Effect regulated projection of robot's action space for production and prediction of manipulation primitives through learning progress and predictability based exploration

Serkan Bugur[1], Erhan Oztop[2], Yukie Nagai[3] and Emre Ugur[1]

[1]Bogazici University, Istanbul, Turkey. [2]Ozyegin University, Istanbul, Turkey.
[3]National Institute of Information and Communications Technology, Osaka, Japan.

***Abstract*—In this study, we propose an effective action parameter exploration mechanism that enables efficient discovery of robot actions through interacting with objects in a simulated table-top environment. For this, the robot organizes its action parameter space based on the generated effects in the environment and learns forward models for predicting consequences of its actions. Following the Intrinsic Motivation approach, the robot samples the action parameters from the regions that are expected to yield high learning progress (LP). In addition to the LP-based action sampling, our method uses a novel parameter space organization scheme to form regions that naturally correspond to qualitatively different action classes, which might be also called action primitives. The proposed method enabled the robot to discover a number of lateralized movement primitives and to acquire the capability of prediction the consequences of these primitives. Furthermore our results suggest the reasons behind the earlier development of grasp compared to push action in infants. Finally, our findings show some parallels with data from infant development where correspondence between action production and prediction is observed.**

***Index Terms*—Intrinsic motivation, learning progress, sensorimotor development, primitive formation**

I. Introduction

PREDICTING the consequences of one's own actions is an important requirement for intelligent control and decision making in both biological and artificial systems. Neurophysiological data suggests that human brain benefits from internal forward models that continuously predict the outcomes of the generated motor commands for trajectory planning and movement control [1]. For higher-level cognitive functions, behavioral data suggests that forward prediction is used to generate plans to achieve different goals [2]. Infants start (learning of) predicting events and consequences of their own actions in early ages. Predictive ability is argued to already exist in 3-month old infants who can smoothly track sinusoidal targets through their eyes [3], and in 6-month old infants who can track objects that move behind occluders [4] or predict target objects of grasp actions of others [5]. On the motor side, motor development related to hand movements is likely to start already during pre-natal period and show intentionality already by 22 weeks of gestation [6]. [7] discusses how fetuses with innate capacity of detecting the consequences of their spontaneously generated activities can develop progressively more complex behaviors including reaction to sensory inputs, intentionality, goal-directed movements. Studies showed that fetuses exhibit kinematic patterns that are observed after birth. For example, hand to mouth and hand to object movement before and after birth was shown to follow similar patterns by [8]. Following a reflex like grasp behavior in 0-2 months, intentional power grasp develops in 4-6 months [9], and it takes 9 months for infants to reach for objects with correct hand-orientation [10]. With developing motor skills, the infants start learning the causality relations and object dynamics in response to various actions such as hitting, grasping and dropping [11]. It is plausible to think that while interacting with the environment, babies monitor the consequences of their actions and relate the consequences to their actions. Experimental evidence suggests that humans use multiple paired forward-inverse models to exhibit behaviors in different contexts rather than relying on a single highly complex model that can deal with all possible contexts [12]. How the motor and prediction capabilities develop, on the other hand, depends on the way infants interact with the environment. One prominent view posits that babies are endowed with an internal drive, named intrinsic motivation (IM) that allows them to actively choose the interactions they engage in for maximizing the speed and the extent of learning [13].

In this paper, inspired by infant development, we propose a developmental progression applicable to a robotic manipulator that interacts with its environment (Fig. 1). Starting with one parametric reach action, the robot organizes its action parameter space based on the generated effects in the environment and learns forward models for predicting consequences of its actions. Following the IM paradigm, the robot samples the action parameters from the regions that are expected to yield high learning progress (LP). In addition to the LP-based action sampling, our method uses a novel parameter space organization scheme to form regions that naturally correspond to qualitatively different action classes, which might be also called action primitives.

We implemented the proposal on a simulated robotic arm

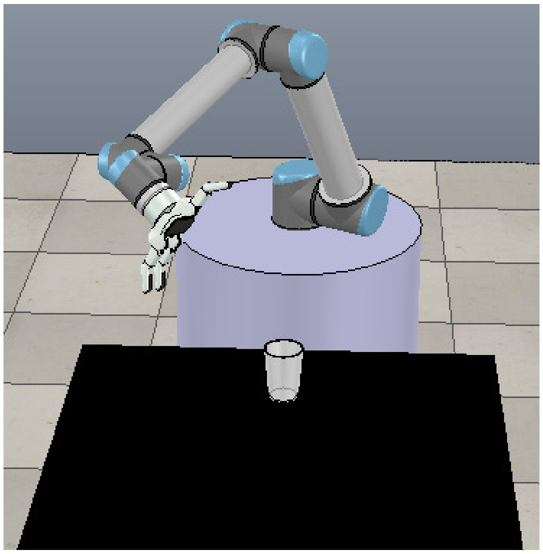


Fig. 1: A simulated manipulator robot interacts with an object using its 3-fingered gripper.

and gripper (Fig. 1), and found that lateralized grasp and push primitives emerge along with the continual sample-execute-learn-reorganize process. We observed an earlier developmental progression for the grasp action; i.e. forward models for predicting the external and the bodily consequences of grasp actions emerged earlier. To assess the parallels between actual infant development and the workings of our system, we compared our results with the behavioral data available in the literature on infant action prediction development. As a result, we found similar developmental timelines in motor and prediction capabilities of infants and our simulated system.

The contribution of this paper can be summarized as follows. First, the action parameter space is partitioned based on the developing capability of forward prediction of action consequences, which leverage the benefit of LP-based action sampling for further speedup in learning. Second, the action parameter sampling is done through a special latent space that represents the action parameters with a topological structure shaped by the effects generated by the actions. Finally, our system provides a developmental progression that has parallels with the infant data [5] where the correspondence between some features of action prediction and execution can be observed.

II. Related Work

A. Intrinsic Motivation (IM)

Intrinsic motivation is defined by Ryan and Deci [13] as “the inherent tendency to seek out novelty and challenges, to extend and exercise one’s capacities, to explore and to learn”. It has an internal locus of causality (i.e. the agent’s perception of the cause of success or failure) and is regulated by interest, enjoyment and inherent satisfaction consequently seems to be one of the essential causes of self-determined behaviors. As the intrinsic motivation promotes the learning in humans, it took attention of the researchers from the artificial intelligence domain as an alternative to the use of external signals for learning. As noted in [14], most of the developmental robotics researchers who work on IM tended to focus on the use of reinforcement learning (RL). While RL systems receive rewards from the external environment and learn policies that maximize the future expected external rewards, in IM systems the rewards are generated within/by the agent itself. As such, RL agents can be considered to be externally motivated systems whereas IM agents are intrinsically motivated.

In a detailed review of existing computational approaches to intrinsic motivation, Oudeyer and Kaplan [15] divided the approaches into two main categories. The first one, *competence-based intrinsic motivation* (CB-IM) stems from the competency measures that are obtained by the achievement of the agent’s goals and the second one, *knowledge-based intrinsic motivation* (KB-IM) relies on the discrepancy between reality and the agent’s expectations. Santucci et al. [16] showed that in order for the agent to acquire different skills from the environment, KB-IM based guidance is not adequate. They stated that the performance of the system improves when the link between the intrinsic motivation signal and the competence metric gets stronger. In KB-IM based systems the goal is to predict every possible future configuration of the environment, which in turn may be harder to predict or even unpredictable. CM-IM based systems, on the other hand, attempt to generate the intrinsic motivation signal by changing the target of the predictions as opposed to KB-IM based systems. Our proposed work can be considered in the scope of KB-IM approaches.

In one of the pioneer work [17] in knowledge-based IM, Oudeyer et al. argued that intrinsically motivated exploration strategies foster efficient and compelling learning in high-dimensional search spaces. To this end, they introduced the Intelligent Adaptive Curiosity (IAC) framework that drives the agent to maximize the “learning progress” that maximizes the change in prediction performance of the learners. Oudeyer and Kaplan [15] described “Learning Progress Motivation (LPM)” within the KB-IM that motivates the agent to decrease the prediction errors by using *learning progress* as the intrinsic motivation signal.

