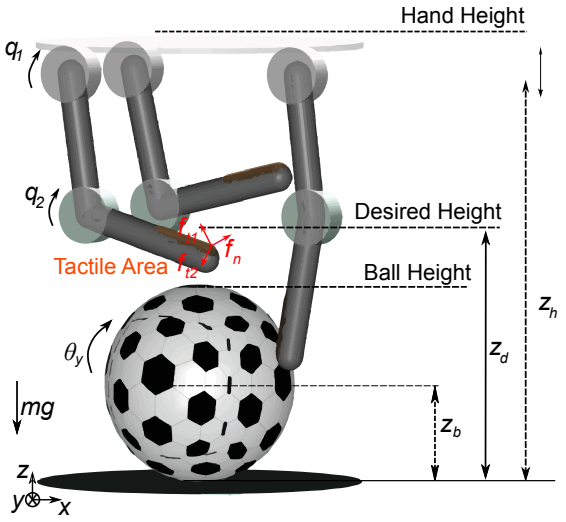

Fig. 1. **High-level overview of the simulation environment and learning approach to autonomous manipulation.** A simulated three-finger robotic hand attempted to lift and rotate (i.e., dexterously manipulate) a ball. The 3D movement of the ball was lightly constrained to the X-Z plane. Changes in the ball state affect the reward, which is a function of rotation, lift, and/or a combination of the two. We tested this approach with 3D tactile force sensing at the pad of each fingertip (colored in brown).

II. METHODOLOGY

In our previous study (Ojaghi, Mir et al. [17]), we collected simulation data on different objects to showcase the progression of average learning of different tasks and tactile sensory input. In this paper, we utilize a subset of the same dataset to explore the progression of learning in five different tasks.

A. System Overview

System Environment: As shown in Fig. 1, our simulation environment consists of a three-finger robotic hand and a 5-gram, 70 mm diameter ball. We demonstrate different learning trends using an autonomous learning algorithm with a combined reward to pick up and manipulate an object against gravity without vision. The state vector of the hand ($\mathbf{s_h}$) includes seven actuated kinematic DOFs and their derivatives: two rotational joints (q_1 and q_2) per finger plus the vertical position of the palm (z_h) with the maximum translation of 130 mm. The robotic hand reaches for and manipulates the state vector of the ball ($\mathbf{s_b}$) by lowering the palm and then actuating its rotational joints to rotate (θ_y) and lift (z_b) the ball to the desired height (z_d). Through simulation constraints, the ball's motion is restricted to be in-plane, i.e., it is free to move vertically (z) and horizontally (x) and rotate in the plane (θ_y).

Our learning algorithm incorporates tactile information from the pad of each finger, specifically utilizing the full contact force vector (*3D-force*), denoted as $\mathbf{f} = [f_{t1}, f_{t2}, f_n]$. This tactile data is integrated into the state vector for the hand ($\mathbf{s_h}$).

Learning: We used the end-to-end Proximal Policy Optimization (PPO) algorithm autonomous learning from OpenAI's stable-baselines2 repository with MultiLayer Perceptron (MLP) Artificial Neural Network (ANN) for the actor-critic model [18], [19]. At every time step t, the robotic hand observes the state of the hand $\mathbf{s}_{h,t}$ and the state of the ball $\mathbf{s}_{b,t}$, predicts the optimized action, executes it $\mathbf{a}_t$, and a reward is used r_t. The state $\mathbf{s}_{h,t}$ contains the angle and angular velocity $q_t, \dot{q}_t$ of each finger and the position and linear velocity of the palm at every time step t.

Learning Rate Strategy: Instead of utilizing a fixed or decreasing learning rate, our method embraces a piece-wise linear learning strategy, defined as follows:

$$Lr = \begin{cases} \phi \cdot \left(1 - \frac{\text{sample number}}{Th_1}\right), & \text{sample number} \leq Th_1 \\ \eta \cdot \left(1 - \frac{\text{sample number}}{Th_2}\right), & \text{sample number} > Th_1 \end{cases}$$

where Th_1 and Th_2 are first and second thresholds with values equal to 10^6 and 2×10^6, respectively. The optimal values for ϕ and η were determined empirically as 1 and 0.98 respectively. These coefficients are then integrated into the PPO linear scheduler according to the equation above. Our scheduler dynamically changes at 1,000 episodes (1,000,000 samples; Th_1), compelling the learning rate to be piecewise linear to accommodate the variations in the dynamics of the reward and tasks. A learning trial (an independent run) consisted of 2,000 episodes (Th_2), where each episode lasted 10 seconds.

