

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/340883675>

Stability-Guaranteed Reinforcement Learning for Contact-rich Manipulation

Preprint · September 2020

CITATIONS

0

READS

140

4 authors, including:



Shahbaz Khader

KTH Royal Institute of Technology

8 PUBLICATIONS 19 CITATIONS

SEE PROFILE



Pietro Falco

ABB

39 PUBLICATIONS 346 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



SAPHARI [View project](#)

Stability-Guaranteed Reinforcement Learning for Contact-rich Manipulation

Shahbaz A. Khader^{1,2}, Hang Yin¹, Pietro Falco² and Danica Kragic¹

Abstract—Reinforcement learning (RL) has had its fair share of success in contact-rich manipulation tasks but it still lags behind in benefiting from advances in robot control theory such as impedance control and stability guarantees. Recently, the concept of variable impedance control (VIC) was adopted into RL with encouraging results. However, the more important issue of stability remains unaddressed. To clarify the challenge in stable RL, we introduce the term *all-the-time-stability* that unambiguously means that every possible rollout will be stability certified. Our contribution is a model-free RL method that not only adopts VIC but also achieves *all-the-time-stability*. Building on a recently proposed stable VIC controller as the policy parameterization, we introduce a novel policy search algorithm that is inspired by Cross-Entropy Method and inherently guarantees stability. Our experimental studies confirm the feasibility and usefulness of stability guarantee and also features, to the best of our knowledge, the first successful application of RL with *all-the-time-stability* on the benchmark problem of peg-in-hole.

I. INTRODUCTION

In contact-rich manipulation, modeling and control of contacts are necessary for successful execution of the task. Traditional planning and control methods that specialize in free motion and obstacle avoidance do not address contact-rich manipulation adequately. Planning and control of contact-rich tasks is exceedingly difficult, especially when precise knowledge of the geometry and location of the manipulator and its surroundings is not available. The control of manipulator-environment interaction under the presence of uncertainties is generally studied as *interaction control* [1], but the terms *compliant manipulation* [2] and more recently *contact-rich manipulation* [3], [4], [5] have also been used. While seeking an appropriate control solution, an important property to be satisfied is stability, without which widespread adoption of robots cannot be realized.

Most *interaction control* methods assume the availability of a nominal reference trajectory, which again presupposes some degree of knowledge of the task geometry. Reinforcement learning (RL) has emerged as a promising paradigm that can alleviate this concern. Many recent works have applied deep RL methods for learning contact-rich manipulation policies that directly outputs torques or forces [5], [6], [7]. Although valid in principle, these methods do not

This work was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

¹Robotics, Perception, and Learning lab, Royal Institute of Technology, Stockholm, Sweden. {shahak, hyin, dani}@kth.se.

²ASEA Brown Boveri (ABB) Corporate Research, Sweden. pietro.falco@se.abb.com.

Correspondence to shahak@kth.se.

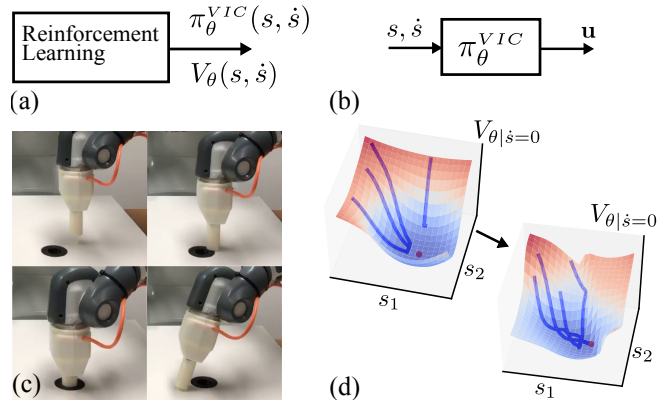


Fig. 1: (a) RL optimizes both the policy (π_{θ}) and Lyapunov function (V_{θ}) (b) The VIC policy maps states (position, velocity) to action (force/torque). (c) Examples of contact states in peg-in-hole (d) RL reshapes untrained V_{θ} (illustrative) that cannot guide trajectories to the goal (red point) to one that can while maintaining *all-the-time-stability*.

benefit from the rich theory of interaction control and more importantly do not guarantee stability. In an encouraging trend, a number of recent works have adopted variable impedance control (VIC), a well-known interaction control concept, into an RL policy and showed its usefulness [3], [8], [9]. Unfortunately, none of the above mentioned methods guarantee stability. To clarify the scope of the stability we are interested in, we define the term *all-the-time-stability* as guaranteed stability for all possible rollouts in the RL process. Our notion of stability is control theoretic (Lyapunov sense) and may not be confused with the requirement of smooth chatter-free contact, common in some industrial applications, that is also sometimes called “contact stability” [10].

We propose a model-free RL algorithm with *all-the-time-stability* that is well suited but not limited to contact-rich tasks. A model-free approach can avoid the complications of model learning [11]. Our approach leverages a previously proposed framework for modeling stable VICs for discrete interaction motions [12]. The main technical contribution is the introduction of a novel Evolution Strategy (ES) policy search method—inspired by Cross-Entropy Method (CEM) [13]—in which the sampling distribution is so constructed that it can only generate stable parameter samples. A novel update law is derived for iteratively updating the sampling distribution. Our experimental results show that not only *all-the-time-stability* is possible without sacrificing sample efficiency, but it also confirms its usefulness. We demonstrate, to the best of our knowledge, the first successful reinforcement learning of the classical benchmark problem of peg-in-hole [14], [4] with *all-the-time-stability*.

