Impedance Primitive-augmented Hierarchical Reinforcement Learning for Sequential Tasks

Amin Berjaoui Tahmaz[†], Ravi Prakash[‡], Jens Kober[†]

***Abstract*—This paper presents an Impedance Primitive-augmented hierarchical reinforcement learning framework for efficient robotic manipulation in sequential contact tasks. We leverage this hierarchical structure to sequentially execute behavior primitives with variable stiffness control capabilities for contact tasks. Our proposed approach relies on three key components: an action space enabling variable stiffness control, an adaptive stiffness controller for dynamic stiffness adjustments during primitive execution, and affordance coupling for efficient exploration while encouraging compliance. Through comprehensive training and evaluation, our framework learns efficient stiffness control capabilities and demonstrates improvements in learning efficiency, compositionality in primitive selection, and success rates compared to the state-of-the-art. The training environments include block lifting, door opening, object pushing, and surface cleaning. Real world evaluations further confirm the framework's sim2real capability. This work lays the foundation for more adaptive and versatile robotic manipulation systems, with potential applications in more complex contact-based tasks.**

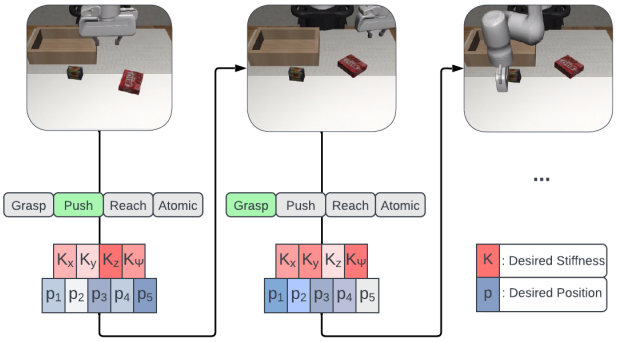

Fig. 1: Figure shows the augmentation of the impedance primitive into HRL policy.

I. Introduction

Realistic manipulation tasks involve a prolonged sequence of motor skills in varying environments. For decades, the challenge of enabling robotic manipulators to solve realistic long-horizon tasks has persisted. While existing research has made strides in addressing important aspects of long-horizon tasks, a critical gap remains in the context of contact-rich environments, highlighting a crucial area that requires further exploration and development. An example can be found in a common manipulation task: object sorting. A robot should be able to plan a series of precise actions over time while adjusting its positioning and applied forces to accommodate objects of varying shapes and sizes while also taking the interaction environment into consideration. This paper focuses on the intersection of deep reinforcement learning (DRL) and adaptive stiffness control to address this longstanding challenge.

Prior works have extensively explored robotic manipulation in long-horizon applications. Conventional methods often use state machines [1][2] or symbolic reasoning [3][4] to learn action sequences for solving a task. However, these approaches explicitly design the decision-making sequence, which may introduce constraints that limit adaptability to different tasks and contribute to error accumulation throughout the task sequence. In response to these limitations, learning techniques such as hierarchical reinforcement learning (HRL) [5] have been employed, establishing themselves as a common approach for problems requiring sequential decision-making.

When deploying long-horizon frameworks in contact-rich environments, the integration of stiffness control becomes crucial for adapting to external forces and uncertainties during task execution. This adaptability ensures precision and stability in navigating contact-rich environments. However, despite a substantial body of research dedicated to variable stiffness control, current approaches are primarily tailored to short-horizon applications. These methods typically involve designing controllers that adjust end-point force in response to environmental forces [6], adapting impedance and damping parameters through learning techniques [7][8], and learning from a human demonstrator [9][10].

This paper aims to bridge the gap between sequential task planning and adaptive stiffness control using a DRL framework. We design an HRL framework, as shown in Figure (1), that selects a high-level action primitive from a pre-defined library and outputs an initial estimate for controller parameters for low-level control. During primitive execution, an adaptive controller is initiated to optimize the robot's stiffness, aiming for an balance between safety (reducing interaction forces with the environment) and performance (ensuring task completion). This design allows the robot to dynamically optimize stiffness parameters, enabling it to transition between high stiffness for precision tasks and increased compliance for enhanced adaptability. We present experiments conducted in both simulation and the real world, focusing on sequential tasks that deal with different contact challenges. Our results highlight notable advantages when compared to a state-of-the-art baseline.

Authors [†] are with TU Delft, Netherlands. Authors [‡] are with IISc Bangalore, India. Email id : amine.berjawi123@gmail.com, ravipr@iisc.ac.in, j.kober@tudelft.nl (in order).