Reward Function: In our algorithm, the goal is reached when the agent rotates the ball while keeping it against gravity between a height span of $[18.75, 31.25]$ mm, which is $\pm\%25$ of the desired height for the first half of learning (first half: episodes 1 to 1,000). In the second half of the learning (second half: episodes 1,001 to 2,000), the rotation reward has been switched off. The reward function is designed in a way that combines primary (positive) reward and punishment (penalty or negative reward) at every time step. Angular velocity of the ball $\dot{\theta}_y$ would be the primary reward, and the absolute distance of the state from the reference state of having the ball at the fixed desired position ($z_d = 25$ mm, (Fig. 1)) would be the punishment.

The reward function is described by:

$$Reward_t = (0.51)\dot{\theta}_{y,t} - (0.49)|z_{h,t} - z_d|,$$

.

We defined a curriculum through various permutations of two objectives (i.e. sub-task) of lift and rotation. We introduced five curricula featuring different combinations of these two objectives of lift **(L)**, rotation **(R)**, and lift & rotation **(L+R)**. Here the curriculum learning consists of training from one subset of a task to another subset of it [20], [21] or adding up different aspects of a more complicated sub-task at each step of the learning [22]. Note that in curricula where only lift is rewarded, the rotation coefficient is set to zero (c_R), and in curricula in which the lift coefficient is set to zero (c_L), only rotation is rewarded (c_R).

B. Clustering Algorithm

Dividing a dataset into subgroups is a popular clustering approach to data analysis [3]. Hierarchical clustering cannot represent distinct clusters with similar expression patterns. Also, as clusters grow in size, the actual expression patterns become less relevant [23]–[25]. K-means clustering is a

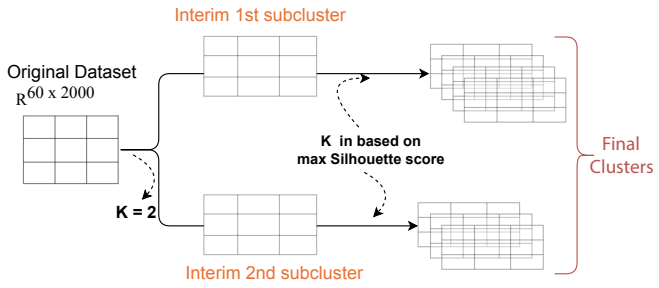

Fig. 2. **Two-Level Iterative K-Means Clustering** This diagram illustrates our iterative K-means clustering approach for analyzing RL learning trends across a comprehensive dataset comprising 60 trials containing three different levels of tactile conditions. In 'level 1', after applying Silhouette Score analysis (here we end up with $K = 2$) for interim clustering, it followed by a detailed exploration using K-means clustering from $K = 2$ to 10 in 'level 2'. The final number of clusters in each iteration was determined using peak Silhouette scores, highlighting distinct learning trajectories influenced by tactile feedback. Refer to Fig. 3 for the Silhouette score line plot.

widely used algorithm for partitioning data into clusters based on similarity. Yet despite its popularity, K-means has several limitations. One significant drawback is its sensitivity to the initial placement of centroids, which can lead to different clustering results. Moreover, K-means is not robust to outliers and noise in the data, and it provides hard assignments, meaning that each data point is assigned to exactly one cluster. In general, while K-means is efficient and straightforward, it is important to consider its limitations and explore alternative clustering methods when dealing with complex datasets [26], [27].

To overcome these limitations, we propose a two-level iterative clustering algorithm. Moving away from conventional approaches, our goal is to maximize the similarity of the time series data observations grouped while maximizing the dissimilarity of the observations clustered in distinct groups. In time series analysis, identifying trends is pivotal. A trend denotes a sustained movement over an extended period within a time series, [28].

Iterative K-means Clustering: As we used a data-driven learning algorithm, we aimed to find any learning patterns and similarities within and between all trials across each curriculum.

Algorithm 1 Two-Level Hierarchical Clustering with K-Means (Silhouette Score)

Require: Time series $X = \{x_1, x_2, ..., x_n\}$
Ensure: Hierarchical clusters $C_1, C_2, ..., C_n$
  $C_1 \leftarrow X$ {Initialize the top-level cluster with all data points}
  $level \leftarrow 1$
  while $|C_{level}| <= 2$ **do**
    $C_{level+1} \leftarrow \{\}$
    for $i = 1$ to 10 **do**
      Compute Silhouette scores for different values of K for $C_{level}[i]$
      Choose K that maximizes the Silhouette score
      Perform K-Means clustering on $C_{level}[i]$ with chosen K clusters, resulting in $C_{level+1}[i][1], C_{level+1}[i][2], ..., C_{level+1}[i][K]$
      $C_{level+1} \leftarrow C_{level+1} \cup \{C_{level+1}[i][1], C_{level+1}[i][2], ..., C_{level+1}[i][K]\}$
    end for
    $level \leftarrow level + 1$
  end while

We employed an iterative K-means approach inspired by Osseward et al. [29], and drawing from the K-means clustering algorithm for time series [30], we utilized it as the foundation for our clustering method.