In order to enhance sampling efficiency and facilitate open-ended learning without external rewards, Hester and Stone [18] combined two intrinsic motivation signals, particularly the variance of the predictions and novelty. Similar to [18], Sequeira et al. [19] combined numerical correspondences of some emotional appraisal dimensions, namely, novelty, motivation, control, and valence as a source of intrinsic motivation. Both of these studies combine the different components of the intrinsic reward linearly. While the former finds the weights of the different components empirically, the latter optimizes them according to the agent’s fitness to the environment. Santucci et al. [20] proposed an architecture called GRAIL that includes a competence-based IM signal allowing the agent both to select from the existing goals and to generate new goals. In their experiments, a simulated humanoid robot that aims to learn how to reach targets that are located in different places in front of it. Hart et al. [21] demonstrated a framework that allows hierarchical structuring of control knowledge by means of the intrinsic motivation signal. In their experiments, they showed three learning stages of a manipulator robot. One of these stages was intended to make a grasp action. Temel et al. [22] introduced "frustration" and "impulsiveness" concepts into an intrinsically motivated reinforcement learning setup and used their proposed method for grasp-learning task. In their work, IM stems from frustration-based action selection mechanism that is regulated by the impulsiveness of the robot. In our previous work [23], we used learning progress to determine which action to explore next and diversity maximization to select the training objects in an affordance learning setup. In Duminy et al. [24], a robot applies intrinsic motivation in learning complex tasks in high-dimensional spaces in a hierarchical manner using heuristics of goal babbling, social learning and strategic learning; whereas in the current work we exploit effect-regulated projection of high-dimensional action space onto a latent space for effective exploration.

In general, the performance of the learning machines on the continuous search space highly depend on the distribution of the state-action pairs. In most of the scenarios, unfortunately, the sensorimotor space has a highly heterogeneous distribution of the state-action pairs, and calculating learning progress on the whole search space causes the signal to be unstable. Therefore, grouping similar state-action pairs is suggested in [17] as a simple but powerful approach to deal with the instability problem. Starting from a single group, the space was partitioned into groups based on the distribution of the learning points. In our current study, instead of using point distribution, the space is partitioned maximizing the performance of the learning machine.

B. Development of Reaching and Grasping

Six key computational ingredients for the development of reaching and grasping were identified in a detailed survey by Caligiore and Baldassare [25]: ecological active vision; motor babbling and associative learning processes; trial and error learning; hierarchical control architectures; embodiment; and finally the intrinsic motivation. They argue that intrinsic motivation leads the infant's cumulative learning of a large collection of skills in an adaptive manner. Gaussier et al. [26] emphasized processing of multi-modal information through their neuro-computational model in explaining reaching and grasping skills of infants. They proposed that the cortical memory may be organized as a reachable region map through the emergencies obtained from the body signals during the sensorimotor exploration. These body signals are the joints between the shoulder and wrist; tactile feedback from the hand; vision from the eye and, the sound of an object after contact. These aforementioned learning mechanisms may be modulated or bootstrapped by reinforcement learning guided by external rewarding stimuli, such as "joy of grasping" [27].

In our previous study [28], we proposed a three-staged developmental framework that is inspired from infant development. In the first stage, the robot was shown to discover movement primitives such as push, carry, and release through clustering the tactile feedback obtained during execution of a swing behavior. The discovered grasp and push primitives were used to learn object affordances and goal-based action execution in the second stage and for learning complex sequence of primitives through imitation and parental scaffolding in the last phase. Different from the previous work, the exploration of the robot is now guided by intrinsic motivation, and the primitives are formed in isolation in one-shot but learned in a hierarchical progression that is determined by the effect prediction capability.

C. Development of Action Production and Prediction

In our work, the action ability of our system develops together with the capability that enables prediction of action consequences. It is argued that others' actions can also be understood through a direct matching with self, which is believed to be realized by the mirror neuron system [29]. The mirror neurons are fired when an individual performs a particular action as well as when the individual observes another person performing the same or similar action. Interestingly, behavioral data suggest that such a mapping is not hardwired and probably depends on the motor development of the agent. For example, Sommerville et al. [30] showed that the infants can infer the goals of others' actions only if they develop the capability of executing the corresponding actions themselves. Kanakogi et al. [5] further supported this developmental account by conducting an experiment that involves examination of the infant's eye movements. In their experiments, infants were shown a hand reaching to objects in grasp posture, in push posture, and with an unfamiliar tool. The experiments showed that the infants exhibit predictive eye movement towards the target objects only after they learned to execute the grasp and push actions themselves, after 6 and 10 months, respectively. This data suggests that a developmental correspondence exists between action prediction and the ability to perform the same action.

Motivated from infant studies, Copete et al. [31] implemented a predictive learning based computational model and showed that the development of predicting action goals is correlated with the development of action production for a robot. In our study, on the other hand, our robot actively explores its action space, discovers action regions during development and selects which actions to learn through intrinsic motivation. Interestingly, such a developmental approach generated motor learning stages similar to infants. Furthermore, the findings of our study, where the robot learns predicting the rest of the hand trajectories have parallels with this experimental data as we discuss in Section VI.

III. Proposed System

In the proposed system, the robot is given a parametrized reach action which it can use to continually explore the parameter space while forming forward models to represent its action capabilities in terms of action effects and action contingencies. Importantly, during this process, the robot also re-organizes its motor space guided by the effects generated due to the interaction with the object. The main components of the proposed system are illustrated in Fig. 2. In the *bootstrapping phase*, the robot executes its reach action with random

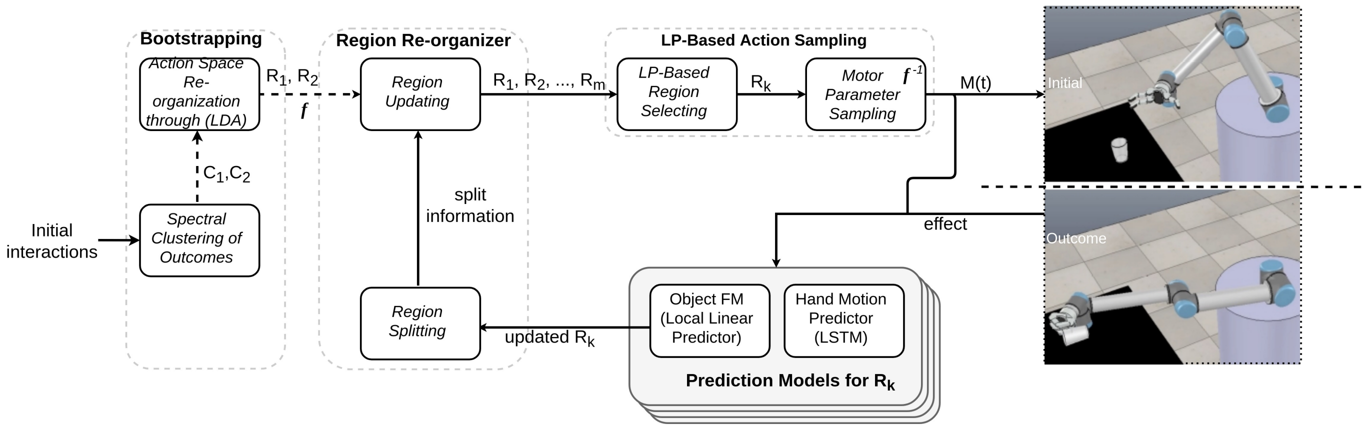


Fig. 2: Overall model of the proposed system. Bootstrapping phase takes the initial interactions, performs outcome clustering (C1, C2) and re-organizes the parameter space; resulting in two initial regions R1, R2, and the transformation function f. Region Re-organizer updates the regions and decides on the splits, LP-based Action Sampling calculates the LP for the regions and samples action ($M(t)$) parameters for the execution, and finally Prediction Models for the region predicts both the changes in the object state and trajectory of the end-effector given the action parameter.

parameters, observes the effects generated in the environment and groups similar effects into effect categories. Based on the corresponding effect categories, the action parameter space is transformed into a latent parameter space and partitioned into regions. For each region, a *forward model* is trained to predict action effects. After the bootstrapping phase, the robot starts *LP-based action sampling*, i.e. sampling of action parameters from the regions with the highest learning progresses. The actions from the selected regions are executed by the robot and the observed effects are used to update the corresponding forward models. If a particular region is explored too often, this region is divided into sub-regions, ensuring maximum prediction accuracy in the sub-regions as well. The proposed LP-based region sampling and prediction-accuracy based region splitting (LPPA) allows the robot to efficiently learn forward models along the development. The following subsections provide the details of these processes.