II. Related Works

Sequential Planning: Extensive work in task and motion planning (TAMP) spans various robotics applications, involving explicit decision-making frameworks and machine learning for learned behavior sequences. Common approaches employ hierarchical task planning, combining high-level planners with low-level controllers. In robot manipulation, this often takes the form of finite state machines [1], [11], [12] or behavior trees [13], [14] as high-level controllers. Similar methods use symbolic reasoning [15], [16], [17], representing high-level tasks and constraints with symbols. Although these methods offer explainability, their pre-defined nature limits adaptability to real-world variability, leading to suboptimal performance. Our proposed framework addresses this by learning the high-level planner and optimizing low-level controller parameters for better generalization and robustness. Recently, learning approaches have emerged to overcome these limitations. Imitation learning (IL) is a key candidate for sequential planning, enabling robots to learn demonstrated behavior sequences. Behavior cloning, a well-established IL method, has robots replicate demonstration sequences [18], [19], [20], but this limits generalizability. Advanced IL methods aim to generalize learned sequences [21], [22], [23], yet they still struggle to adapt to new environments. Our framework adapts action sequences to the environment state, addressing this limitation and mitigating suboptimal performance from human error in demonstration data. Hierarchical Reinforcement Learning (HRL) has gained attention for long-horizon planning. State-of-the-art approaches like MAPLE [24], RAPS [25], and STAP [26] train hierarchical policies to choose and execute primitives from a behavior library. Despite handling complex tasks and improving sample efficiency, these methods rely on static controllers, which hinder performance in contact tasks and pose risks in real-world settings. Our method builds on these concepts, optimizing stiffness to maximize compliance without compromising task success.

Variable Stiffness Control: Existing methods for adapting the stiffness of an impedance controller typically use task-specific impedance profiles. Common approaches include learning from demonstration methods, such as Dynamic Motion Primitives [27], [28], [29] or Gaussian Mixture Models [9], [30]. Alternatively, some methods schedule variable stiffness gains for different task phases [31], [32], [33]. Despite their ease of application, these methods struggle to generalize stiffness profiles across tasks and depend on expert demonstrators. RL has emerged as a promising method for learning stiffness profiles. Some methods bootstrap the RL policy with initial stiffness demonstrations [34], [35], [36] to accelerate learning, which are then optimized for specific tasks. However, the reliance on expert demonstrators remains an issue. Other RL approaches focus on designing an appropriate action space in which an agent samples impedance parameters as actions to adapt controller behavior. For adaptive stiffness applications, an impedance action space allows the agent to learn stiffness and damping parameters in joint space [37] and end-effector space [8]. Similar approaches use residual reinforcement learning, where a policy outputs actions to support an existing controller [1], [38], [39]. However, these methods fail in long-horizon tasks due to their limited ability to capture sequential dependencies.

Contributions: The main contributions of this work are i) an impedance primitive augmented HRL framework for sequential contact tasks, ii) a novel behavior affordance that concurrently optimizes for position and compliance; (iii) an adaptive controller for dynamic stiffness modifications for optimal execution in varying environments.

III. Problem Statement

The long-horizon robotics manipulation task can be formulated within the framework of HRL combined with Parameterized Action Markov Decision Processes (PAMDPs)[40]. Let S represent the state space, and $\pi_H : S \rightarrow A_H$ be the high-level policy that selects high-level actions $a_H \in A_H$, which define primitives. For each high-level action a_H, let $\pi_L^{a_H} : S_L^{a_H} \rightarrow A_L^{a_H} \times \Theta$ be the corresponding low-level policy that selects parameterized actions (a_L, θ). The overall policy $\pi(s) = \pi_L^{\pi_H(s)}(s)$ determines the hierarchical decision-making process. The environment dynamics are captured by the transition function $P(s'|s, \bar{a})$ and reward function $R(s, \bar{a})$, where $\bar{a} = (a_L, \theta)$. The objective is to find the hierarchical policy π that maximizes the expected cumulative reward $J(\pi) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right]$, optimizing both the high-level task decomposition and the execution of parameterized actions for efficient manipulation.

IV. Preliminaries: MAPLE [24]

State-of-the-art framework MAPLE [24] employs a hierarchical policy structure within the framework of HRL and PAMDPs. The policy structure consists of a high-level task policy π_H and a low-level parameter policy $\pi_L^{a_H}$. Both policies receive an observation containing information regarding the state of the environment and the robot, with $\pi_L^{a_H}$ additionally taking in the output of π_H. The high-level policy, implemented as a neural network, selects a primitive based on the observation. In contrast, the low-level policy has a separate neural network for each primitive and aims to predict the parameters for the chosen primitive.

The sequential decision-making problem is formulated as a PAMDP. At each time step, π_H selects and executes a parameterized behavior primitive p_n from a library of primitives $\mathcal{L} = \{p_1, p_2, ..., p_n\}$. Each primitive is characterized by a function $f_n(s, \theta)$ in which s represents the current state of the robot while θ represents the parameters outputted by $\pi_L^{a_H}$. This function initiates a closed-loop control sequence over a finite time horizon, whose length is determined by the number of *atomic actions* needed to execute the selected primitive. These atomic actions are essentially short motions that cannot be further subdivided.