The Silhouette score, a crucial metric in evaluating the quality of clustering results in unsupervised learning algorithms like K-means, measures how well a data point fits within its assigned cluster compared to other clusters [31]. Ranging from -1 to 1, the score quantifies cluster separation, with higher values indicating that a point is closely grouped with its cluster and well-separated from others, while negative values suggest potential misclassification, where a data point may be closer to a neighboring cluster than its own.

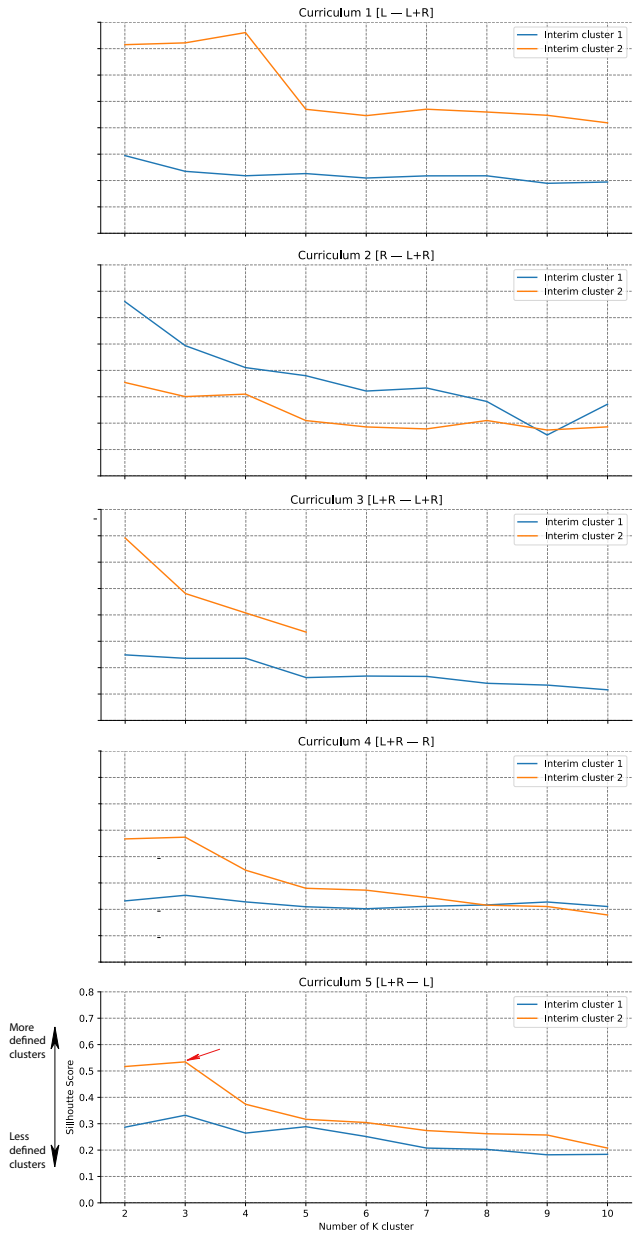

Fig. 3. **Silhouette line plot for each curriculum**. Silhouette scores for interim clusters after the first level of iterative K-means clustering for $K = 2$ to 10. Scores near $+1$ indicate a perfect fit between the trends and their corresponding cluster, while scores below 0.5 suggest a poor fit. Results from 60 runs under the 3D-force tactile condition reveal a Silhouette score below 0.5 for one interim cluster, indicating weak separation. However, through the second interim cluster, the peak suggests optimal K. As an example, for Curriculum 5 [L+R–L], one interim cluster is further subdivided, achieving an optimal Silhouette score at $K = 3$ (indicated by the red arrow), which suggests the data is best clustered into three groups, resulting in four clusters overall. Additionally, note that 'interim cluster 2' in Curriculum 3 [L+R–L+R] contains no more than 5 trials (data points), which limits the maximum number of clusters the K-means algorithm can create to 5, as it cannot exceed the number of data points.

For the first level of clustering, and its number of branches, which we refer to as *interim clusters*, is determined through

Silhouette score, we then further apply K-means clustering to each interim cluster. The peak Silhouette score then determines the final number of clusters in each interim clusters. This second sub-clustering was utilized to encourage separation based on a more modest heterogeneity in the initial branches.

