2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Hierarchical Interest-Driven Goal Babbling for Efficient Bootstrapping of Sensorimotor skills

Rania Rayyes, Heiko Donat and Jochen Steil

***Abstract*— We propose a novel hierarchical online learning scheme for fast and efficient bootstrapping of sensorimotor skills. Our scheme permits rapid data-driven robot model learning in a "learning while behaving" fashion. It is updated continuously to adapt to time-dependent changes and driven by an intrinsic motivation signal. It utilizes an online associative radial basis function network, which is the first associative dynamic network to be constructed from scratch with high stability. Moreover, we propose a parameter-sharing technique to increase efficiency, stabilize the online scheme, avoid exhaustive parameter tuning, and speed up the learning process. We apply our proposed algorithms on a 7-DoF physical robot manipulator and demonstrate their performance and efficiency.**

I. Introduction

It is widely accepted since the 1990's that the human motor control is organized on the basis of forward and inverse models (i.e., the relation between actions/motor commands and outcomes) [1], these also play a main role in motor control architectures in robotics and obtaining them is essential for any embodied agent to master its body. Learning inverse models of robot manipulators, i.e., estimate the motor command required to achieve a desired outcome, has been considered as a promising alternative solution to analytical methods since obtaining an accurate analytical model for dexterous high degrees of freedom (DoF) robots and soft robots requires a lot of engineering knowledge and can be challenging if no accurate parameters are available [2], [3].

Learning robot models and skills has also been a core research topic of the developmental and cognitive robots [4]. The developmental robots autonomously develop and adapt in open-ended environments through lifelong learning [5]–[7]. In contrast to the industrial robots, which are used to accomplish predefined tasks, the developmental robots have to solve unforeseen and unpredictable challenges, acquire repertoires of skills and should be versatile, flexible, and able to adapt to time-dependent changes (e.g., friction [3] or tool usage [8], [9]). That makes online learning and online adaptation essential requirements. Therefore, intrinsically motivated goal-directed methods, which rely on online data-driven learning, have gained a lot of attention recently. Interest-driven or curiosity-driven behaviors have been observed in growing children. They get bored by already known things and try to discover new ones and learn new skills [10]. This inspires the idea to implement intrinsic motivation schemes for robots to explore their environments actively [10]–[12].

The authors are with Technische Universität Braunschweig, Institut für Robotik und Prozessinformatik, 38106 Braunschweig, Germany {rrayyes,hdo,jsteil}@rob.cs.tu-bs.de

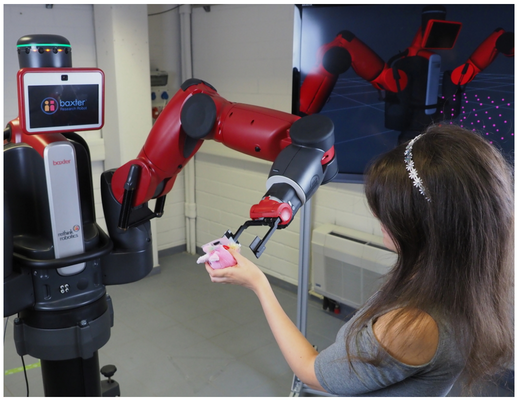


Fig. 1. Our proposed scheme permits efficient learning of IK models which can be applied for example to enable reaching tasks with a Baxter robot

The main challenge of these methods is efficiency, i.e., the number of required training samples. Most online learning methods in robotics tend to require a high number of training samples, which increases exponentially with the number of DoF [3], [13], while sampling with real robot applications is costly and intractable in terms of time, tear and wear. Therefore, the majority of the related works on intrinsic motivation have been demonstrated only in simulation as a proof of concept and only a few works in real robot experiments (e.g., [14], [15]). In this paper, we focus on tackling the online data-driven learning challenges, i.e., efficiency, stability, and entirely online without any intermediate offline methods. Therefore, we propose a novel online interest-driven exploration scheme which is updated continuously with Online Episodic Mental Replay (OEMR) method in order to speed up the learning process and drastically reduce the number of required sample. We apply our scheme on a physical 7-DoF manipulator (Baxter robot by Rethink Robotics cf. Fig. 1).

Most of the recent intrinsic motivation methods are competence-based approaches [8], [10]–[12], [16], [17], which derives the intrinsic motivation signal based on the robot's learning progress. In contrast to the previously proposed competence-based schemes which rely on storing the complete data set, which is not feasible through lifelong learning, we propose an interest measurement which is updated on the fly based on the current robot performance without the need of storing complete data sets. Moreover, our scheme utilizes only the interest measurement to intrinsically drive the system in contrast to other schemes, e.g., [12], which is combined with random goal selection.

Our scheme consists of a high-level goal selection utilizing interest measurement and a low-level exploration, which

978-1-7281-7395-5/20/$31.00 ©2020 IEEE

relies on Goal Babbling (GB) [18] (cf. Fig. 2) for direct inverse model learning through exploration. GB has been proposed as a promising goal-directed method for online bootstrapping sensorimotor skills. It is inspired by infants' motor learning skills "Learning while Behaving" [19], e.g., learning how to reach by trying to reach and improving the acquired skills by iterating the trials. Its flexibility and adaptability have been verified (see [8], [12], [20]–[23]).

For redundant high DoF robots, GB by design learns only one preferred redundancy resolution [18]. However, humans demonstrate more flexibility to solve the required tasks with different redundant solutions. To gain more flexibility and versatility in solving the required tasks, it is preferable to learn multiple solutions. Associative dynamic networks [24], [25] have been proposed to tackle this challenge and provide a suitable representation for multiple redundant solutions. However, these approaches work only offline, which requires the full data set to be stored. Besides, the network complexity (e.g., a hidden layer size) needs to be set in advance without any online adaptation possibility, while any change (e.g., new tasks or environment changes) will require to recollect the data and retrain the network from scratch.