A. Bootstrapping phase

For continual region formation and refinement, our system requires a bootstrapping phase that finds initial action parameter regions and an action space transformation function. These regions are formed by using the initial sensorimotor experience created by randomly sampled actions.

1) Effect Clustering: For differentiating actions with different effects, clustering is applied in the effect space, and as a result, qualitatively distinct effect categories (C_1, C_2 in Fig. 2) are found. As the effect space might be high dimensional and composed of diverse set of variables, spectral clustering method is used for this purpose. We used spectral clustering method as it does not make any assumptions on the distribution of the data.

Effect clustering is also used to group the action parameters: the parameters that cause the same effect are grouped together and the formed groups compose the first set of sub-regions (R_1, R_2 in Fig. 2). The effect clustering is finally exploited to produce an operator that transforms the original action parameter space to a low dimensional space.

2) Action Space Re-organization: This module aims to re-organize action parameter space so as to facilitate effective splitting of the parameter space in the subsequent stages. An effective splitting in the subsequent phases require a low-dimensional space that well-reflects the interaction dynamics between the robot and the object. As each generated region is assigned to an outcome predictor in the next phases, interactions with similar outcomes are better located close to each other in the low-dimensional space where region generation occurs. The original high-dimensional action parameter space can be represented with a diverse set of variables such as approach direction vector (in meters), gripper aperture width (with a ratio) or wrist angle (in radians). Neither the original action parameter space nor an unsupervised dimensionality reduction in this space would provide a space where similar interactions are located together. One way to do this is to transform the original parameter space to a latent space where the similarity in effect space is preserved. For this, a supervised dimensionality reduction method, namely Linear Discriminant Analysis (LDA) is used where effect clustering results are used as the sample labels. Given the parameters and the corresponding effect categories, a projection function is computed to ensure well-separation of the effects in the formed lower dimensional latent parameter space. The formed latent parameter space defines the boundaries between the regions, and thus allows direct parameter sampling from the desired region. We used LDA as it is stable with few examples and computationally light as it is a linear method whose inverse can also be efficiently computed.

B. Prediction Models

The regions formed in the action parameter space may be considered as movement primitives that cause similar changes in the environment. As such, each primitive may be accompanied by predictive mechanisms for maintaining execution robustness, error detection, and recovery. Here, without digressing these motor control aspects, we focus on the predictive capacity that is developed along with motor re-organization. We envision prediction models that help motor control, namely object forward model and hand motion prediction model. The former deals with the prediction of effects of the executed actions in the environment, whereas the latter learns to predict the continuation of an ongoing hand movement action.

1) Object Forward Model: In our setting, given the action parameters, the object forward model estimates the change in the position and the orientation of the object after the interaction. In detail, action parameters that include approach direction, gripper aperture width and orientation of the hand are used as inputs to the object forward model that outputs the position and orientation change of the interacted object. For each formed new region, a new object forward model is generated and trained using the data from the corresponding region. Instead of a single complex predictor, we adopt the notion that multiple local linear predictors are less costly to use, once the responsibility region of each local predictor is determined. Here we do not address this computational step; but, focus only on the generation of local predictors. Interested readers are referred to [32], [33].

2) Hand motion prediction model: The hand motion prediction model aims to predict the upcoming hand trajectory given an initial portion of it. In our implementation, we propose individual models for each action defined by the initial parameter regions. In a biological system, such ability can be used for the benefit of the organism, for example detecting deviations from a planned action, and can be developed for example via associative learning. In particular, a spatio-temporal Hebbian mechanism can yield such an ability [34]. In our study, instead of implementing an associative learning system we adopted Long Short Term Memory (LSTM) network that emulates the desired functionality. In our system, the hand forward motion prediction model works with the 3d trajectory data pertaining to the hand (end-effector) of the robot. In detail, the 3d position trajectory of the end-effector of the robot prior to object contact is used as input of the model that outputs the rest of the end-effector trajectory. We used LSTM method as it is state-of-the-art in temporal data prediction [35].

C. LP-Based Action Sampling

LP-based selection guides the robot towards the regions that are neither too complicated nor too simple. Maintaining the desirable difficulty during the exploration is achieved by guiding the agent to regions that provide maximal learning progress. Therefore, after the bootstrapping phase, in each step, the robot selects the region where the learning progress is maximal. Recall that the regions are organized in the latent parameter space. Learning progress (LP) of a region is defined based on the actual increase in the mean prediction accuracy of the object forward model of the corresponding region ($Pred_i$) as follows:

$$LP_i(t+1) = \overline{\gamma}_i(t+1) - \overline{\gamma}_i(t+1-\tau)$$

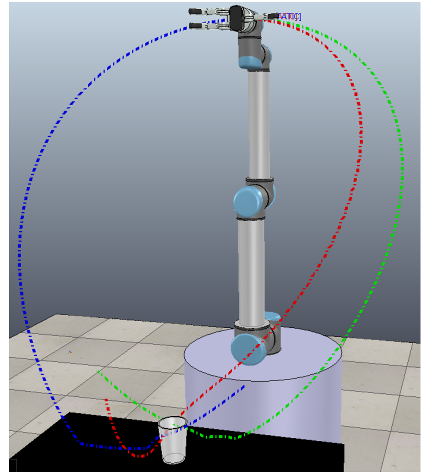


Fig. 3: Colored curves indicate three different trajectory examples that the end-effector of the robot followed in order to perform an action.

where $\overline{\gamma}_i(t+1)$ and $\overline{\gamma}_i(t+1-\tau)$ are defined as the current and previous mean prediction accuracies of the effect predictor, and τ is a time window, set to 5.

Here the mean prediction accuracy is defined by empirically setting a smoothing parameter ω to 25:

$$\overline{\gamma}_i(t+1) = \frac{\sum_{j=0}^{\omega} \gamma_i(t+1-j)}{\omega+1}$$

where the prediction accuracy of the region ($\gamma_i(t)$) is equal to the ratio of the correct predictions on objects explored by the action at step t. This is only a local measure that approximates the real accuracy. We used this local accuracy measure in our online incremental learning setup as the robot cannot access to ground truth, i.e. it cannot know the outcome of the actions without actually executing them in a real setting.

Finally, the next region to be explored is selected based on the above learning progress criteria using ϵ-greedy strategy [36].

$$R^t_{\text{sel}} = \begin{cases} R_r & \text{if } \zeta < \epsilon \\ \arg\max_{R_i} LP_{R_i}(t) & \text{otherwise} \end{cases}$$

where R^t_{sel} denotes the selected region at learning step t, R_r corresponds to a random region, $0 \leq \zeta \leq 1$ is a uniform random number, and ϵ is set to 0.10.

D. Region Partitioning

The robot explores regions through the LP-based active selection mechanism described in Section III-C. When the number of interactions sampled from a region exceeds a specified threshold, the corresponding region is split into two sub-regions as in [17]. Potential sub-regions are generated by sampling splitting points uniformly for each dimension in the latent parameter space. For each splitting point, the corresponding two potential sub-regions are formed and their prediction accuracies are calculated by training predictors with the corresponding set of interactions. Finally, the pair with the maximum prediction accuracy is selected to replace the parent region in the rest of the development.

IV. Experiments

The experimental setup is created in V-REP simulator [37] to include a six degrees of freedom manipulator (UR10) with a 3-finger gripper (Robotiq), which can interact with a cylindrical object placed in a table-top environment. Since our focus is to investigate how a given basic skill (reaching) can lead to motor specialization when sensorimotor exploration and adaptation are guided by prediction and learning progress measures, we kept the object identity and location fixed in the experiments reported in this paper.

A. Reach action

This basic action enables the robot to interact with the object from different approach directions, with different apertures and wrist orientations[1]. Different means of interactions is achieved through the following parametrization:

$$(x, y, z, \theta, \partial)$$

where (x, y, z) defines the approach vector towards the object, θ represents the wrist orientation and ∂ corresponds to the gripper aperture width. In each interaction, the gripper starts moving from the home position provided in Fig. 3 and follows a trajectory that passes through an approach position, the center of the object and a final position. The approach position is calculated by subtracting the approach vector from the object position. The final position is set to the symmetrically opposite side of the approach position with respect to the object center, but with higher elevation. The target joint angles that bring the gripper to the approach position, the object center, and the final position are calculated using inverse kinematics library[2]. Finally, a trajectory is generated in joint space through fitting a spline that connects these four points starting from the angles at the home position. While this trajectory is executed, the gripper angle θ and gripper aperture width ∂ are set to the desired values. At the time of contact of the gripper with the object, the fingers of the gripper are enclosed simulating grasp reflex. Finally, the robot arm is allowed to move to the final position after the gripper-object contact. Different reach trajectories with different parameters are overlaid on the snapshot of the robot at its home position in Fig. 4. Note that the effects of the interactions are not shown in this figure; depending on the parameters the object might be pushed away towards different directions by different parts of the gripper, or might be grasped by the gripper.