As shown in Fig. 2, $K = 2$ shows the optimal number of clusters for the first level of clustering, and then apply K-means clustering to each branch from $K = 2$ to 10. The final number of clusters in each branch is then determined by the peak Silhouette score. This second sub-clustering was employed to encourage separation based on resulting more modest heterogeneity in the initial branches.

The code for clustering is available at GitHub repository.

Dataset: Employing a clustering algorithm, we group the in-hand manipulation data to extract distinct clusters. For this purpose, we conducted evaluations across 60 trials in all curricula defined in Tab. I. For ease of understanding, we look at the percentage of time the ball's height spent in the desired height range (lift success rate) in each episode throughout the trials. We ran our algorithm separately for each curriculum. We visualize the singular objective of lifting to facilitate comprehension of our clustering algorithm's outcomes.

As part of preprocessing, we used signal smoothing to remove noise from the data and reveal patterns. The noise originates from the ball's movement during the dynamic task and the hand's attempts to incorporate rotation based on the reward.

Task	First 1,000 episodes	Second 1,000 episodes
Curriculum 1	Lift	Lift and Rotate
Curriculum 2	Rotate	Lift and Rotate
Curriculum 3	Lift and Rotate	Lift and Rotate
Curriculum 4	Lift and Rotate	Rotate
Curriculum 5	Lift and Rotate	Lift

TABLE I

LEARNING CURRICULA

III. RESULTS AND DISCUSSION

A. Looking beyond average learning performance

We first look at the average progression of performance for all five curricula, Fig. 4. We find that different curricula produced different *average learning trends*, showing that the curriculum matters to this task. As an example, for the average learning performance in Curriculum 2 [R–L+R], we observe no lift at the outset (it was *not* rewarded) or, surprisingly, throughout the second half of the learning trajectory (when it was rewarded). Consequently, there is minimal *average learning of lift* for this curriculum. In contrast, looking at Curriculum 4 [L+R–R], where we reward through the learning trajectory and reward lift in the first half, we see the average progression of successful lifting in the first half of learning (first 1,000 episodes).

Moving away from the average progression of learning, however, our clustering algorithm unveiled multiple *'learning trends,'* allows us to detect multiple learning trends within and across the five curricula shown in Fig. 5. For instance, three distinct learning trends are detected in Curriculum 3 [L+R–L+R] . This was found (in an unsupervised way) by the Silhouette score that peaked at around 0.68 for two clusters at the second level of sequential sub-clustering for one branch, and no reasonable Silhouette score for the other branch (Fig. 3). Applying this clustering analysis to all curricula uncovers multiple learning trends as listed below, which may occur in some but not other curricula. We named them as per their salient features, namely:

- *No learner*: Worse learner in Curriculum 2, 3, 4 where there was minimal learning and no learning in Curriculum 1 and 5.
- *Saturate high*: in Curriculum 1, 3 and 5: where there is an asymptote to a high final performance level
- *Saturate low*: Only in Curriculum 3, same as above but with a lower final performance
- *Steady learner*: In Curriculum 1 where there seems to be learning and improvement in performance over time.
- *Ramp and drop*: in Curriculum 1 where initial high performance shows a small drop when the curriculum is implemented
- *Learn with no drop*: in Curriculum 4: Similar to Ramp and from where initial high performance shows a small drop when the curriculum is implemented
- *Learn and drop high*: In Curriculum 4 similar to above but where initial high-performance follows a big dip in performance as the change in the goal
- *Learn and drop low*: In Curriculum 4 similar to above but where initial high-performance follows a smaller dip in performance as the change in the goal

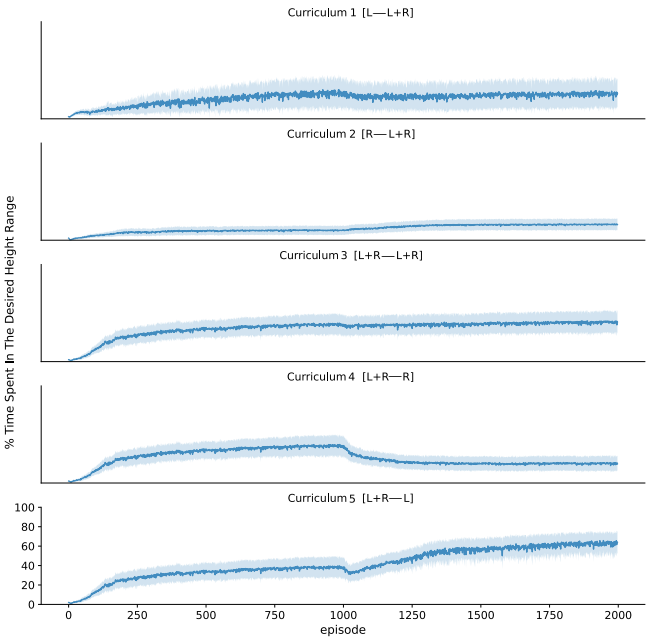

Fig. 4. **Average progression of learning in each curriculum.** We plot the mean lift success rate ± 1 standard deviation through 60 trials in each curriculum over a learning trajectory spanning 2,000 episodes.