During primitive execution, the control loop aims to minimize the error between the current state, defined by s, and the target state, defined by the parameters θ. For instance,

TABLE I: Description of primitives and their parameters

Primitive	Description	Parameters
Reach	Moves the end-effector to a target location	(x, y, z)
Grasp	Moves end-effector to grasp location then activates gripper	(x, y, z, ψ)
Push	Moves end-effector to a target location, then applies a displacement δ	$(x, y, z, \delta_x, \delta_y, \delta_z)$
Atomic	Apply atomic action	$(\delta_x, \delta_y, \delta_z)$
Gripper	Open/Close binary gripper	g

the agent receives an observation and accordingly selects a grasping primitive. Subsequently, this primitive initiates a closed-loop control sequence to guide the end-effector toward specific coordinates (determined by θ), then closes the gripper. The primitives and their parameters are documented in Table I.

During the learning of the hierarchical policy, MAPLE incorporates *affordances* to improve exploration and incentivize desired behaviors, which is a common practice in the existing literature [41][42][43]. A typical affordance is position-based where executing a primitive around an object of interest leads to higher affordance rewards, promoting exploration in the proximity of relevant objects. When selecting primitive p with parameters θ in a given state s, an affordance value $a(s, \theta; p) \in [0, 1]$ is added to the reward. For instance, executing a grasping primitive yields a higher affordance around graspable objects. This position affordance is modeled as

$$a_{\text{pos}}(s, \theta; p) = \max_{\kappa \in \mathcal{K}} (1 - \tanh(\max(||\theta - \kappa|| - \tau, 0))) \quad (1)$$

where $\mathcal{K}$ represents the set of object keypoints and θ is the chosen parameters for a primitive.

V. IMP-HRL

We propose Impedance Primitive-augmented HRL (IMP-HRL) for robust sequential contact tasks. We introduce two components into the MAPLE framework that allow us to to achieve variable impedance control for sequential contact tasks.

A. Impedance Primitive

To accommodate contact-rich environments, the target states need to be extended from exclusively position-based parameters as in MAPLE to also include variable impedance parameters. We propose augmenting HRL with the primitive parameter action space containing the position and impedance parameters [8]. It allows the agent to control the impedance parameters by sampling them as actions. This augmentation extends the parameter space, θ, to now contain (K_x, K_y, K_z) for variable stiffness/impedance control along different coordinate axes and K_ψ for handling orientation or angular variations (shown in Figure (1)). The damping term D in the impedance parameters are selected based on critical damping of system's closed loop response to reduce the number of learnable parameters.

A limitation of this primitive representation arises from the sequential nature of decision-making: once the policy triggers a behavior primitive, it is required to wait for the primitive to complete its execution before modifying the stiffness value again. On the other hand, using an action space with dynamically adapting stiffness parameters introduces a learning challenge. Therefore, the stiffness parameters predicted by the parameter policy will act as an initial stiffness prediction which will be further adjusted using an adaptive stiffness controller.

Affordance Coupling - Combining Position and Stiffness Affordances: In the context of tasks that can benefit from stiffness control, these position-based affordances (1) are insufficient since they focus exclusively on spatial information. To address this limitation, we propose an additional *stiffness affordances* to maximize compliance whenever possible. In turn, this translates to a reduction in interaction forces between the robot and the environment, which improves the overall safety of the system. Accordingly, stiffness is only increased when it is necessary to meet task requirements. This stiffness affordance is modeled as

$$a_{\text{stiff}}(s, \theta; p) = 1 - \frac{K(s, \theta; p) - K_{\text{min}}}{K_{\text{max}} - K_{\text{min}}} \quad (2)$$

where $K(s, \theta; p)$ is the selected stiffness and $(K_{\text{min}}, K_{\text{max}})$ represent a pre-defined stiffness range in the action space. In practice, a_{stiff} increases linearly as stiffness decreases.

To effectively leverage both position and stiffness affordances, a geometric mean of both affordances is used to balance the two objectives. This approach leads to *affordance coupling*, which makes increments in one affordance have a more pronounced impact when the other affordance is also high. This affordance is visualized in Figure 2 and modeled as

$$a_{\text{combined}}(s, \theta; p) = \sqrt{a_{\text{pos}}(s, \theta; p) \cdot a_{\text{stiff}}(s, \theta; p)} \quad (3)$$

This coupling model improves exploration efficiency and encourages the agent to select low-stiffness parameters dur-

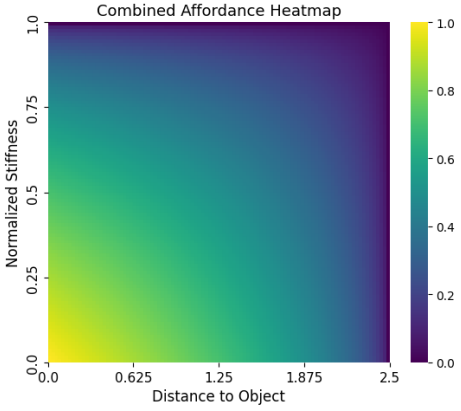

Fig. 2: Heatmap visualization of affordance coupling

ing the early stages of training. Furthermore, this method eliminates the necessity for careful reward weight tuning that is typically required when directly penalizing high stiffness values. Such tuning would otherwise need to be conducted for each new environment, potentially having a detrimental effect on learning performance [44]. Note that the *atomic* and *gripper release* always have an affordance of 1 due to their general utility.