B. Data Collection

The action parameters are sampled from a uniform distribution from the following range:

[1]It is known that infants reach for an object with the intention of grasping by the age of 15 weeks [38]. Assuming that infants can reach to the objects successfully from early months, we designed our system such that the reach action is planned and executed without any error. However, infants have variability in perception of the world, and planning and execution of their actions; and this variability changes through their development. We did not model such variances and changes, therefore the parallels with the infant data is not conclusive but rather suggestive.

[2]www.orocos.org/kdl

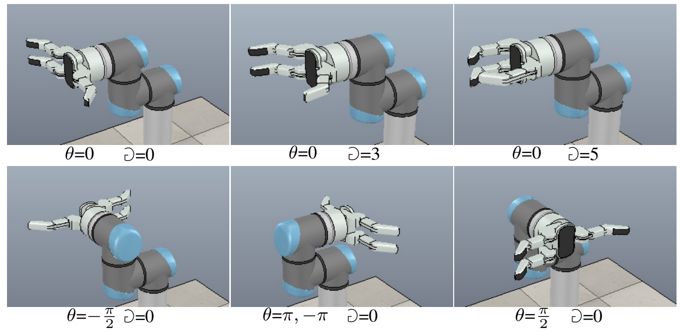


Fig. 4: Gripper configuration parameters, θ = raw angle, ∂ = gripper closeness degree.

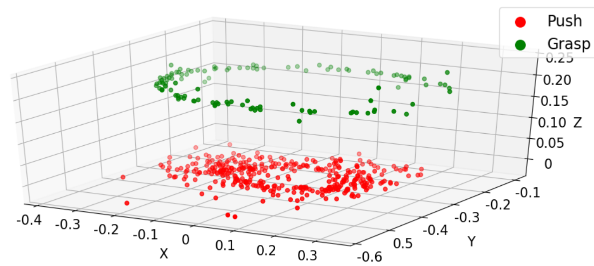


Fig. 5: Effect clustering results are provided. Each point shows the displacement of the object in 3d as the result of the interaction with the robot. Two clusters were formed and referred to as *Push* and *Grasp* clusters.

- For the approach position ((x, y, z)), an imaginary cylinder around the object with a radius of 30 cm and a height of 5 cm is generated. The cylinder is placed 5 cm above the table to avoid gripper-table collision during the execution. Finally, x, y and z are set to a random point on that cylinder.
- The gripper angle is set to a random value in the range of $[-\pi, \pi]$ radians. This effectively corresponds to a gripper orientation where the normal vector of gripper palm is parallel to the table plane, and is randomly rotated around the normal vector of the table plane (see Fig. 4 (a-d)).
- The gripper aperture width is set to a random value in a range ([0-5]) that fully opens and closes the fingers at its so-called basic grasping mode as shown in Fig. 4 (e-g).

In trajectory generation, the spline fit is performed using Bezier curves[3] as Bezier curves are argued to well model human-like reaching motions [39]. As mentioned before, the spline is fit in joint space considering the home position, the approach position, the object center and the final position.

In order to execute the action, 60 points are sampled from the corresponding spline and the corresponding joint angles are sent as target angles to the robot controller. The position and orientation of the object are extracted after the action execution. Finally, the change in the pose of the object is stored along with the action parameters in each interaction.

[3]https://github.com/chen0040/cpp-spline

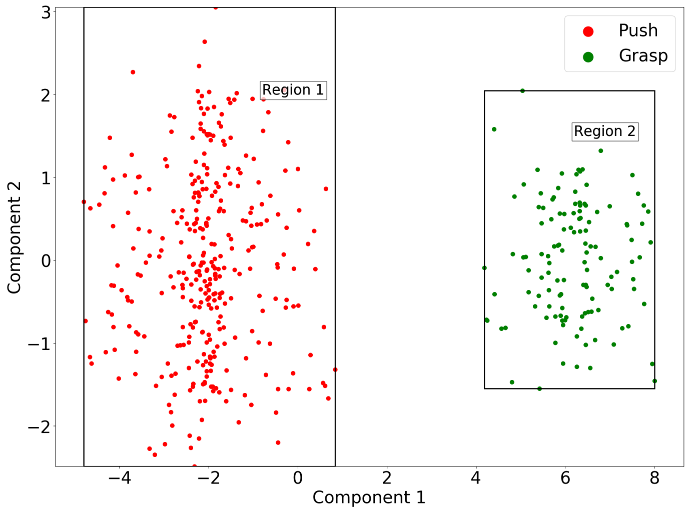


Fig. 6: Bootstrapping phase interaction instances are plotted in the 2d the latent parameter space formed by the LDA. The red and green points correspond to the clusters formed through effect clustering, and referred to as *Push* and *Grasp* regions.

V. Results

A. Developmental progress in a single run

In this section, the exploration behavior of the robot is analyzed, the generated regions are presented, and the change in the performance by the proposed method is assessed. For this purpose, the agent was allowed to sample 3000 interactions, which correspond to 3000 sets of action parameters, based on the proposed LPPA method.

a) Initial effect clusters: In this subsection, we provide the effect clusters that were formed in the bootstrapping phase. The robot used 500 random interactions for this purpose, forming two initial regions by applying spectral clustering on the effect space. In Fig. 5, the points correspond to the end position of the object subsequent to interactions of the robot, where colors indicate the clusters found. The interactions that increase the elevation of the object are clustered in the *Grasp* group represented by the green color, and the interactions that only change the position of the object on the table plane are clustered in the *Push* group, which is represented by the red color. Note that, the number of the points belonging to the *Grasp cluster* is almost half of the number of points belonging to *Push cluster*. The reason is that the class instances were unevenly distributed in the uniformly sampled data where the push effect was observed twice as much as the grasp effect.

b) Initial regions: In this subsection, we presented the result of action space re-organization and the generated regions in the latent parameter space. For this, the 5d action parameter space, which was composed of the 3d approach vector, the gripper aperture width and the gripper orientation, was projected to 2d latent space using LDA method. Recall that the clusters obtained in the spectral clustering were used as labels for the projection performed in LDA. Further, the axis weighting used in LDA was based on the most discriminating component (i.e. the z component) of the cluster means found by the spectral clustering. In Fig. 6, the interaction instances are plotted in the 2d Linear Discriminant space. Since z position was found to have a high weight in LDA computation, well-separable regions in the LD space are observed. The red and green colors represent the interactions from the first and second regions, referred to as *Push* and *Grasp* regions, respectively, in the rest of this section. The boundaries of these initial two regions are shown with the black bounding boxes.

c) Progressive region splitting results: After the bootstrapping phase, where the action parameter space was re-organized and the first two regions were found, the robot executed 2500 actions following our LPPA algorithm. The regions generated during the exploration are provided by hierarchical region split tree in Fig. 7. Note that given 2 dimensions and 30 cutting values, 60 pairs of potential child regions were obtained in each split step. Here the root node corresponds to the initial region, the nodes in the second level correspond to the *Push-Region* and the *Grasp-Region* obtained from the initial set of 500 random interactions; and the terminal nodes show the final regions obtained after making 3000 interactions. The boxes provide the region index and from which value and dimension the parent region was partitioned into the child regions. Fig. 9 provides a visualization of the regions overlaid in the latent parameter space where each vertical bar shows the corresponding division boundary, the region indices are shown with the numbers in the boxes. The *Grasp-Region* *Grasp-Region* were divided two and three times, respectively, generating two and three regions at the end. These regions correspond to the terminal terminal nodes in Fig. 7. Analyzing the region indices reveals the fact that, the splits in *Grasp-Region* occurred earlier than the splits in *Push-Region*. This splitting mechanism generates a compact and semantically meaningful set of regions that lead to an effective learning as will be discussed in the rest of this section. For comparison, Fig. 8 provides a split tree that was directly generated in the original 5d action parameter space without action space re-organization.