We devise a fully online associative network to learn multiple models through exploration. The different solutions are biased by different default "home postures" to start the exploration from. The complexity of the network is autonomously adapted to the learning problem. It turns out that the main challenge for online dynamic associative network is stability. In real applications, the exploration while behaving produces very noisy data, which makes online incremental associative dynamics potentially unstable. To mediate this effect, we propose a parameter-sharing technique that assures stability by combining incremental regression with associative dynamics to leverage both advantages: stability, accuracy, and multi-model representations. This increases efficiency and avoids exhaustive hyperparameter tuning.

To summarize our contributions in this paper: We devise a fully online, efficient, applicable, flexible, and stable hierarchical learning scheme where it combines four new methods:

- A new intrinsic motivation signal based on the interest measurement to enable efficient online exploration.
- An online incremental associative radial basis function network (OARBF), which is the first online associative dynamic network to be constructed totally from scratch with high stability.
- A new online episodic mental replay (OEMR) to accelerate the learning process.
- A parameter-sharing technique which increases the efficiency and stabilizes the full learning scheme in the presence of highly noisy data.

The video illustrating the work ia available in [26].

II. Hierarchical Interest-Driven Associative Goal Babbling

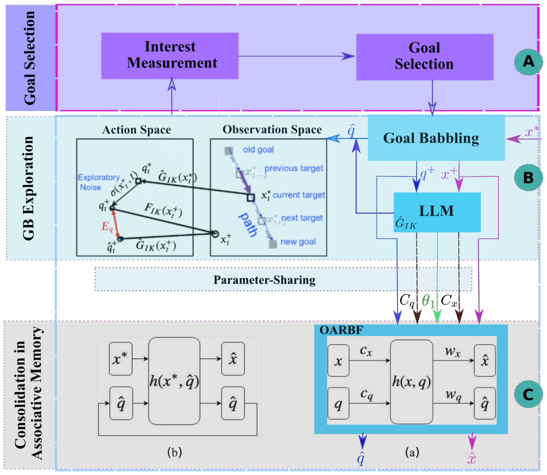


Fig. 2. Hierarchical interest-driven associative GB scheme. (A) Goal selection mechanism, (B) GB exploration scheme, (C): Associative memory (a) OARBF training, (b) establish a feedback for OARBF exploitation.

Fig. 2 illustrates our proposed Hierarchical Interest-Driven Associative GB scheme. It consists of a high-level layer: goal selection (A) and two low-level layers: exploration (B) and an associative memory to consolidate multiple solutions (C). We demonstrate our scheme for learning kinematics, where the task is to learn how to reach some desired spatial positions (goals). In the following, we will explain in detail each component of the scheme. Note that each component is rather a general independent algorithm and can be implemented in other learning schemes.

A. Interest measurement and goal selection

The interest measurement determines which goal the robot will try to attain. At the beginning, all goals are interesting as the robot does not have any knowledge about them. The interest measurement is updated continuously on the fly based on the robot's performance. There are two main criteria to determine the interest measurement: the relative error (RE) and the forgetting factor (FF). The RE (cf. Eq. (1)) measures the performance error on each goal relative to the other goals' errors. A high performance error indicates that the attained task/goal has not been learned well yet. The better the performance is, (i.e., the error decreases), the less interesting the goal becomes.

$$RE(g_i) = \frac{E(g_i) - E_{min}}{E_{max} - E_{min}} \tag{1}$$

where $E(g_i)$ is the current performance error on the goal $g_i \in \mathcal{G}$, E_{min} and E_{max} are the current minimum and maximum performance errors over all goals respectively. As the inverse model is updated locally and continuously, the robot performance might get enhanced (the robot benefits from other experiences) or get deteriorated (the robot forgets potentially about the previously learned scenarios). Hence, we propose a forgetting factor (FF) which indicates the goals that the robot starts to forget about. The higher the factor gets, the more interesting the goal becomes. FF can be estimated using the following criteria:

1) Current Progress: The current progress factor $Prog$ is measured based on the robot performance on a goal g_i over a sliding time window with n goal trials as illustrated in Fig. 3. When the error increases over the time window, the goal

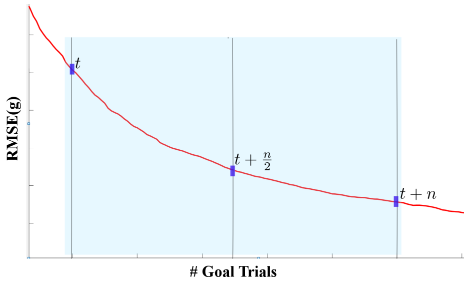


Fig. 3. Performance progress on a g goal through n samples

becomes more interesting as the robot starts to forget about it. $Prog$ is normalized relative to the minimum $Prog_{min}$ and maximum $Prog_{max}$ of all goals' $Prog$ factors (cf. Eq. (2))

$$\left.\begin{aligned} Prog(g_i) &= \frac{\sum_{j=\frac{n}{2}}^{j=n} E_j(g_i) - \sum_{j=1}^{j=\frac{n}{2}} E_j(g_i)}{n} \\ Prog(g_i) &= \frac{Prog(g_i) - Prog_{min}}{Prog_{max} - Prog_{min}} \end{aligned}\right\} \tag{2}$$