B. Adaptive Controller

After the policy selects a primitive and its parameters, the behavior is executed through a closed loop control scheme. Using the stiffness parameters outputted by the parameter policy as an initial estimate of the required stiffness to complete a given stage of the task, this stiffness is adapted in real-time using an adaptive stiffness controller. Figure (4) shows the adaptive impedance controller integrated within the low level parametrized policy.

The **Adaptive Controller** used in this mimics human muscle stiffness during motion execution [45] by adapting the stiffness in accordance with the output of

$$\dot{K}(t) = \beta|\epsilon(t)| - \gamma E \tag{4}$$

where $\epsilon(t)$ is the closed loop feedback error and E is the energy consumed by the robot joints, while β and γ scale these values to influence the stiffness behavior. As for the corresponding damping matrix, it satisfies a critical damping condition such that $D(t) = 2\sqrt{K(t)}$. It is important to note that interpolation is used to generate intermediate points along the trajectory toward a target state. This ensures that the controller avoids drastically increasing the stiffness whenever a new target state is set.

In practice, the controller uses the stiffness output of the low-level RL policy as an initial value. Then, it calculates the stiffness at the next step by using β to scale the increase in stiffness proportional to the feedback error $e(s-\theta)$. Simultaneously, it reduces stiffness by scaling the current energy consumption E with γ. This process yields a net increase or decrease in the controller's stiffness. An example is visualized in Figure (3) wherein a robot executes an elliptical wiping motion. Further information on the acquisition of β and γ can be found in the Appendix (**??**).

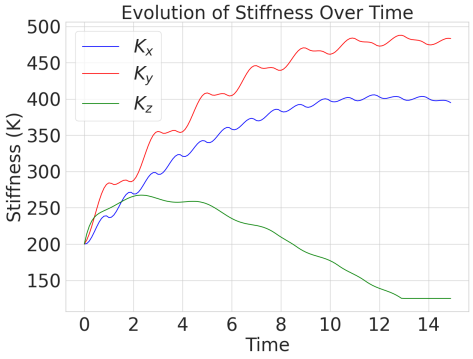

Fig. 3: Example of adaptive stiffness when wiping.

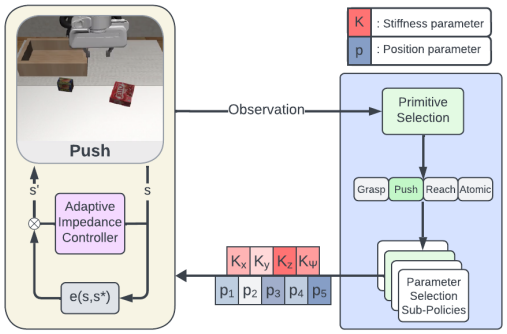

Fig. 4: Adaptive impedance controller integrated within the low-level parametrized policy.

VI. Experimental Results

In the experiments, we investigated the framework's learning efficiency, analyzed its stiffness and force behavior, highlighted patterns in primitive selection, and evaluated its performance in a real-world setting. This section is divided into experimental setup, evaluation in simulation and real robot, and comparative analysis with respect to state-of-the-art method on sequential task execution. Ablation studies are included in the Appendix (**??**).

A. Experimental Setup

We evaluated our framework in four contact-rich environments: Lift, Door, Wipe, and Cleanup. These interactions include basic object manipulation in the Lift environment, continuous contact in the Door and Wipe environments, and a mix of contact and manipulation interactions in the Cleanup environment. The robot utilized for these experiments was a Franka Emika Panda in the Robosuite simulator [46] (see Figure (5)) and real-world (see Figure (6) and (**??**) for details). We additionally apply domain randomization by randomly varying table friction, table height, object positions, and initial

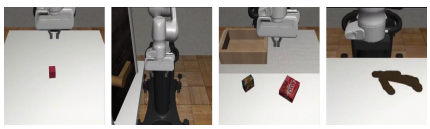

Fig. 5: Simulation Experiments: Lift, Door, Cleanup, Wipe

Fig. 6: Real Experiments: Lift, Cleanup, Wipe

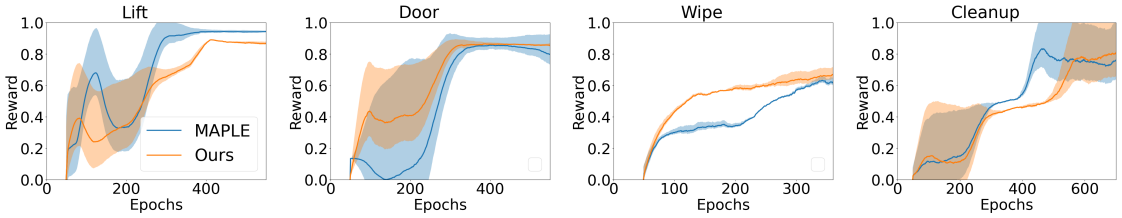

Fig. 7: Comparison of learning behavior and convergence times for various tasks. The rewards are averaged over 20 episodes then normalized between 0 and 1 (which represents the maximum reward at each timestep).