d) Region exploration: In this part, the region exploration strategy of our system and the order of emergence of the

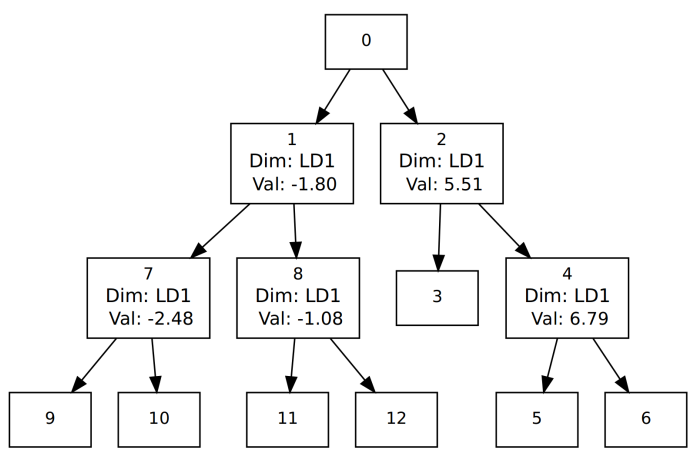


Fig. 7: The visualization of the generated regions. The nodes 1 and 2 correspond to *Push* and *Grasp* regions, the child nodes correspond to their sub-regions.

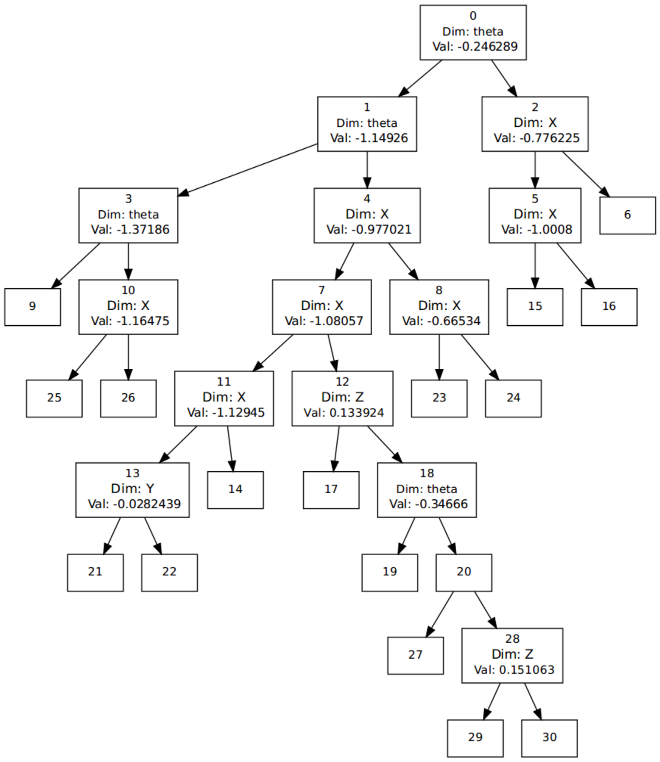
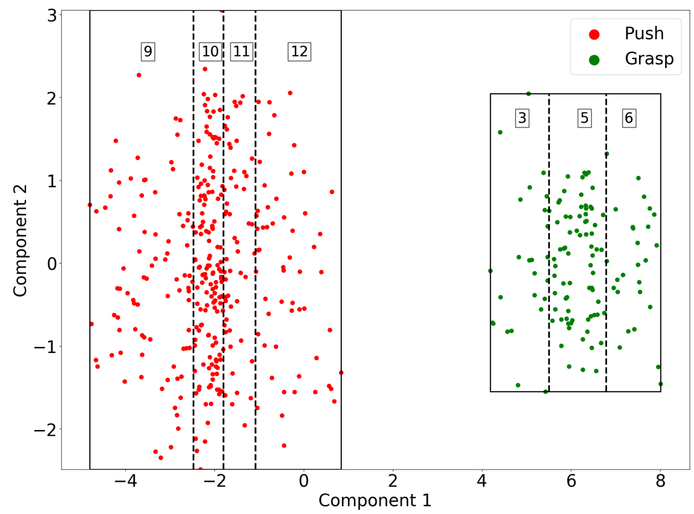


Fig. 8: The visualization of the generated regions without action space re-organization.



(a)

Fig. 9: Terminal regions in LD space. Dashed lines represent the cutting value and dimension for the corresponding split. Region numbers are also shown in boxes, smaller the region number, earlier the split to form that region. 3 terminal regions emerged for grasping whereas for pushing 4 terminal regions emerged.

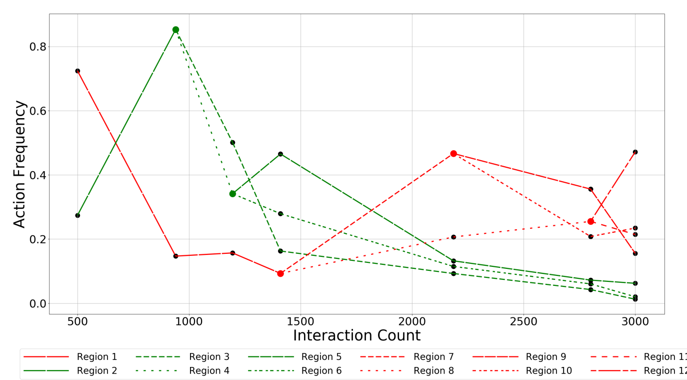


Fig. 10: Action frequencies of all the regions over the 3000 interactions. Red lines correspond to pushing regions whereas green lines correspond to grasping regions. Split points are shown with red and green circles for pushing and grasping regions respectively. The frequencies are provided from 500 interactions as there is no active sampling earlier.

sub-regions are analyzed. Fig. 10 demonstrates the exploration frequencies of the regions along the developmental timeline of the robot. The frequencies are provided from 500 interactions as there was no active sampling earlier. As discussed earlier, *Push-Region* was explored twice as much as *Grasp-Region* due to the uneven effect distribution with the uniform action parameter sampling for the initial 500 interactions. After this point, the learning progress in *Grasp-Region* was larger than the one in the *Push-Region*. Therefore, the actions were sampled from the grasp region and after the maximum number of interactions were reached, the *Grasp-Region* was partitioned into two at around $t = 900$. One of the emerged regions were split into two again and the frequency of exploration of these grasp regions gradually decreased due to low learning progress. The *push-region*, on the other hand, started being explored more after around $t = 1900$ and continued to be explored more until the end.

e) Detailed inspection of the generated regions: The regions are further analyzed in order to understand why such splitting occurred, and what the final regions correspond to, as the visualization on the latent parameter space does not directly provide this information. Consequently, the regions were back-projected into the original motor parameter space and the mean values of each region is inspected. Fig. 11 provides the mean motor parameter values for (x, y, θ) for each final region. For example, regions 3, 5 and 6 correspond to the regions that were emerged from the *Grasp-Region*. Based on the parameter values, regions 3 and 6 correspond to approach movement from the left and the right, respectively, with the wrist angles oriented towards the object. Region 5, on the other hand, is a mixture of left and right approaches. These results show that the system found *grasp-from-left* and *grasp-from right* primitives. There is similar lateralization for the push primitives as well, where regions 9 and 12 correspond to *push-from-left* and *approach-from-right* with the back side of the gripper. Fig. 12 provide snapshots from interactions instances observed in regions 3, 6, 9 and 12. These interactions correspond to the mean point of the regions in the latent parameter space.

B. Analysis of results from independent runs

In the previous section, the results of a single developmental progress were provided in detail. This section aims to assess the overall performance of the proposed method. Therefore, the developmental learning procedure is repeated 10 times with different initial seeds that practically correspond to different initial 500 interaction samples for bootstrapping phase, which at the end causes different sets of regions to emerge. In this section rather than providing the individual results of these 10 different runs, the statistics are provided as region groups in the rest of this section.

1) Developmental order: Fig. 13a provides the exploration frequencies of action parameters for regions that emerge from grasp and push regions. The squares and the bars provide the mean frequencies and the standard deviations for the 10 independent runs, respectively. In the beginning, push region motor parameter were sampled twice as much as the grasp motor parameters due to the uneven grasp/push distribution obtained from random parametrization. With our LPPA method, the robot focused on exploring the grasp region until around $t = 2500$. Fig. 13b provides the learning progress statistics of the corresponding regions, and as shown at around $t = 2500$, the learning progress of push regions exceeded the learning progress of grasp regions. Therefore, all the runs changed their exploration strategy, focusing more on the push regions towards the end.