This $Prog$ factor is similar to the competence progress [12]. However, $Prog$ considers only when the robot starts to forget, while competence progress focuses on the learning progress whether it enhances or deteriorates Further comparison is illustrated in Sec. IV-A.1

2) General Progress Overview: The second forgetting factor is the general overview of the learning progress. Because the data set is not stored, the general progress is measured by the current performance error on the current goal relative to the minimum ($E_{min}(g_i)$) and the maximum ($E_{max}(g_i)$) performance error of the goal g_i (cf. Eq. 3).

$$REg(g_i) = \frac{E(g_i) - E_{min}(g_i)}{E_{max}(g_i) - E_{min}(g_i)} \tag{3}$$

The initial values $E_{min}(g_i) = inf,\ E_{max}(g_i) = 0,\ \forall g_i \in \mathcal{G}$. FF could be measured either by using one of these factors or as a combination of them (cf. Eq. 4). However, it might be better to forget about poorly performing previous learned experiences and take only $Prog$ factor into consideration ($\lambda = 0$). This assumption is investigated further in our experiments (cf. Sec. IV-A.1). The interest measurement is given by the combination of RE and FF (cf. Eq. 5)

$$FF(g_i) = \gamma Prog(g_i) + (1-\gamma) REg(g_i) \tag{4}$$

$$interest(g_i) = \lambda RE(g_i) + (1-\lambda) FF \tag{5}$$

where $\{\gamma,\ \lambda\} \in [0,\ 1]$ are the weighting parameters. Note that all factors and measurements are normalized in order to have comparable measures for all required goals/tasks.

B. Interest-Driven Goal Babbling

GB is a goal-directed method for learning inverse robot models. It has been proposed for learning inverse kinematics (IK) [18], i.e., learning the required joint configurations $\boldsymbol{q} \in \mathcal{Q} \subset \mathbb{R}^m$ to attain some desired positions $\boldsymbol{x}^* \in \mathcal{P} \subset \mathbb{R}^n$. Where m is the number of DoFs, n is the dimension of the target variable (e.g. $n \in \{2,3\}$ for the spatial position of the end-effector), $\mathcal{Q}$ and $\mathcal{P}$ are the permissible configurations and the corresponding positions respectively. IK maps $\boldsymbol{x}$ from the lower dimensional task space, which represents the observation space, to $\boldsymbol{q}$ in the higher dimensional configuration space, which represents the action space. Hence, GB permits efficient exploration of the task space [27].

The robot starts exploring from its default (home) posture q^{home} corresponding to the starting (home) position x^{home} by trying to reach some predefined targets $g_i \in \mathcal{G} \subseteq \mathcal{P}$ which are selected based on the interest measurement mechanism (cf. Sec. II-A). A linear path of intermediate targets is generated by interpolating between the current target and the next selected one. The robot tries to reach each generated and the selected target, using the local inverse estimate as follows: A correlated exploratory noise σ is added to the estimated output $\hat{\boldsymbol{q}}_t^*$ in order to discover and learn new outcomes ($\boldsymbol{q}_t^+ = \sigma(\boldsymbol{x}, t) + \hat{\boldsymbol{q}}_t^*$). $\boldsymbol{q}_t^+$ is executed and the resulting end effector position $\boldsymbol{x}_t^+$ is observed.

A Local Linear Map (LLM) [18], [20], [28] is used as an incremental regression algorithm to build and update the inverse estimates. Note that any incremental regression technique can be implemented, we choose LLM as it demonstrates a very good accuracy for estimating complex models (e.g., inverse statics [20]). For redundant kinematics, GB tries to select and learn the most efficient solution using the following weighting scheme:

$$\left.\begin{aligned} \boldsymbol{w}_t^{dir} &= \frac{1}{2}(1 + cos\sphericalangle(\boldsymbol{x}_t^* - \boldsymbol{x}_{t-1}^*, \boldsymbol{x}_t^+ - \boldsymbol{x}_{t-1}^+) \\ \boldsymbol{w}_t^{eff} &= \parallel \boldsymbol{x}_t^+ - \boldsymbol{x}_{t-1}^+ \parallel \cdot \parallel \boldsymbol{q}_t^+ - \boldsymbol{q}_{t-1}^+ \parallel^{-1} \\ \boldsymbol{w}_t^{gb} &= \boldsymbol{w}_t^{dir} \cdot \boldsymbol{w}_t^{eff} \end{aligned}\right\} \tag{6}$$

where t is the time step, $\boldsymbol{x}^*$ is the desired position, $\boldsymbol{x}^+$ is the real end-effector position which corresponds to the real configuration $\boldsymbol{q}^+$, $\boldsymbol{w}_t^{dir}$ assess whether the actual movement aligns well with the intended one, and $\boldsymbol{w}_t^{eff}$ measures the efficiency of the actual movement. $(\boldsymbol{x}_t^+, \boldsymbol{q}_t^+, \boldsymbol{w}_t^{gb})$ is used to update the local inverse estimate online in a supervised learning fashion in order to minimize the weighted error E_t^q (cf. Fig. 2, Eq. (7)) between the actual $\boldsymbol{q}_t^+$ and the estimated $\hat{\boldsymbol{q}}_t^+$ configurations as following:

$$\left.\begin{aligned} E_t^q &= \boldsymbol{w}_t^{gb} \|\boldsymbol{q}_t^+ - \hat{\boldsymbol{q}}_t^+\|^2 \\ \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t - \eta \cdot \frac{\partial E_t^q}{\partial \boldsymbol{\theta}_t} \end{aligned}\right\} \tag{7}$$

where θ are the LLM parameters and η is the learning rate. $\boldsymbol{q}^{home}$ is used with a weighting scheme in order to control which solution will be learned. For example, if the robot starts exploring with an elbow-down home posture, the samples with an elbow-down configuration will receive higher weights than the samples with an elbow-up configuration and vice-versa. $\boldsymbol{x}^{home}$ has been used as a resting position in the original GB with probability $\rho \ll 1$ in order to avoid drifting [18]. In our proposed scheme, $\boldsymbol{x}^{home}$ is considered as one of the predefined targets, where the robot is able to autonomously return to it due to the forgetting factor.