- *Late learner*: Only in Curriculum 2 where there seems to be no learning or defined goal in the first 1,000 episodes of learning and it learns only in the second half of the curriculum
- *Modest learning and jump high*: In Curriculum 5 where initial performance is minimal and then we see a big jump as the curriculum is implemented
- *Steady learn and jump low*: In Curriculum 5 where initial performance is moderate and steady and then we see a a jump as the curriculum is implemented– yet better than the the previous learning trend (Modest learning and jump high)

Clustering across all curricula was conducted in an unsupervised manner, solely guided by Silhouette scores. The algorithm revealed insightful distinctions, retrospectively uncovering meaningful categorizations of trials based on their success or failure in learning. Most notably, the clustering effectively differentiated between 'learners' and 'non-learners.' As seen in Fig. 3, interim clusters that did not undergo further sub-clustering were consistently identified as *No-learner*. This underscores the value of sub-clustering, not only in distinguishing learners from non-learners but also in further exploring diverse learning trends within the 'learners' category.

Figure 5, shows all of these learning trends. In Curriculum 1 [L–L+R] there is a *No learner* trend, and the remaining three exhibit successful learning, with a lift success rate ranging from 50–70%. Two learning trends exhibit a change in the success behavior as we introduce the new subtask of rotation, while one learning trend does not continue its steady learning over the learning trajectory (*Steady learner*).

In Curriculum 2 [R–L+R], three discernible learning trends emerge corresponding with shifts in the reward structure. One trend depicts a failure to learn, while the other two showcase distinct patterns: one characterized as a *Steady learner* exhibiting rapid, consistent performance improvement even in the absence of lifting rewards during the first half of the learning trajectory. The second learning trend features *Late learner*, where lifting is absent in the first half, followed by a subsequent rise in learning.

In Curriculum 3 [L+R–L+R], the *No learner* shows some early learning, but could not continue the path of learning projection in the lift. In contrast, the remaining two learning trends, *Saturate high* and *Saturate low*, demonstrate consistent improvement in lifting performance.

In Curriculum 4 [L+R–R], alongside the persistent *No learner* trend, we observe three distinct learning trajectories while removing lifting goals in the latter half of the learning trajectory. The *Learn and no drop* trend maintains consistent lifting success, albeit with minimal change. Conversely, another trend, characterized by a shift away from lifting objectives, still achieves approximately 40% lift success,*Learn and drop low*. However, the *Learn and drop high* trend, which initially focused on lifting goals, shifts its emphasis during the latter half of the learning trajectory, leading to a dramatic decrease in lifting to the point that it can be considered a *No learner* at the end of the second half.

In the final case, Curriculum 5 [L+R–L], besides a (*No learner*), three distinct trends demonstrate varying degrees of success in learning to lift. Despite reaching a similar success rate of nearly 80% by the end of the learning trajectory, they display different learning patterns over time. This is quite interesting as one may think that such (few) runs would not merit continuation as they are not performing well at the midpoint, but they flourish towards the end—which may not have been necessarily predictable early on. The *Saturate high,* trend shows minimal change in learning despite changes in reward. The other two trends focus differently: *Modest learning and jump high* achieves a lower lift success rate of 20%, while *Steady learn and jump low* maintains a 40% success rate in lifting, at the first 1,000 episodes.

B. Why Does Curriculum Impact Learning Trends?

It came as no surprise to us that all curricula displayed a *No learner* trend, as it's not expected for the PPO algorithm to generate learning in 100% of the trials (minimal to no learning in all curricula, in Fig. 5). It was surprising, however, how the *No learner* trend was not equally common in each curriculum, objectively showing that some curricula are more and less successful than others, Fig. 6. In Curriculum 3 [L+R–L+R], where the reward remains constant, only three learning trends (i.e., the fewest number of trends, also seen in Curriculum 2 [R–L+R]) emerge, with a mere 10% dedicated to the *No learner* trend. Interestingly, the two successful trends closely follow each other, but are nevertheless distinct as per their Silhouette score. In the other curricula, we naturally expected that the change in reward, as per the goal of curriculum learning, would change performance (i.e. as per learning trends we encounter in Curriculum 2 [R–L+R]).