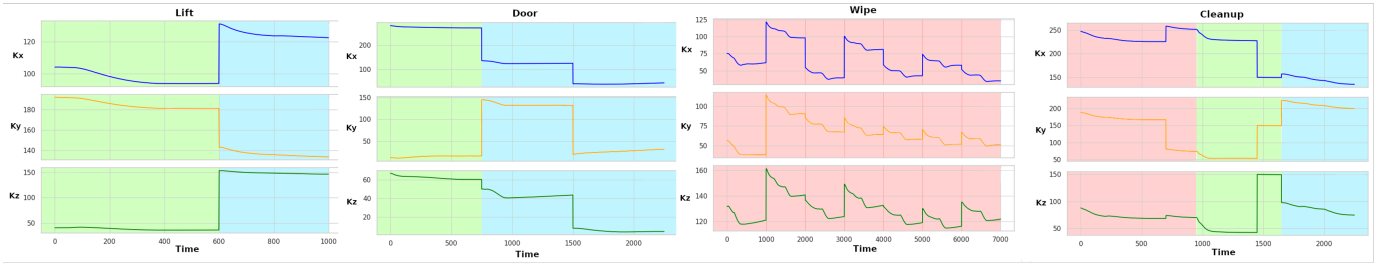

Fig. 8: Variable stiffness behavior demonstrating an emphasis on compliance and stiffness reduction. Each background grid colour represents a different primitive being executed - grasp, reach, push.

end-effector position. Lastly, all the documented results were averaged across 5 random seeds.

B. Comparative Analysis - Simulation

We compare our proposed framework with the MAPLE baseline. As for the hyperparameters employed during training, they are documented in Appendix **??**. The chosen evaluation metrics are Learning Performance, Maximum Interaction Force, Compositionality, and Success Rate.

Evaluation Metrics. In *Learning Performance*, we examine learning convergence time to assess learning efficiency of the proposed framework. In *Maximum Interaction Force*, we evaluate our framework's ability to adapt stiffness across diferent contexts and its effect on the applied forces. In *Compositionality*, we quantify recurring patterns in primitive selection using a *compositionality metric* [24](details in Appendix (**??**)). Lastly, in *Success Rate*, we analyze the framework's ability to consistently achieve the desired task objectives across the different environments.

Evaluation Results - Learning Performance. We analyzed convergence times by referring to the learning curves in Figure 7. Given that our approach and MAPLE use different affordances, then direct comparisons with MAPLE may not be appropriate since the reward functions are different. However, we can still assess convergence times, defined here as the time taken to learn a near-optimal policy for a given task.

In the Door environment, both our approach and MAPLE show approximately equal convergence times. For the Lift and Cleanup tasks, MAPLE converges slightly faster, possibly due to fewer primitive parameters and less exploration constraints from affordance coupling. In the Wipe task, our approach converges much faster, likely due to its ability to leverage variable stiffness, adapting force behavior to task requirements.

Evaluation Results - Maximum Interaction Force We demonstrate samples of the variable stiffness behavior across the different environments in Figure (8). We also include a graph showing the average applied end-effector forces over a sample of 500 evaluation runs in Figure (9) highlighting our framework's ability to finish the task while exerting less force. These forces were acquired directly from the simulation environment.

In the Lift and Cleanup environments, both of which are tabletop settings, K_z is maintained low when interacting near the table, while K_x and K_y are higher to ensure precise alignment with the objects of interest. In the Door environment, K_x is relatively high to provide stability during initial contact, with K_y increasing as the door handle is pushed down and all stiffness values decreasing when pulling the door open. In the Wipe environment, K_x and K_y are low since the primary action involves contact along the z-axis, while K_z maintains a higher value to exert enough force for effective wiping without excessive interaction forces.