C. Incremental Online Associative Radial Basis Function

In order to gain more flexibility and learn several solutions with GB, we propose an Online Associative Radial Basis

Function (OARBF) network which is constructed incrementally from scratch and its network complexity (e.g., hidden layer size) is adapted to the learned problem autonomously. As illustrated in Fig. 2(a), OARBF consists of three layers: input, output and hidden layer. The input and output layers are usually identical with the same number of neurons. The input $\boldsymbol{inp}$ and output $\boldsymbol{out}$ data for learning kinematics are vectors that concatenate the end effector positions $\boldsymbol{x} \in \mathcal{P} \subset \mathbb{R}^n$ and configurations $\boldsymbol{q} \in \mathcal{Q} \subset \mathbb{R}^m$ ($\boldsymbol{inp} = \boldsymbol{out} = [\boldsymbol{x}, \boldsymbol{q}]^T \in \mathbb{R}^{n+m}$). Hence, the data should be normalized in order to have equivalent contributions. The data here is normalized to $[-1,\ 1]$, given the joint limits and the task dimension. The neurons in the hidden layer are added incrementally based on the online data stream. $\boldsymbol{c}_x$ and $\boldsymbol{c}_q$ determine the RBF centers which are added incrementally using an online clustering algorithm(cf. Sec. II-D.2). $\boldsymbol{w}_x$ and $\boldsymbol{w}_q$ are the output weights updated at each step by performing a gradient descent in order to minimize the weighted error

$$\left.\begin{aligned} E_t &= \boldsymbol{w}_t^{gb} \| \boldsymbol{inp}_t - \hat{\boldsymbol{out}}_t \|^2 \\ \boldsymbol{w}_{t+1}^{out} &= \boldsymbol{w}_t^{out} - \eta \cdot \frac{\partial E_t}{\partial \boldsymbol{w}_t^{out}} \end{aligned}\right\} \tag{8}$$

where η is a learning rate, t is a time step, $\boldsymbol{w}^{out} = [\boldsymbol{w}_x, \boldsymbol{w}_q]^T$, $\boldsymbol{w}_t^{gb}$ is the weight of the data sample given in Eq. (6) and $\hat{\boldsymbol{out}}$ is the estimated output of OARBF. The association setup solves the redundancy resolution by utilizing different hidden layer's activations, i.e., the data pairs ($\boldsymbol{x}_i$,$\boldsymbol{q}_i$), ($\boldsymbol{x}_j$,$\boldsymbol{q}_j$) where $\boldsymbol{x}_i = \boldsymbol{x}_j$, $\boldsymbol{q}_i \neq \boldsymbol{q}_j$, $i \neq j$ have $h(\boldsymbol{x}_i, \boldsymbol{q}_i) \neq h(\boldsymbol{x}_j, \boldsymbol{q}_j)$ [24]:

$$\left.\begin{aligned} h_t(\boldsymbol{x}, \boldsymbol{q}) &= \frac{f_t(\boldsymbol{x},\ \boldsymbol{q})}{\sum_{i=1}^{H} f_i(\boldsymbol{x},\ \boldsymbol{q})} \\ f_t(\boldsymbol{x},\ \boldsymbol{q}) &= exp(-\beta_x d(\boldsymbol{x}, \boldsymbol{c}_x)^2 - \beta_q d(\boldsymbol{q}, \boldsymbol{c}_q)^2) \end{aligned}\right\} \tag{9}$$

where β is used to control the spread and the overlap of RBFs $\left(\beta_q = \frac{n}{m}\beta_x\right)$, d is the Euclidean distance between the newly received sample and all existent RBF centers, and H is the number of the hidden neurons. The outputs are estimated based on the hidden layer's activation as follows

$$\hat{\boldsymbol{out}}(\boldsymbol{x}, \boldsymbol{q}) = \boldsymbol{w}^{out} h(\boldsymbol{x}, \boldsymbol{q}) \tag{10}$$

Note that the equations Eq. (9) and Eq. (10) are similar to [24]. However, the output weights are initialized and updated differently. Furthermore, OARBF is constructed incrementally online and from scratch. IK as well as forward kinematics (FK) are learned simultaneously utilizing OARBF. Similar to [24], an output feedback-driven loop is established (cf. Fig. 2(b)) to query the learned model. The network converges to one of the learned solutions based on the previous state of the network.

D. Learners setup and parameter-sharing

We propose a parameter-sharing technique in order to increase the learning efficiency, avoid exhaustive hyperparameter tuning, stabilize and speed up the learning process. LLM has demonstrated high stability and high accuracy [18], [20], while OARBF can model and represent multiple solutions using multi-stable dynamic attractors. Both learners are constructed online, incrementally and from scratch. Hence, online clustering of the input data is needed for both to add the prototypes of LLM and the neurons of OARBF. The main differences are: First, the basis functions in an LLM are local linear functions centered around the prototypes, in contrast to the OARBF where activation functions are radial basis functions represented by hidden neurons. (cf. Eq. (9)). Second, the input and output dimensions of the LLM and OARBF. For LLM the input for learning IK is $\boldsymbol{x} \in \mathbb{R}^n$ and the output is $\boldsymbol{q} \in \mathbb{R}^m$. While OARBF input and output data is defined as $[\boldsymbol{x}, \boldsymbol{q}]^T \in \mathbb{R}^{n+m}$.