However, it was unexpected to see how the curriculum created so many different learning trends both within each half of learning, and after the transition to a new reward. It was interesting to observe that within these learning trends, some exhibited rapid rises while others progressed more gradually in the same half of the learning trajectory; some experienced minor declines, while others saw significant drops. Our future work will delve into examining the evolution of various components of the reward throughout the learning trajectory. Notably, in this study, we only presented the lift portion of the reward.

At this point, we can nevertheless propose that the effect of curriculum on the progression of learning could be attributed to the fact that learning is a dynamical process. If we consider a dynamical process to be nonlinear, then small changes in parameters, initially or during learning, may lead each run to fall into a specific time-varying minimum (i.e., a trend). Our results show that there are likely not infinite local minima (else each run would be different in principle). Rather, the landscape of performance has a structure with few (at most five) local minima that are then populated by the different curricula. However, exploring this interaction among task mechanics, reward, and progression of learning merits further study.

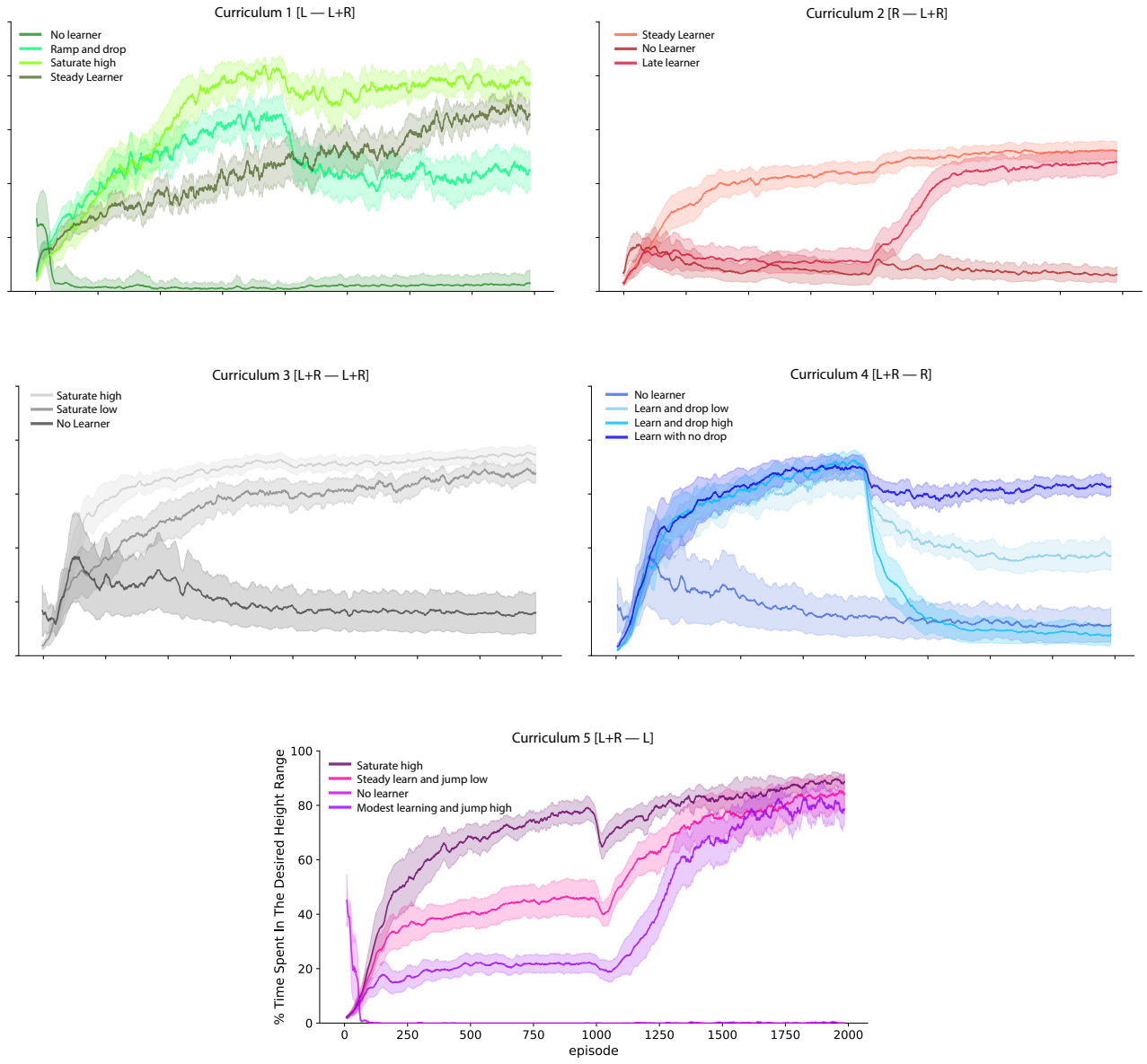

Fig. 5. **Learning trends over 60 trials for each 5 different curricula.** The solid line is the mean of the learner and the shades show the ± 1 standard deviation.