As shown in Fig. 12a and Fig 12b, the higher the learning progress of an action, the more it is explored by the robot. Furthermore, an analysis of Figs. 12a and Fig. 12b together shows the correspondence between changes in action exploration frequencies and in variances of LPs. High amount of exploration for grasp and push actions in the initial and latter stages of development generates more data points and hence less variance in their learning progress.

2) Efficiency in learning: In this subsection, we compared the learning speed of our system with two alternative strategies. The first strategy, named as *random sampling*, randomly samples regions to explore and follows our prediction accuracy based sampling approach. The second strategy, names as *variance-based splitting*, samples the regions based on their

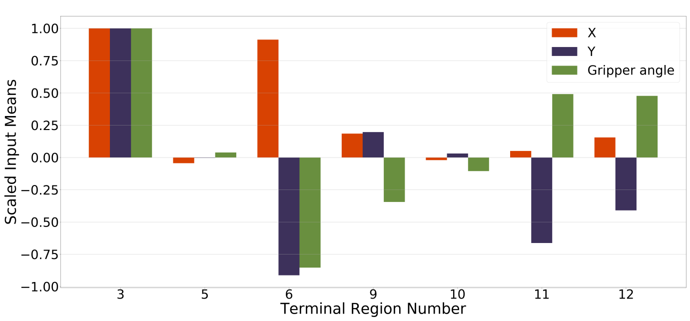


Fig. 11: Scaled mean values for the action space of the terminal regions are shown for a single run. 2 of the action parameters which are y and ∂ are not shown in the figure since no significant difference is seen in the distribution of these parameters.

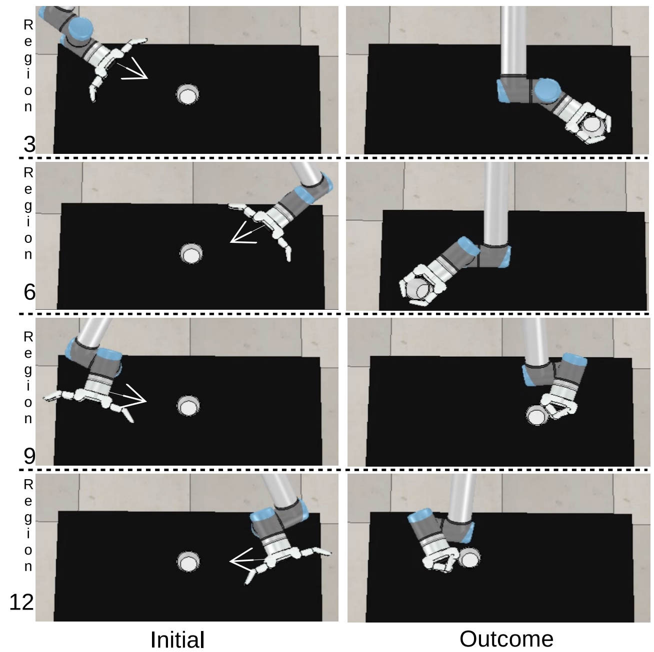
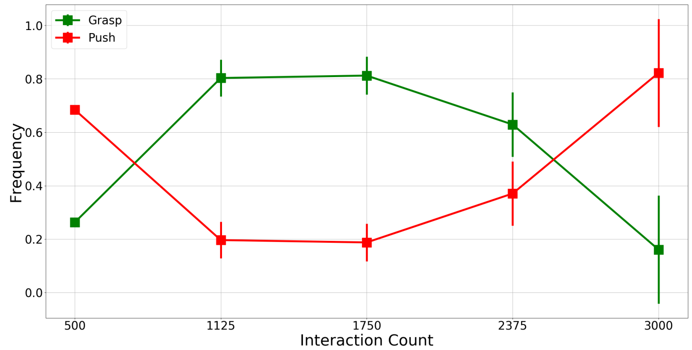


Fig. 12: Snapshots of example robot executions from regions 3, 6, 9 and 12 are provided from top to bottom. The robot at the selected approach positions and at the final configurations are shown on the left and right respectively.

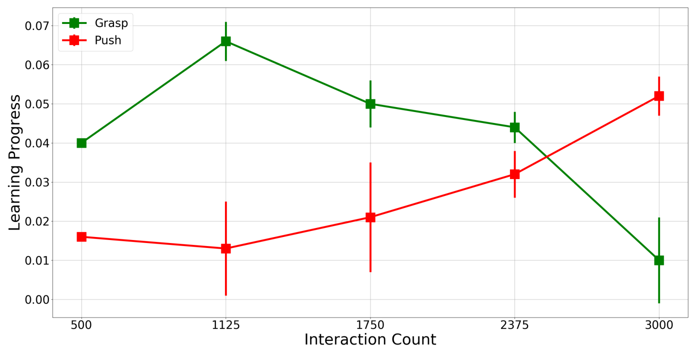
LP, but uses variance based splitting criteria of Oudeyer et al. [17]. 10 independent runs were performed using all three strategies, including our LPPA, and their generalization performances are compared in Fig. 14. As shown, the error decreased with more interactions in both variance-based splitting and LPPA, and decrease with LPPA exceeded the alternative model after 1500 interactions. This result shows that the prediction accuracy based region partitioning strategy employed by LPPA result in faster learning compared to the region splitting approach that only used the distribution of the data in the regions.

C. Development of hand motion prediction model

In this section, the change in hand motion prediction model performance along the developmental timeline is analyzed. Given the initial 20 steps of the trajectory, the robot learns to predict the rest of the trajectory by training an LSTM network in each region. Fig. 15 provides the prediction results for grasp and push regions along the developmental timeline. After the bootstrapping phase, the error in the grasp region had a sharp decrease compared to the error drop in the push region. Towards the end of the developmental timeline set as 3000 interactions, the performance of predicting the hand trajectory in push region becomes similar to the performance in grasp region. This progress in performance is the result of sampling from different regions in different amounts during the developmental timeline.



(a) The exploration frequencies of action parameters for regions that emerge from grasp and push regions.



(b) The average LP fluctuation of the actions corresponding to grasping and pushing actions.

Fig. 13: The average frequencies of the explored actions (13a) and LP fluctuation of the actions (13b). The squares and the bars provide the mean frequencies and the standard deviations for the 10 independent runs respectively.

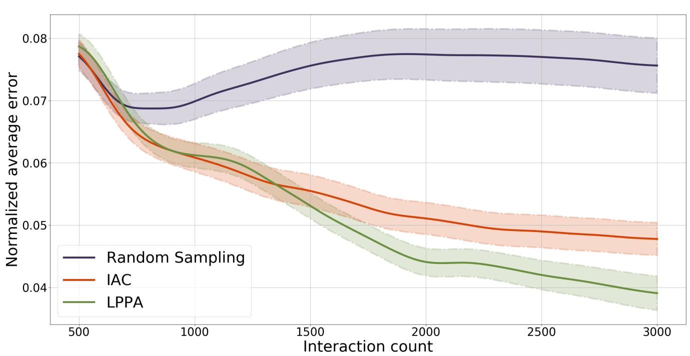


Fig. 14: Comparison between the error plots for 10 runs. The blue line represents the average error for 10 runs throughout the exploration of the robot when all the actions are selected randomly (LP not used) whereas red and green lines represent the splitting mechanism of [17], and splitting mechanism of our method (LPPA) respectively. Shaded areas represent the variance.

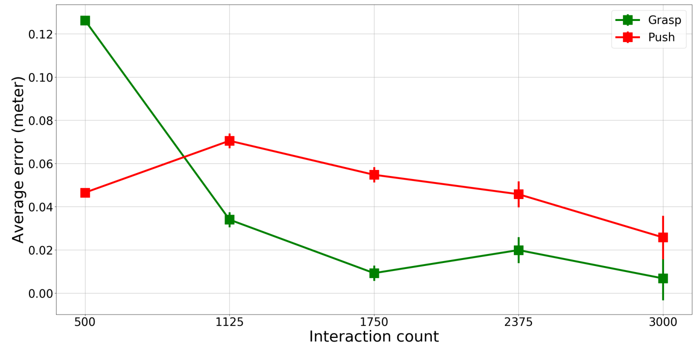


Fig. 15: The average errors for trajectory prediction. The red plot represents errors corresponding to pushing actions whereas the green plot represents errors for grasping actions. The squares and the bars provide the mean frequencies and the standard deviations for the 10 independent runs.