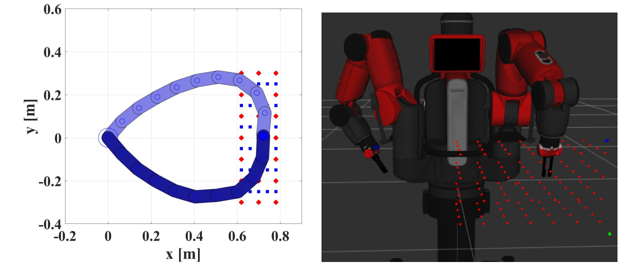
Therefore, the online clustering is done only for the LLM in low dimensional Cartesian space. Consequently, when the prototypes of LLM are added, the neurons of OARBF are added instantaneously. This accelerates the clustering and stabilizes the full learning system as it yields better homogeneous distribution of the prototypes. Moreover, only one parameter set $\theta = \{\eta, r\}$ is shared with both learners, where η is the learning rate and r is the radius which determiners the of the vicinity of each basis function. $\beta_q = \frac{n}{m}\beta_x$ (cf. Eq. (9)). Thus we have to tune only 3 parameters for both learners, and only one clustering which increases the efficiency factor up to 4.

1) Learner initialization: The LLM is initialized with the first linear function $\hat{\boldsymbol{G}}^{(1)}(\boldsymbol{x})$ centered around a prototype vector ${\boldsymbol{q_p}}^{(1)} = \boldsymbol{x}^{home}$ corresponding to the home posture $\boldsymbol{q}^{home}$ [18] [20]. OARBF is initialized with the first neuron centered around the first received sample $\{\boldsymbol{c}_{x_1} = \boldsymbol{x}^{home}, \boldsymbol{c}_{q_1} = \boldsymbol{q}^{home}\}$. The output weights are initialized with the inputs weights, i.e., $\{\boldsymbol{w}_{x_1} = \boldsymbol{c}_{q_1}, \boldsymbol{w}_{x_1} = \boldsymbol{c}_{q_1}\}$ in order to shift the output of the network to the current initial sample.

2) Online clustering for the learners: When LLM receives a new sample with a distance of at least a radius r to all existent prototypes, a new local linear function $\hat{\boldsymbol{G}}^{(i+1)}(\boldsymbol{x})$ will be added and centered around the newly received sample ${\boldsymbol{q_p}}^{(i+1)} = \boldsymbol{x}_{new}$. $\hat{\boldsymbol{G}}^{(i+1)}(\boldsymbol{x})$ is initialized with the last inverse estimation before adding the new function in order to avoid abrupt changes in the inverse estimate function, i.e., the insertion of the new function will not change the local behavior of $\hat{\boldsymbol{G}}(\boldsymbol{x})$ at $\boldsymbol{x}_{new}$. LLM parameters are initialized as in [18], [20]. Accordingly, a new neuron will be added to OARBF, initialized with the output weights of its closest neighbor to avoid drastic changes in the learned function and centered around the newly received sample.

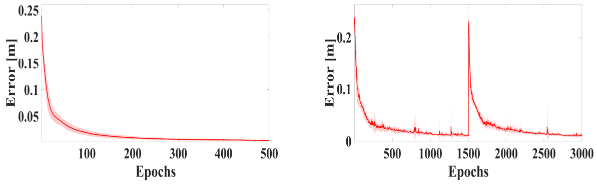
III. Online episodic mental replay

For rapid online adaptation and to accelerate the learning process, we propose an online episodic mental replay (OEMR). Mental replay mechanism has been proved to be an essential component in human learning process [29]. Several replay mechanisms have been proposed for artificial agents to speed up the learning process, e.g., Experience Replay [30], [31] which stores the data set and randomly samples again from it with mini batch learning. Imaginary Experience



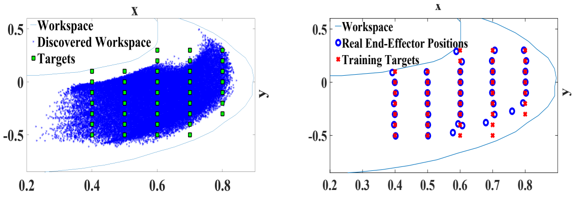
(a) 10R reaching the same position with different configurations (b) Virtual target grid in visualized in rviz

Fig. 4. Robots Setup



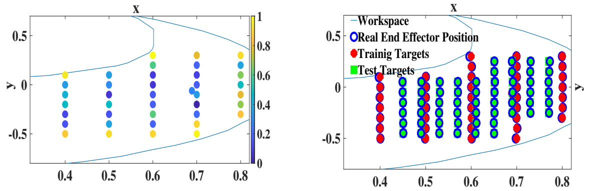
(a) GB LLM (b) AGB - OARBF

Fig. 5. Interest-Driven std RMSE



(a) Competence-Based GB (b) Original GB

Fig. 6. Original GB and Competence-based GB performance



(a) Interest Measurement (b) Performance Results

Fig. 7. Interest-Driven GB

Replay [32] and Hindsight Experience Replay [33], [34] use sample augmentation in order to sample imaginary goals which needs the full goal space to be known in advance. Mental Replay [15] samples several trajectories where each is replayed once to intensify each experience situation. In contrast to these approaches, OEMR doesn't require storing several episodes nor producing additional imaginary samples. The samples of each last epoch, which consists of M trials of some selected goals, are stored in the replay buffer. At the end of each epoch, these samples are replayed online without an execution on the robot in order to intensify previous experiences. The online replay mechanism accelerates the convergence of the learner as the model is built incrementally and updated continuously, where only one gradient descent step is done for each sample (cf. Eq. (7), Eq. (8).