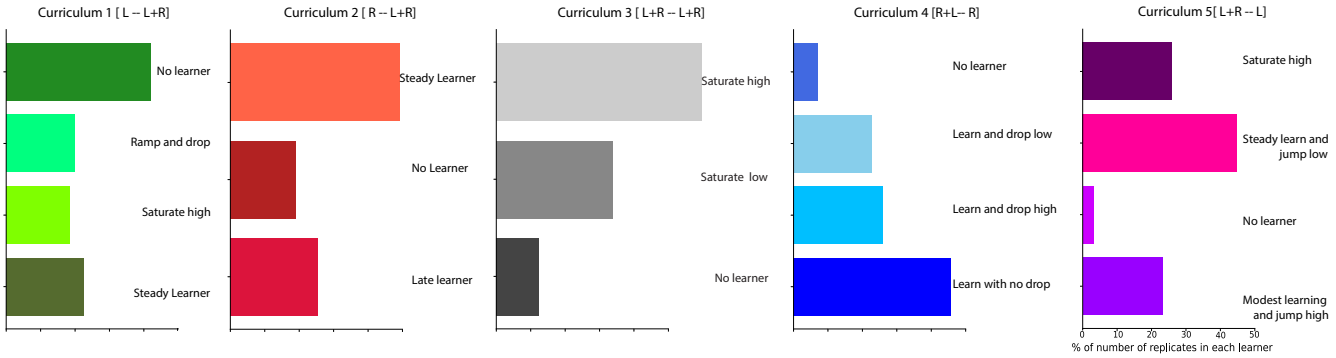

Fig. 6. **Distribution of trials across learning trends (based on Fig. 5).** The bar graph represents the percentage of trials in each learning trend over each curriculum (with 60 trials).

IV. CONCLUSION

Recognizing the existence of trends in RL is crucial, especially when comparing withing and across curricula. Our exploration revealed learning trends not visible in the average trend for each curriculum. Identifying learning trends allows one to and shedding light on the multi-objective dynamics of in-hand manipulation tasks. This was facilitated by employing a novel unsupervised clustering algorithm, which allowed us to uncover the impact of curriculum learning on the emergence of diverse learning trajectories and patterns. Our results place curriculum learning in the context of a dynamical—likely nonlinear—dynamical process that results from the interaction reward, task mechanics, and experience of the learning (i.e., details of each run).

ACKNOWLEDGMENT

This work is supported in part by the NIH R21 NS113613-01A1, NSF 2113096 CRCNS US-Japan, DOD CDMRP Grant MR150091, DARPA-L2M W911NF1820264 awarded to FV-C, and the USC Viterbi School of Engineering Fellowships to RM. This work does not necessarily represent the views of the NIH, NSF, DoD, or DARPA.

REFERENCES

[1] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.

[2] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, "Deep reinforcement learning that matters," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[3] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering–a decade review," *Information systems*, vol. 53, pp. 16–38, 2015.

[4] H. Azadjou, A. Marjaninejad, and F. Valero-Cuevas, "Play it by ear: A perceptual algorithm for autonomous melodious piano playing with a bio-inspired robotic hand," *bioRxiv*, pp. 2024–06, 2024.

[5] V. J. Santos, C. D. Bustamante, and F. J. Valero-Cuevas, "Improving the fitness of high-dimensional biomechanical models via data-driven stochastic exploration," *IEEE transactions on biomedical engineering*, vol. 56, no. 3, pp. 552–564, 2008.

[6] A. Dunne, M. Etropolski, A. Vermeulen, and P. Nandy, "On average: data exploration based on means can be misleading," *The AAPS journal*, vol. 14, no. 1, pp. 60–67, 2012.

[7] A. D. Mancini, "The impact of major life events and acute stress may not be what you think," 2013.

[8] S. L. Savage and H. M. Markowitz, *The flaw of averages: Why we underestimate risk in the face of uncertainty*. John Wiley & Sons, 2009.

[9] C. Speelman and M. McGann, "How mean is the mean?" *Frontiers in Psychology*, vol. 4, p. 53511, 2013.