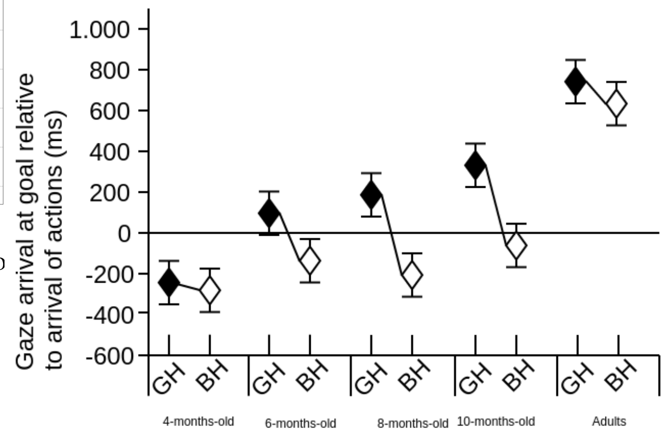


Fig. 16: Adapted from [5], the relative times of the arrival of the gaze of the infant at the goal area. The time of the agent's actions in each condition/age group is represented by a horizontal line at 0 ms. Black and white diamonds indicate the grasping (GH) and back hand pushing (BH) respectively. Positive values imply the gaze precedes the action and similarly, negative values imply the gaze arrival is later than the action. Error bars are standard error of the mean.

VI. Discussion and Conclusion

Action prediction and production correspondence: In Section II, we introduced an infant study from literature in which the development of the ability to predict the target of certain actions (push or grasp) was dependent upon or facilitated by the capacity to perform the action in consideration [5]. Fig. 16 illustrates the results of that experiment where gaze arrival times to the target objects of ongoing actions are shown for different conditions and different ages. GH and BH represent the grasp-hand and back-hand (push) postures. Positive and negative gaze arrival times correspond to gaze arrival to the target object before and after the hand-object contact. As shown, the performance of predictive eye movement develops first for grasp movement compared to push movement. While timing of gaze arrival was used as a measure of infant prediction capability in [5], we used the performance of the hand trajectory predictors as the measure for the capability of predicting actions in our work (Fig. 15). While the results measure different variables, the parallels in the results suggest that the action prediction depends on motor development in both cases, which in turn depends on the amount of experience.

Action re-organization: As mentioned before, the robot is required to partition the parameter space into regions for further processing. However, in high dimensional parameter spaces with non-homogeneous component scales, the partitioning procedure is not straightforward. In [17], [40], the parameter space was partitioned by considering a single parameter at a time. These methods assumed that underlying regions in the parameter space are organized as hyper-rectangles. However, this assumption does not hold in many cases, especially when one considers high dimensional spaces with parameters having diverse types, semantics, metrics, etc. Therefore, in this study, we proposed to re-organize the parameter space by preserving the effect related similarity while reducing the effective dimensions. To this end, a supervised dimensionality reduction method that can impose the effect topology in the formed latent action space is used. This way, the parameters that caused different types of outcomes were ensured to be well-separated in the latent space.

Action distribution: In the bootstrapping phase, the parameters of the sampled actions followed a uniform distribution. However, during development we observed that the reach actions that approach from the opposite side of the object vanished. Further inspection revealed that because of the robot kinematics, the hand collided with the object while moving towards the approach position at the opposite side, pushing the object in an seemingly random way, thus making it impossible to predict the action consequences for the corresponding action parameters. Consequently, the LP based sampling favored to explore more direct reach actions in both grasping and pushing. This is consistent with one of the main characteristics of reaching movements: straight paths towards the object [41].

Locality of the learned representations: Each object forward model emerged in the system is trained in a similar way: Given the parameterized action inputs, they predict the displacement in the position and orientation of the object. Action input parameters consist of 3d approach direction around the object, wrist orientation and gripper aperture width. Because the object is put to the same position in the experiments, the system learns locally around the object. As action parameters are defined with respect to object, if the object is placed to a reachable local neighbourhood of the trained position, similar results are expected to be seen. But, kinematic constraints might limit generalization to the full workspace.

Region splitting: Our splitting mechanism, which partitioned the 2d latent space into rectangular grids from uniformly distributed potential values, found semantically meaningful sub-regions, and generated significantly better performance compared to the original variance based splitting mechanism. However, if the underlying sub-regions were distributed with non-linear separation boundaries, more advanced splitting methods would be required. To deal with this problem, [42] used unsupervised Growing Neural Gas clustering method to find possibly non-linear region boundaries. However the categorization was still done in an unsupervised independent of the prediction error, and the prediction error was only used in deciding whether to generate new regions in their case. In our work, on the other hand, the categorization is directly performed based on the prediction errors. The most similar approach to our work in terms of splitting is by [43], where partitioning was also performed with hyperplanes perpendicular to one dimension. They maximized the dissimilarity of learning progress comparing the two created regions, whereas we minimized the overall prediction error of the created regions. While these two splitting mechanisms share the common idea of using the performances of the predictors, only with effect-regulated projection of the action parameter space onto the latent parameter space, we were able to obtain sub-regions that could be linearly separated. However, as no merge or remove operation is defined over regions, it is not possible to adapt to significant changes in an efficient way. In case a significant change is introduced in the underlying conditions, such as changes in robot morphology or kinematics (e.g. broken fingers, limited arm joint angles, etc), our system will attempt to deal with such changes by starting from the current sub-regions and splitting them into even smaller sub-regions that encode the new localities in the new settings. This would correspond to an over-segmented region generation compared to starting from scratch.

In summary, the proposed LPPA method enabled the robot to explore its parameter space efficiently and effectively, to enable it to discover a number of specialized movement primitives for which predictive forward models and prediction models are also built. Furthermore, the parallels between our results and the development of infant action prediction capacity suggests that LPPA based mechanisms may be employed by the sensori-motor systems of developing infants.

Acknowledgment

This research was partially supported by JST CREST "Cognitive Mirroring" [grantnumber: JPMJCR16E2] and Bogazici Research Fund (BAP) project IMAGINE-COG++ [grant number: 18A01P5]. The authors would like to thank to Melisa Idil Sener and Mert Imre for their comments on the manuscript.

References

[1] Mitsuo Kawato. Internal models for motor control and trajectory planning. *Current opinion in neurobiology*, 9(6):718–727, 1999.

[2] Mark Steedman. Plans, affordances, and combinatory grammar. *Linguistics and Philosophy*, 25(5-6):723–753, 2002.

[3] Claes Von Hofsten and Kerstin Rosander. Development of smooth pursuit tracking in young infants. *Vision research*, 37(13):1799–1810, 1997.

[4] Scott P Johnson, J Gavin Bremner, Alan Slater, Uschi Mason, Kirsty Foster, and Andrea Cheshire. Infants perception of object trajectories. *Child Development*, 74(1):94–108, 2003.

[5] Yasuhiro Kanakogi and Shoji Itakura. Developmental correspondence between action prediction and motor ability in early infancy. *Nature communications*, 2:341, 2011.

[6] Stefania Zoia, Laura Blason, Giuseppina DOttavio, Maria Bulgheroni, Eva Pezzetta, Aldo Scabar, and Umberto Castiello. Evidence of early development of action planning in the human foetus: a kinematic study. *Experimental Brain Research*, 176(2):217–226, 2007.

[7] Jaqueline Fagard, Rana Esseily, Lisa Jacquey, J ORegan, and Eszter Somogyi. Fetal origin of sensorimotor behavior. *Frontiers in Neurorobotics*, 12:23, 05 2018.

[8] Stefania Zoia, Laura Blason, Giuseppina DOttavio, Marina Biancotto, Maria Bulgheroni, and Umberto Castiello. The development of upper limb movements: from fetal to post-natal life. *PloS one*, 8(12):e80876, 2013.

[9] R Jason Gerber, Timothy Wilks, and Christine Erdie-Lalena. Developmental milestones: motor development. *Pediatrics in review*, 31(7):267–277, 2010.

[10] David A Rosenbaum. *Human motor control*. Academic press, 2009.