IV. Experimental Results

We first implement our proposed scheme in an illustrative 10 R planar manipulator experiment (cf. Fig. 4) in order to demonstrate the efficiency as well as the advantages gained by our proposed scheme and compare it to the state-of-art [12], [18]. The parameters are obtained with pattern search [35]. Then, we demonstrate our proposed scheme with a 7 DoF physical robot manipulator (cf. Fig. 1).

A. 10 R planar manipulator Experiment without Replay

1) Hierarchal Interest-Driven Goal Babbling: Our proposed Hierarchal Interest-Driven GB (cf. Sec. II-A, Sec. II-B), the original GB [18] and GB with Competence measurement [12] have been implemented for a 10R planar manipulator shown in Fig. 4(a), each link length is 10 cm. The task is to achieve multiple predefined targets. Each experiment was repeated 20 times with 500 epochs, each epoch consists of 100 samples collected at each time step.

Table I illustrates the average validation and test performance root mean squared error (RMSE) as well as the average standard deviation (std) of RMSE. Although the avg. RMSE is reasonable for all GB schemes, the original GB, as well as the GB with competence measurement, do not guarantee to achieve all the required targets, especially the targets which are difficult to be reached (e.g., the targets located near to the workspace border) as illustrated in Fig. 6. The reason is, on the one hand, the original GB relies on a random goal selection. Consequently, all targets receive similar attention by the robot, while in our proposed scheme, the robot focuses on hard-to-reach and untrained targets, and it recovers from forgetting previously learned targets due to the application of FF. On the other hand, the exploration in [12] relies on the combination of a random goal selection and the competence measurement. The competence measurement gives high interest to the area of the highest progress, whether the robot is learning (RMSE decreases) or the robot forgets (RMSE increases). While in our learning system is completely driven by its interest, concentrating on tasks that have either been forgotten or not well learned. In addition, the exploration with competence measurement originally relies on the nearest neighbor strategy [8], [12], [16], which is not applicable in our proposed methods as the models are updated online and no data set is stored.

Table I also illustrates the high stability of the Interest-Driven GB with the minimum std RMSE 0.8 mm (the shaded area in Fig. 5(a)) which surpasses the other methods and shows a robust performance over all experiments. All training and test targets are always reached with our proposed interest measurement with high accuracy as illustrated in Fig. 7(b), where the red points represent the training targets, the green ones represent the test targets, and the blue circles are the observed end-effector positions. Table I also shows that $Prog$ factor yields higher stability than Reg, as expected when it is desirable to forget about the poorly performing previous experiences (cf. Sec.II-A). The validation RMSE has been computed at each epoch to test the robot's performance on all training targets. The validation RMSE converges after 35 epochs as shown in Fig. 5(a). Fig. 7(a) shows the interest

TABLE I

10R Experimental Results Comparison

Goal Babbling	Goal Selection	avg. Validation RMSE $[m]$	avg. Test RMSE $[m]$	avg. RMSE std $[m]$
Hierarchal GB 10R Experimental Results				
Original	random selection	$7.4\cdot10^{-3}$	$3.6\cdot10^{-3}$	$5\cdot10^{-3}$
Interest-Based	$RE, Prog$	$2.7\cdot10^{-3}$	$1.9\cdot10^{-3}$	$0.8\cdot10^{-3}$
	RE, REg	$4.7\cdot10^{-3}$	$2\cdot10^{-3}$	$1.5\cdot10^{-3}$
	$RE, Prog, REg$	$4.3\cdot10^{-3}$	$2.3\cdot10^{-3}$	$1.5\cdot10^{-3}$
Competence		$5.5\cdot10^{-3}$	$2.3\cdot10^{-3}$	$1.5\cdot10^{-3}$
Hierarchical Interest-Driven AGB vs AGB				
AGB	random selection	$10\cdot10^{-3}$	$9\cdot10^{-3}$	$30\cdot10^{-3}$
AGB	Interest-Based	$9.7\cdot10^{-3}$	$8\cdot10^{-3}$	$2.5\cdot10^{-3}$

measurements over all epochs. The "yellow" targets represent the most interesting targets for the robot. They represent the hard-to-reach targets, e.g., targets near the workspace border. The targets in dark blue indicate that the robot barely tries to attain them as the learned model benefits from other experiences to generalize well.

2) Hierarchal Interest-Driven Associative Goal Babbling: Fig. 4(a) shows the experimental setup. First, the robot starts exploring with a curvature-up home posture utilizing our proposed scheme, trying to reach some predefined targets. After N epochs, the robot moves to its new home posture, which is in this setup a curvature-down configuration and continues exploring trying to reach the same target set. LLM and OARBF are updated at each step. The targets are chosen iteratively based on the interest measurement. 66 neurons were added incrementally to OARBF. The robot managed to reach all test and training targets based on the initial robot state (curvature-up or down) without inconsistencies with avg. RMSE. 9 mm. Fig. 5(b) shows although it is a highly dynamic and very noisy system, the network demonstrates high stability due to the parameter-sharing technique. Table I shows the comparison between the associative GB (AGB) where the goals are chosen randomly and the hierarchical interest-driven AGB. The later one demonstrates very robust performance illustrated with minimum std RMSE of 2.5 mm.