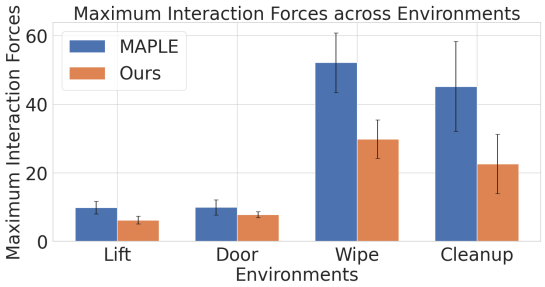

Fig. 9: Comparison of maximum interaction forces

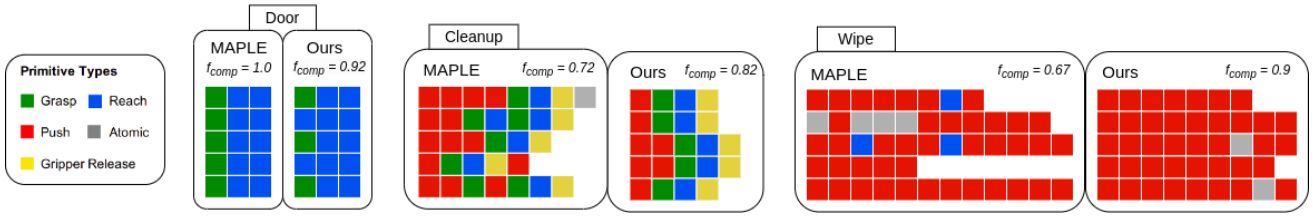

Fig. 10: Compositionality comparison showcasing the learned sequential behavior. The rows correspond to primtive sequences generated by 5 sample environment runs.

This reduction in stiffness translates to lower interaction forces across environments, as shown in Figure 9. Our approach consistently exerts less force, with lower standard deviation, implying less sensitivity to task randomization. Note that the average force was only calculated across the successful trials in order to avoid biasing the results, since a robot not performing any actions would yield no interaction force.

Evaluation Results - Compositionality. We quantify recurring patterns of primitive choices for solving a given task using a *compositionality metric* introduced by Nasiriany et al. [24]. A detailed mathematical explanation regarding compositionality calculations can be found in Appendix **??**.

The compositionality was calculated for a sample of 30 successful environment runs, illustrated in Figure (10). The Lift task was excluded as it had the same compositionality score ($f_{comp} = 1$), with a grasp and reach primitive sequence. In the Door task, we share the same number of primitive executions as MAPLE, but it shows more consistent primitive selection. In the Cleanup task, our approach reduces the number of primitive executions needed, likely due to more robust pushing and precise object approach. In the Wipe environment, our method has more consistent primitive selection than MAPLE, indicating better understanding of task requirements.

Evaluation Results - Success Rate. A comparison of success rates between MAPLE and our method is shown in Table II. Following training, the simulation and real world experiments were run 20 times to obtain the success rates. In the real world experiments, the policy was directly deployed onto the hardware with no fine-tuning to test the sim2real capabilities of the framework (**??**). MAPLE was not tested in real-world experiments due to its rigidity and potential operational hazards. Specifically, if the target state was defined at a location on or below the table surface, the robot's motion would lead to unintended force application and potentially cause damage to the environment.

Our approach achieves comparable success rates in Lift, Door, and Cleanup tasks while improving the safety of the system due to its higher degree of compliance. As for the Wipe task, our approach achieves double MAPLE's success rate. This significant improvement is attributed to our method's stiffness control capacity, as compared to MAPLE's use of position control. Position control strictly follows predefined trajectories, which worked well for tasks like Lift and Door, but is inadequate for wiping due to the rigidity of the end-effector. This can cause failures like applying insufficient/excessive force or losing contact with the surface during wiping. In contrast, our method demonstrated that leveraging stiffness control allows the system to dynamically adjust the force applied by the end-effector. This adaptability ensures consistent surface contact and prevents under- or over-application of force.

TABLE II: Success Rates (%) for Simulation and Real World

	Lift	Door	Wipe	Cleanup
MAPLE (Simulation)	100.0 ± 0.0	100.0 ± 0.0	42.0 ± 11.7	91.0 ± 5.8
Ours (Simulation)	100.0 ± 0.0	100.0 ± 0.0	86.0 ± 6.2	87.0 ± 6.1
Ours (Real World)	90.0	-	70.0	80.0

VII. Conclusions and Limitations

This paper presents a hierarchical reinforcement learning framework aimed at enabling adaptive stiffness control in sequential contact tasks. It utilizes a pre-defined library of behavior primitives and equips them with variable stiffness capabilities. This was done by incorporating an expanded action space to allow the agent to modify its stiffness and an adaptive controller for dynamic stiffness modifications during primitive execution. During training, we introduce affordance coupling to combine position and stiffness affordances, which promotes efficient exploration while incentivizing compliance. The framework showcases notable results in learning efficiency, variable stiffness control, compositionality in primitive selection, and success rates when compared to MAPLE, a state-of-the-art framework in sequential planning. Furthermore, real-world evaluations validate the proposed approach's sim2real capability.

The proposed method faces some limitations. The use of affordance coupling may limit learning efficiency when the task or subtask relies on accurate manipulation rather than contact or force interaction. This was evident in the experimental results for the Lift and Cleanup environments in which our method required more epochs to learn accurate manipulation. This can be attributed to the fact that affordance coupling incentivizes compliance, while manipulation tasks typically require some degree of stiffness to align the end-effector with a graspable object accurately. Another limitation lies in the acquisition of the adaptive stiffness controller parameters. Specifically, the controller relies on pre-defined scaling factors (β and γ) that need to be set. They are acquired either through kinesthetic demonstrations, which require physical interaction with the robot, or iteratively tuning β and γ to match the desired performance, which can be time-consuming.