[11] Minoru Asada, Koh Hosoda, Yasuo Kuniyoshi, Hiroshi Ishiguro, Toshio Inui, Yuichiro Yoshikawa, Masaki Ogino, and Chisato Yoshida. Cognitive developmental robotics: A survey. *IEEE transactions on autonomous mental development*, 1(1):12–34, 2009.

[12] Daniel M Wolpert and Mitsuo Kawato. Multiple paired forward and inverse models for motor control. *Neural networks*, 11(7-8):1317–1329, 1998.

[13] Richard M Ryan and Edward L Deci. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American psychologist*, 55(1):68, 2000.

[14] Angelo Cangelosi and Matthew Schlesinger. *Developmental robotics: From babies to robots*. MIT Press, 2015.

[15] Pierre-Yves Oudeyer and Frederic Kaplan. What is intrinsic motivation? a typology of computational approaches. *Frontiers in neurorobotics*, 1:6, 2009.

[16] Vieri Giuliano Santucci, Gianluca Baldassarre, and Marco Mirolli. Which is the best intrinsic motivation signal for learning multiple skills? *Frontiers in neurorobotics*, 7:22, 2013.

[17] Pierre-Yves Oudeyer, Frédéric Kaplan, and Verena V. Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11:265–286, 2007.

[18] Todd Hester and Peter Stone. Intrinsically motivated model learning for developing curious robots. *Artificial Intelligence*, 247:170–186, 2017.

[19] Pedro Sequeira, Francisco S Melo, and Ana Paiva. Emotion-based intrinsic motivation for reinforcement learning agents. In *International Conference on Affective Computing and Intelligent Interaction*, pages 326–336. Springer, 2011.

[20] Vieri Giuliano Santucci, Gianluca Baldassarre, and Marco Mirolli. Grail: a goal-discovering robotic architecture for intrinsically-motivated learning. *IEEE Transactions on Cognitive and Developmental Systems*, 8(3):214–231, 2016.

[21] Stephen Hart, Shiraj Sen, and Rod Grupen. Intrinsically motivated hierarchical manipulation. In *2008 IEEE International Conference on Robotics and Automation*, pages 3814–3819. IEEE, 2008.

[22] Ercin Temel, Beata Joanna Grzyb, and Sanem Sariel. Learning graspability of unknown objects via intrinsic motivation. In *AIC*, pages 98–109. Citeseer, 2014.

[23] Emre Ugur and Justus Piater. Emergent structuring of interdependent affordance learning tasks using intrinsic motivation and empirical feature selection. *IEEE Transactions on Cognitive and Developmental Systems*, 9(4):328–340, 2017.

[24] Nicolas Duminy, Sao Mai Nguyen, and Dominique Duhaut. Learning a set of interrelated tasks by using a succession of motor policies for a socially guided intrinsically motivated learner. *Frontiers in Neurorobotics*, 12:87, 2019.

[25] Daniele Caligiore and Gianluca Baldassarre. The development of reaching and grasping: Towards an integrated framework based on a critical review of computational and robotic models. In *Reach-to-Grasp Behavior*, pages 335–364. Routledge, 2018.

[26] Philippe Gaussier and Alexandre Pitti. Reaching and grasping: what we can learn from psychology and robotics. *hal-01609659*, 2017.

[27] Erhan Oztop, Nina S Bradley, and Michael A Arbib. Infant grasp learning: a computational model. *Experimental brain research*, 158(4):480–503, 2004.

[28] Emre Ugur, Yukie Nagai, Erol Sahin, and Erhan Oztop. Staged development of robot skills: Behavior formation, affordance learning and imitation with motionese. *IEEE Transactions on Autonomous Mental Development*, 7(2):119–139, 2015.

[29] Giacomo Rizzolatti and Laila Craighero. The mirror-neuron system. *Annu. Rev. Neurosci.*, 27:169–192, 2004.

[30] Jessica A. Sommerville, Amanda L. Woodward, and Amy Needham. Action experience alters 3-month-old infants' perception of others' actions. *Cognition*, 96:b1–b11, 2005.

[31] Jorge Luis Copete, Yukie Nagai, and Minoru Asada. Motor development facilitates the prediction of others' actions through sensorimotor predictive learning. *2016 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 223–229, 2016.

[32] Stefan Schaal, Christopher G Atkeson, and Sethu Vijayakumar. Scalable techniques from nonparametric statistics for real time robot learning. *Applied Intelligence*, 17(1):49–60, 2002.

[33] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, Geoffrey E Hinton, et al. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[34] Thierry Chaminade, Erhan Oztop, Gordon Cheng, and Mitsuo Kawato. From self-observation to imitation: Visuomotor association on a robotic hand. *Brain research bulletin*, 75(6):775–784, 2008.

[35] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016.

[36] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[37] Eric Rohmer, Surya P. N. Singh, and Marc Freese. V-rep: A versatile and scalable robot simulation framework. *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1321–1326, 2013.

[38] Claes von Hofsten and Karin Lindhagen. Observations on the development of reaching for moving objects. *Journal of experimental child psychology*, 28(1):158–173, 1979.

[39] Julian J Faraway, Matthew P Reed, and Jing Wang. Modelling three-dimensional trajectories by using bézier curves with application to hand motion. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 56(5):571–585, 2007.

[40] Melisa I Sener and Emre Ugur. Partitioning sensorimotor space by predictability principle in intrinsic motivation systems. In *2018 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 54–59. IEEE, 2018.

[41] Amir Karniel and Gideon F Inbar. A model for learning human reaching movements. *Biological cybernetics*, 77(3):173–183, 1997.

[42] Rachel Lee, Ryan Walker, Lisa Meeden, and James Marshall. Category-based intrinsic motivation. In *Proceedings of the ninth international conference on epigenetic robotics*, volume 146, pages 81–88, 2009.

[43] Adrien Baranes and Pierre-Yves Oudeyer. Robust intrinsically motivated exploration and active learning. In *2009 IEEE 8th International Conference on Development and Learning*, pages 1–6. IEEE, 2009.

Serkan Bugur obtained a Bachelor Degree in Computer Engineering from Bogazici University and is pursuing Masters Degree in the same department. He is an Active Member of the Cognition, Learning and Robotics Research Group (CoLoRs) in Bogazici University. His research is focused on Humanoid Robots, Cognitive and Developmental Robotics, Artificial Intelligence, Human-Robot Interaction, Robot Learning.

Erhan Oztop earned his Ph.D. at the University of Southern California in 2002. In the same year, he joined the Computational Neuroscience Laboratories at the Advanced Telecommunications Research Institute International, (ATR) in Japan. There he worked as a researcher and later a senior research and group leader where he also served as vice department head for two research groups and held visiting associate professor position at Osaka University (2002-2011). Currently, he is a faculty member and the chair of the Computer Science Department at Ozyegin University, Istanbul. His research involves computational study of intelligent behavior, human-in-the loop systems, computational neuroscience, machine learning, cognitive and developmental robotics.

Yukie Nagai received the masters degree from Aoyama Gakuin University, Tokyo, Japan, in 1999, and the Ph.D. degree from Osaka University, Osaka, Japan, in 2004, both in engineering. She was a Post-Doctoral Researcher with the National Institute of Information and Communications Technology (NICT), Kyoto, Japan, from 2004 to 2006, and with Bielefeld University, Bielefeld, Germany, from 2006 to 2009. She was then a Specially Appointed Associate Professor with Osaka University from 2009 to 2017, and became a Senior Researcher with NICT in May 2017. Her research interests include computational modeling of human cognitive functions such as self-other cognition, imitation, and joint attention, and design of assistant systems for developmental disorders. She was the Project Leader of MEXT Grant-in-Aid for Scientific Research on Innovative Areas Computational modeling of social cognitive development and design of assistance systems for developmental disorders from 2012 to 2017, and is the Project Leader of JST CREST Cognitive Mirroring: Assisting people with developmental disorders by means of self understanding and social sharing of cognitive processes since 2016.

Emre Ugur is an assistant professor in Department of Computer Engineering Department, Bogazici University, Turkey. After receiving his PhD in Computer Engineering from Middle East, he worked at ATR Japan as a researcher (2009-2013), at University of Innsbruck as a senior researcher (2013-2016), and at Osaka University as specially appointed assistant professor (2015, 2016). He participated in several EU funded projects, including Xperience, ROSSI, MACS and Swarm-bots, and is currently PI of IMAGINE project supported by European Commission, Horizon 2020 Programme. He is interested in developmental and cognitive robotics, and intelligent and adaptive manipulation.