A similar experiment has been done in [24], where ARBF is trained offline with a pre-fixed network size of 300 neurons to achieve similar accuracy without the possibility of any online adaptation. The exploration in [24] has been done in two different phases. The full data sets were stored to be consolidated in ARBF offline. In contrast, our scheme is fully updated online on a fly with simultaneous exploration and consolidation, and the network size is tailored to the problem. Besides, two learners need to be tuned and an additional clustering phase is required in [24], in contrast to the saved efforts in our system due to the parameter-sharing techniques.

B. 7 DoF physical robot manipulator (Baxter) with Replay

To demonstrate the applicability of our approach on a real robot, we utilized the left 7-DoF arm of a Baxter Robot by Rethink Robotics (cf. Fig. 1). We chose Baxter because of its inaccuracy with a precision of 5 mm [36], which is challenging for the learners to cope up with more noisy data. Baxter sampling rate using MoveIt - Motion Planning Framework [37] is 3 sec in our experiments. The parameter set is $\{\eta = 0.0725, \sigma = 0.0452, r = 0.0869 \beta_x = 5\}$

1) Hierarchical Interest-Driven Goal Babbling: In the initialization phase of the experiment, the experimenter shows the robot where to explore by simply moving its arm to define the desired workspace. A three-dimensional virtual grid of targets is generated in the detected workspace illustrated in a 3D visualizer (Rviz provided by ROS) as shown in Fig. 4(b) while the experiment is conducted with the physical robot. The targets are scattered in a cuboid shape with a vertical and horizontal distance of 10 cm between them.

In the training phase, the robot started exploring from its home posture, trying to reach the targets based on the interest measurement. Each target trial consists of N intermediate samples, which varies depending on the distance between the targets. Each training episode consists of 1000 samples, including all intermediate samples. OEMR is performed after each epoch, which took 10 seconds. Only 4 epochs were needed to learn to reach 41 targets with 6.7 mm training RMSE. After 3 $hours$ and 20 min training phase with a sampling rate of 3 sec per sample, the robot tried to reach 93 new targets randomly scattered in the explored workspace. All the targets were reached with an RMSE of 7.8 mm, which is acceptable accuracy considering the low positional accuracy of the Baxter robot of ± 5 mm.

2) Hierarchical Interest-Driven Associative Goal Babbling: To test our hierarchical interest-driven AGB with Baxter, the exploration was done as described in IV-A.2 with two different home postures (angles in radian): $\boldsymbol{q}^{home1} = [-0.17, -0.25, -0.12, 0.93, -0.71, 1.72, 0.61]^T$, $\boldsymbol{q}^{home2} = [-1.20, -0.615, 0.38, 1.34, 0.29, 1.28, -0.329]^T$. The training phase is 5 episodes for each exploration phase with OEMR. 30 virtual Targets were generated in the training phase. The robot performance is evaluated on 27 new targets with two different initial starting configurations. The robot manages to reach all the targets without any inconsistencies, i.e., without switching solutions or averaging between them for the OARBF. 22 targets were reached with a test RMSE of 6.4 mm, and only 5 targets with an avg. RMSE of 8.6 mm as they were difficult to reach because of self-collision avoidance by the robot system.

V. Conclusion

We proposed four general methods that are integrated into a novel hierarchal online interest-driven learning scheme. It guarantees to accomplish all required tasks, accelerates the exploration and learning procedures by drastically reducing the number of required samples, and demonstrates robust performance with a smaller RMSE standard deviation compared to the current state of the art. The system is able to learn multiple solutions for solving required tasks flexibly by utilizing our proposed OARBF, which is the first associative dynamic network to be constructed incrementally with high stability. We have demonstrated the applicability of our scheme with direct online training on a real 7 DoF physical humanoid arm. The robot learned online from scratch and accomplished the required task with a reasonable number of samples and sufficient accuracy.

References

[1] D. Wolpert and M. Kawato, "Multiple paired forward and inverse models for motor control," *Neural Networks*, vol. 11, no. 7-8, pp. 1317–1329, Oct. 1998. [Online]. Available: https://doi.org/10.1016/s0893-6080(98)00066-5

[2] M. Rolf and J. Steil, "Efficient exploratory learning of inverse kinematics on a bionic elephant trunk," vol. 25, no. 6, 2014, pp. 1147–1160.

[3] D. Nguyen-Tuong and J. Peters, "Model learning for robot control: a survey," *Cognitive Processing*, vol. 12, no. 4, pp. 319–340, Apr. 2011.

[4] M. Asada, K. F. MacDorman, H. Ishiguro, and Y. Kuniyoshi, "Cognitive developmental robotics as a new paradigm for the design of humanoid robots," *Robotics and Autonomous Systems*, vol. 37, no. 2-3, pp. 185–193, Nov. 2001. [Online]. Available: https://doi.org/10.1016/s0921-8890(01)00157-9

[5] J. Schmidhuber, "Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts," *Connection Science*, vol. 18, no. 2, pp. 173–187, 2006.

[6] A. Cangelosi, M. Schlesinger, and L. B. Smith, *Developmental robotics: From babies to robots*. MIT Press, 2015.

[7] E. Ugur, Y. Nagai, E. Sahin, and E. Oztop, "Staged development of robot skills: Behavior formation, affordance learning and imitation with motionese," *IEEE Transactions on Autonomous Mental Development*, vol. 7, no. 2, pp. 119–139, June 2015.

[8] S. Forestier and P. Oudeyer, "Modular active curiosity-driven discovery of tool use," in *IEEE/RSJ, IROS2016*, pp. 3965–3972.

[9] M. Rolf, J. J. Steil, and M. Gienger, "Learning flexible full body kinematics for humanoid tool use," in *2010 International Conference on Emerging Security Technologies*. IEEE, 2010, pp. 171–176.