REFERENCES

[1] A. Ranjbar, N. A. Vien, H. Ziesche, J. Boedecker, and G. Neumann, "Residual feedback learning for contact-rich manipulation tasks with uncertainty," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 2383–2390.

[2] Q. Li, M. Meier, R. Haschke, H. Ritter, and B. Bolder, "Object dexterous manipulation in hand based on finite state machine," in *2012 IEEE International Conference on Mechatronics and Automation*. IEEE, 2012, pp. 1185–1190.

[3] S. Nguyen, O. Oguz, V. Hartmann, and M. Toussaint, "Self-supervised learning of scene-graph representations for robotic sequential manipulation planning," in *Conference on Robot Learning*. PMLR, 2021, pp. 2104–2119.

[4] Z. Zhao, Z. Zhou, M. Park, and Y. Zhao, "Sydebo: Symbolic-decision-embedded bilevel optimization for long-horizon manipulation in dynamic environments," *IEEE Access*, vol. 9, pp. 128 817–128 826, 2021.

[5] M. M. Botvinick, "Hierarchical reinforcement learning and decision making," *Current opinion in neurobiology*, vol. 22, no. 6, pp. 956–962, 2012.

[6] D. W. Franklin, G. Liaw, T. E. Milner, R. Osu, E. Burdet, and M. Kawato, "Endpoint stiffness of the arm is directionally tuned to instability in the environment," *Journal of Neuroscience*, vol. 27, no. 29, pp. 7705–7716, 2007.

[7] L. Johannsmeier, M. Gerchow, and S. Haddadin, "A framework for robot manipulation: Skill formalism, meta learning and adaptive control," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5844–5850.

[8] R. Martín-Martín, M. A. Lee, R. Gardner, S. Savarese, J. Bohg, and A. Garg, "Variable impedance control in end-effector space: An action space for reinforcement learning in contact-rich tasks," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 1010–1017.

[9] F. J. Abu-Dakka, L. Rozo, and D. G. Caldwell, "Force-based variable impedance learning for robotic manipulation," *Robotics and Autonomous Systems*, vol. 109, pp. 156–167, 2018.

[10] S. Dou, J. Xiao, W. Zhao, H. Yuan, and H. Liu, "A robot skill learning framework based on compliant movement primitives," *Journal of Intelligent & Robotic Systems*, vol. 104, no. 3, p. 53, 2022.

[11] I.-A. Gal, A.-C. Ciocîrlan, and M. Mărgăritescu, "State machine-based hybrid position/force control architecture for a waste management mobile robot with 5dof manipulator," *Applied Sciences*, vol. 11, no. 9, p. 4222, 2021.

[12] Y. Onishi and M. Sampei, "Priority-based state machine synthesis that relaxes behavior design of multi-arm manipulators in dynamic environments," *Advanced Robotics*, vol. 37, no. 5, pp. 395–405, 2023.

[13] K. French, S. Wu, T. Pan, Z. Zhou, and O. C. Jenkins, "Learning behavior trees from demonstration," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7791–7797.

[14] F. Rovida, B. Grossmann, and V. Krüger, "Extended behavior trees for quick definition of flexible robotic tasks," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 6793–6800.

[15] S. Cheng and D. Xu, "Guided skill learning and abstraction for long-horizon manipulation," *arXiv preprint arXiv:2210.12631*, 2022.

[16] C. Agia, T. Migimatsu, J. Wu, and J. Bohg, "Taps: Task-agnostic policy sequencing," *arXiv preprint arXiv:2210.12250*, 2022.

[17] B. Wu, S. Nair, L. Fei-Fei, and C. Finn, "Example-driven model-based reinforcement learning for solving long-horizon visuomotor tasks," *arXiv preprint arXiv:2109.10312*, 2021.

[18] Y. Liu, D. Romeres, D. K. Jha, and D. Nikovski, "Understanding multi-modal perception using behavioral cloning for peg-in-a-hole insertion tasks," *arXiv preprint arXiv:2007.11646*, 2020.

[19] T. Zhang, Z. McCarthy, O. Jow, D. Lee, X. Chen, K. Goldberg, and P. Abbeel, "Deep imitation learning for complex manipulation tasks from virtual reality teleoperation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 5628–5635.

[20] B. Wu, F. Xu, Z. He, A. Gupta, and P. K. Allen, "Squirl: Robust and efficient learning from video demonstration of long-horizon robotic manipulation tasks," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9720–9727.

[21] J. Liang, B. Wen, K. Bekris, and A. Boularias, "Learning sensorimotor primitives of sequential manipulation tasks from visual demonstrations," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 8591–8597.

[22] A. Mandlekar, D. Xu, R. Martín-Martín, S. Savarese, and L. Fei-Fei, "Learning to generalize across long-horizon tasks from human demonstrations," *arXiv preprint arXiv:2003.06085*, 2020.

[23] D.-A. Huang, S. Nair, D. Xu, Y. Zhu, A. Garg, L. Fei-Fei, S. Savarese, and J. C. Niebles, "Neural task graphs: Generalizing to unseen tasks from a single video demonstration," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8565–8574.