[10] M. A.-M. Khan, M. R. J. Khan, A. Tooshil, N. Sikder, M. P. Mahmud, A. Z. Kouzani, and A.-A. Nahid, "A systematic review on reinforcement learning-based robotics within the last decade," *IEEE Access*, vol. 8, pp. 176 598–176 623, 2020.

[11] L. C. Garaffa, M. Basso, A. A. Konzen, and E. P. de Freitas, "Reinforcement learning for mobile robotics exploration: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 8, pp. 3796–3810, 2021.

[12] A. Billard and D. Kragic, "Trends and challenges in robot manipulation," *Science*, vol. 364, no. 6446, 2019.

[13] F. J. Valero-Cuevas and M. Santello, "On neuromechanical approaches for the study of biological and robotic grasp and manipulation," *Journal of neuroengineering and rehabilitation*, vol. 14, no. 1, pp. 1–20, 2017.

[14] R. Murray, Z. Li, and S. Sastry, "A mathematical introduction to robotic manipulation crc press," *Boca Raton, FL*, 1994.

[15] M. R. Cutkosky and R. D. Howe, "Human grasp choice and robotic grasp analysis," in *Dextrous Robot Hands*. Springer, 1990, pp. 5–31.

[16] V. Caggiano, G. Durandau, H. Wang, A. Chiappa, A. Mathis, P. Tano, N. Patel, A. Pouget, P. Schumacher, G. Martius *et al.*, "Myochallenge 2022: Learning contact-rich manipulation using a musculoskeletal hand," in *NeurIPS 2022 Competition Track*. PMLR, 2023, pp. 233–250.

[17] P. Ojaghi, R. Mir, A. Marjaninejad, A. Erwin, M. Wehner, and F. J. Valero-Cuevas, "Curriculum is more influential than haptic information during reinforcement learning of object manipulation against gravity," *arXiv preprint, arXiv:2407.09986*, 2024.

[18] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[19] J. T. Kristensen and P. Burelli, "Strategies for using proximal policy optimization in mobile puzzle games," in *International Conference on the Foundations of Digital Games*, 2020, pp. 1–10.

[20] X. Wang, Y. Chen, and W. Zhu, "A survey on curriculum learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[21] J. L. Elman, "Learning and development in neural networks: The importance of starting small," *Cognition*, vol. 48, no. 1, pp. 71–99, 1993.

[22] A. Marjaninejad, "Autonomous learning for robots in the context of brain-body interactions," Ph.D. dissertation, University of Southern California, 2021.

[23] B. Chen, P. C. Tai, R. Harrison, and Y. Pan, "Novel hybrid hierarchical-k-means clustering method (hk-means) for microarray analysis," in *2005 IEEE computational systems bioinformatics conference-workshops (CSBW'05)*. IEEE, 2005, pp. 105–108.

[24] J. H. Do and D.-K. Choi, "Clustering approaches to identifying gene expression patterns from dna microarray data," *Molecules and cells*, vol. 25, no. 2, pp. 279–288, 2008.

[25] P. D. Reeb, S. J. Bramardi, and J. P. Steibel, "Assessing dissimilarity measures for sample-based hierarchical clustering of rna sequencing data using plasmode datasets," *PLoS One*, vol. 10, no. 7, p. e0132310, 2015.

[26] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," 2000.

[27] B. Chen, P. Tai, R. Harrison, and Y. Pan, "Novel hybrid hierarchical-k-means clustering method (h-k-means) for microarray analysis," in *2005 IEEE Computational Systems Bioinformatics Conference - Workshops (CSBW'05)*, 2005, pp. 105–108.

[28] N. Tavakoli, S. Siami-Namini, M. Adl Khanghah, F. Mirza Soltani, and A. Siami Namin, "An autoencoder-based deep learning approach for clustering time series data," *SN Applied Sciences*, vol. 2, no. 5, pp. 1–25, 2020.

[29] P. J. Osseward, N. D. Amin, J. D. Moore, B. A. Temple, B. K. Barriga, L. C. Bachmann, F. Beltran Jr, M. Gullo, R. C. Clark, S. P. Driscoll *et al.*, "Conserved genetic signatures parcellate cardinal spinal neuron classes into local and projection subsets," *Science*, vol. 372, no. 6540, pp. 385–393, 2021.

[30] R. Tavenard, J. Faouzi, G. Vandewiele, F. Divo, G. Androz, C. Holtz, M. Payne, R. Yurchak, M. Rußwurm, K. Kolar, and E. Woods, "Tslearn, a machine learning toolkit for time series data," *Journal of Machine Learning Research*, vol. 21, no. 118, pp. 1–6, 2020. [Online]. Available: http://jmlr.org/papers/v21/20-091.html

[31] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.