[10] J. Schmidhuber, "Formal theory of creativity, fun, and intrinsic motivation (1990 - 2010)," *IEEE Trans. on Auton. Ment. Dev.*, vol. 2, no. 3, pp. 230–247, Sep. 2010.

[11] V. G. Santucci, G. Baldassarre, and M. Mirolli, "Grail: A goal-discovering robotic architecture for intrinsically-motivated learning," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 8, no. 3, pp. 214–231, Sep. 2016.

[12] A. Baranes and P. Oudeyer, "Active learning of inverse models with intrinsically motivated goal exploration in robots," *Robot. Auton. Syst.*, vol. 61, no. 1, pp. 49–73, 2013.

[13] D. Kubus, R. Rayyes, and J. J. Steil, "Learning forward and inverse kinematics maps efficiently," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Oct. 2018. [Online]. Available: https://doi.org/10.1109/iros.2018.8593833

[14] P. Oudeyer, F. Kaplan, and V. V. Hafner, "Intrinsic motivation systems for autonomous mental development," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 2, pp. 265–286, April 2007.

[15] D. Tanneberg, J. Peters, and E. Rueckert, "Intrinsic motivation and mental replay enable efficient online adaptation in stochastic recurrent networks," *CoRR*, vol. abs/1802.08013, 2018.

[16] S. M. Nguyen and P. Oudeyer, "Socially guided intrinsic motivation for robot learning of motor skills," *Autonomous Robots*, vol. abs/1804.07269, 2014.

[17] V. Santucci, G. Baldassarre, and M. Mirolli, "Which is the best intrinsic motivation signal for learning multiple skills?" *Frontiers in Neurorobotics*, vol. 7, p. 22, 2013.

[18] P. O. Stalph and M. V. Butz, "Learning local linear jacobians for flexible and adaptive robot arm control," *Genetic Programming and Evolvable Machines*, vol. 13, no. 2, pp. 137–157, 2012.

[19] C. von Hofsten, "An action perspective on motor development," *Trends in CogSci*, vol. 8, pp. 266–272, 2004.

[20] R. Rayyes, D. Kubus, and J. Steil, "Learning inverse statics models efficiently with symmetry-based exploration," *Frontiers in Neurorobotics*, vol. 12, p. 68, 2018.

[21] P. Loviken, N. Hemion, A. Laflaquiere, M. Spranger, and A. Cangelosi, "Online learning of body orientation control on a humanoid robot using finite element goal babbling," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Oct. 2018. [Online]. Available: https://doi.org/10.1109/iros.2018.8593762

[22] A. K. Philippsen, R. F. Reinhart, and B. Wrede, "Goal babbling of acoustic-articulatory models with adaptive exploration noise," in *2016 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. IEEE, Sep. 2016. [Online]. Available: https://doi.org/10.1109/devlrn.2016.7846793

[23] R. F. Reinhart, "Autonomous exploration of motor skills by skill babbling," *Auton. Robots*, vol. 41, no. 7, pp. 1521–1537, 2017.

[24] R. Reinhart and M. Rolf, "Learning versatile sensorimotor coordination with goal babbling and neural associative dynamics," in *IEEE ICDL*, Aug 2013, pp. 1–7.

[25] R. F. Reinhart and J. J. Steil, "Learning whole upper body control with dynamic redundancy resolution in coupled associative radial basis function networks," in *IROS*. IEEE, 2012, pp. 1487–1492.

[26] R. Rayyes, H. Donat, and J. Steil, "Hierarchical interest-driven (associative) goal babbling for efficient exploration video." [Online]. Available: https://www.rob.cs.tu-bs.de/node/809

[27] M. Rolf, J. J. Steil, and M. Gienger, "Goal babbling permits direct learning of inverse kinematics." *IEEE Trans. Autonomous Mental Development*, vol. 2, no. 3, pp. 216–229, 2010.

[28] H. Ritter, "Learning with the Self-Organizing Map," in *ICANN-91*, T. Kohonen, Ed., vol. 1. North Holland, 1991, pp. 379–384.

[29] D. Foster and M. Wilson, "Reverse replay of behavioural sequences in hippocampal place cells during the awake state," *Nature*, vol. 440, no. 7084, pp. 680–683, 3 2006.

[30] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[31] L.-J. Lin, *Reinforcement learning for robots using neural networks*. Technical report, DTIC Document, 1993.

[32] A. Gerken and M. Spranger, "Continuous value iteration (cvi) reinforcement learning and imaginary experience replay (ier) for learning multi-goal, continuous action and state space controllers," *IEEE ICRA*, 2019.

[33] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, P. Abbeel, and W. Zaremba, "Hindsight experience replay," *Advances in Neural Information Processing Systems 30*, pp. 5048–5058, 2017.

[34] M. Riedmiller, R. Hafner, T. Lampe, M. Neunert, J. Degrave, T. V. de Wiele, V. Mnih, N. Heess, and J. T. Springenberg, "Learning by playing solving sparse reward tasks from scratch," in *Proceedings of the 35th International Conference on Machine Learning*, vol. 80. PMLR, 10–15 Jul 2018, pp. 4344–4353.

[35] R. Lewis and V. Torczon, "Pattern search algorithms for bound constrained minimization," *SIAM Journal on Optimization*, vol. 9, no. 4, pp. 1082–1099, 1999.

[36] "Rethink robotics baxter - hardware specifications," http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications, accessed: 2019-09-12.

[37] "Moveit - motion planning framework," https://moveit.ros.org/, accessed: 2019-09-14.