[24] S. Nasiriany, H. Liu, and Y. Zhu, "Augmenting reinforcement learning with behavior primitives for diverse manipulation tasks," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 7477–7484.

[25] M. Dalal, D. Pathak, and R. R. Salakhutdinov, "Accelerating robotic reinforcement learning via parameterized action primitives," *Advances in Neural Information Processing Systems*, vol. 34, pp. 21 847–21 859, 2021.

[26] C. Agia, T. Migimatsu, J. Wu, and J. Bohg, "Stap: Sequencing task-agnostic policies," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 7951–7958.

[27] Y. Zhou, M. Do, and T. Asfour, "Learning and force adaptation for interactive actions," in *2016 IEEE-RAS 16th international conference on humanoid robots (humanoids)*. IEEE, 2016, pp. 1129–1134.

[28] B. Nemec, F. J. Abu-Dakka, B. Ridge, A. Ude, J. A. Jørgensen, T. R. Savarimuthu, J. Jouffroy, H. G. Petersen, and N. Krüger, "Transfer of assembly operations to new workpiece poses by adaptation to the desired force profile," in *2013 16th International Conference on Advanced Robotics (ICAR)*. IEEE, 2013, pp. 1–7.

[29] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 763–768.

[30] T. Cederborg, M. Li, A. Baranes, and P.-Y. Oudeyer, "Incremental local online gaussian mixture regression for imitation learning of multiple tasks," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 267–274.

[31] Y. Li, G. Ganesh, N. Jarrassé, S. Haddadin, A. Albu-Schaeffer, and E. Burdet, "Force, impedance, and trajectory learning for contact tooling and haptic identification," *IEEE Transactions on Robotics*, vol. 34, no. 5, pp. 1170–1182, 2018.

[32] D. Mitrovic, S. Klanke, and S. Vijayakumar, "Learning impedance control of antagonistic systems based on stochastic optimization principles," *The International Journal of Robotics Research*, vol. 30, no. 5, pp. 556–573, 2011.

[33] E. A. Rückert, G. Neumann, M. Toussaint, and W. Maass, "Learned graphical models for probabilistic planning provide a new class of movement primitives," *Frontiers in computational neuroscience*, vol. 6, p. 97, 2013.

[34] E. Theodorou, J. Buchli, and S. Schaal, "A generalized path integral control approach to reinforcement learning," *The Journal of Machine Learning Research*, vol. 11, pp. 3137–3181, 2010.

[35] J. Rey, K. Kronander, F. Farshidian, J. Buchli, and A. Billard, "Learning motions from demonstrations and rewards with time-invariant dynamical systems based policies," *Autonomous Robots*, vol. 42, pp. 45–64, 2018.

[36] M. Kim, S. Niekum, and A. D. Deshpande, "Scape: Learning stiffness control from augmented position control experiences," in *Conference on Robot Learning*. PMLR, 2022, pp. 1512–1521.

[37] M. Bogdanovic, M. Khadiv, and L. Righetti, "Learning variable impedance control for contact sensitive tasks," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6129–6136, 2020.

[38] C. C. Beltran-Hernandez, D. Petit, I. G. Ramirez-Alpizar, T. Nishi, S. Kikuchi, T. Matsubara, and K. Harada, "Learning force control for contact-rich manipulation tasks with rigid position-controlled robots," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5709–5716, 2020.

[39] P. Kulkarni, J. Kober, R. Babuška, and C. Della Santina, "Learning assembly tasks in a few minutes by combining impedance control and residual recurrent reinforcement learning," *Advanced Intelligent Systems*, vol. 4, no. 1, p. 2100095, 2022.

[40] W. Masson, P. Ranchod, and G. Konidaris, "Reinforcement learning with parameterized actions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, 2016.

[41] D. Xu, A. Mandlekar, R. Martín-Martín, Y. Zhu, S. Savarese, and L. Fei-Fei, "Deep affordance foresight: Planning through what can be done in the future," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6206–6213.

[42] P. Mandikal and K. Grauman, "Learning dexterous grasping with object-centric visual affordances," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 6169–6176.

[43] N. Vulin, S. Christen, S. Stevšić, and O. Hilliges, "Improved learning of robot manipulation tasks via tactile intrinsic motivation," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2194–2201, 2021.

[44] A. Faust, A. Francis, and D. Mehta, "Evolving rewards to automate reinforcement learning," *arXiv preprint arXiv:1905.07628*, 2019.

[45] M. Ulmer, E. Aljalbout, S. Schwarz, and S. Haddadin, "Learning robotic manipulation skills using an adaptive force-impedance action space," *arXiv preprint arXiv:2110.09904*, 2021.

[46] Y. Zhu, J. Wong, A. Mandlekar, R. Martín-Martín, A. Joshi, S. Nasiriany, and Y. Zhu, "robosuite: A modular simulation framework and benchmark for robot learning," *arXiv preprint arXiv:2009.12293*, 2020.