II. RELATED WORKS

This section is organized according to different topics:

1) *RL for Contact-rich Manipulation*: Learning artificial neural networks (ANN) policies for contact-rich manipulation has a long history [14], [15], but these early works adopted an admittance control approach that is more suited to non-rigid interactions [16]. Recent examples are [5], [6], [4], [17], where optimal control [5], motion planning [6], multimodal feature learning [4] and residual learning [17] were leveraged for making the problem tractable. Other approaches are model predictive control (MPC) with learned dynamics [18], [7] and learning reference force profile [19]. None of these methods guarantee control stability during learning or for the final policy.

2) *Learning Variable Impedance Policies*: Learning a VIC policy can be done either through learning from demonstration (LfD) [20], [21], [22], where human demonstration data is used to optimize policy parameters, or RL where a cost-driven autonomous trial-and-error process leads to a policy. A policy with VIC structure incorporates two levels of control loops: a trajectory generation loop and an impedance control loop. Guarantees on stability can be obtained only if both loops are considered in a unified way [12]. An example of LfD with stability guarantee is [22].

In this paper we adopt the paradigm of RL which has the benefit of alleviating the demand of human demonstrations. Time-dependent policies, without any stability guarantees, were learned in [8] and [23]. In [8] a Dynamical Movement Primitive (DMP) that encodes both reference and joint impedance trajectories is used, whereas in [23], a policy parameterized as a mixture of proportional-derivative systems is adopted. State-dependent policies were learned in [9], [3]. In [9], both the reference and stiffness trajectories are predicted from a policy parameterized by Gaussian Mixture Regression (GMR) model. Interestingly, stability was guaranteed for the trajectory generation loop, but the overall stability was unclear because of the lack of unified analysis of both loops. The ANN based method [3] learns a state-dependent policy that outputs reference trajectory and impedance parameters but offers no stability guarantees.

3) *Stable Variable Impedance Controllers*: A recent work [24] performed stability analysis for already existing trajectory and impedance profiles, but does not propose any policy structure for RL. In another interesting work [25], the stability issue of VIC was tackled by a passivity-based approach, but it too assumed a reference trajectory to be given. Khazari et al. [12] proposed a trajectory-independent modeling framework i-MOGIC for discrete motions, featuring VIC and stability guarantees. Our method adopts i-MOGIC as the policy parameterization and makes progress to form a model-free RL algorithm with *all-the-time-stability*.

4) *Cross-Entropy Method for RL*: The Cross-Entropy method [13], a general purpose sampling-based optimization method, has been previously used for policy search both in an unconstrained [26] and constrained [27] setting. Our approach can also be seen as a constrained policy search, but unlike [27], the constraint (symmetric positive definiteness)

is automatically guaranteed by the choice of the Wishart sampling distribution. Furthermore, unlike most cases where a Gaussian sampling distribution allows an analytical maximum likelihood estimation (MLE) based update, our special sampling distribution that consists of both Gaussian and Wishart factors necessitates a novel approach. Our policy search method may be seen as a CEM-inspired Evolution Strategy rather than a faithful implementation of CEM.

5) *Relation to Safe RL*: Stability has been considered as a means for safe RL in [28], [29], [30]. The method in [28] depends on smoothness properties of the learned Gaussian process (GP) dynamics model, something that is inappropriate for contact. [29] also used GP and is further limited to learning only the unactuated part of the dynamics, while the remaining is assumed to be a known control-affine model. Another example [30] assumes the availability of a stabilizing prior controller, which, along with the magnitude of the disturbance in the assumed nominal dynamics model, determines the region of guaranteed stability. In our case, such limitations do not exist since no model is learned; instead, our approach assumes a fully known model of the manipulator dynamics against which stability is derived. This is not a limitation because such models are easily available. Unknown interaction dynamics does not undermine stability as long as the interaction is passive (see Sec. III-D).

III. BACKGROUND AND PRELIMINARIES

A. Reinforcement Learning

A Markov decision process is defined by a set of states \mathcal{X} , a set of actions \mathcal{U} , a reward function $r(\mathbf{x}_t, \mathbf{u}_t)$, an initial state distribution $p(\mathbf{x}_0)$, a time horizon T , and dynamics $p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t)$. In policy search RL, the goal is to obtain the optimal stochastic policy $\pi_\theta(\mathbf{u}_t|\mathbf{x}_t)$ by maximizing the expected trajectory reward $\mathbb{E}_\tau[\sum_{t=0}^T r(\mathbf{x}_t, \mathbf{u}_t)]$, where $\tau = (\mathbf{x}_0, \mathbf{u}_0, \dots, \mathbf{x}_T)$ represents a sample from a distribution of trajectories $p(\tau)$ that is induced by $p(\mathbf{x}_0)$, $\pi_\theta(\mathbf{x}_t)$ and $p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t)$ over the time period T . This is usually done by a gradient-based ($\nabla_\theta J(\theta)$) search for the optimum value θ^* of the parameterized policy π_θ :

$$\theta^* = \underset{\theta}{\operatorname{argmax}} J(\theta) = \underset{\theta}{\operatorname{argmax}} \mathbb{E}_\tau \left[\sum_{t=0}^T r(\mathbf{x}_t, \mathbf{u}_t) \right], \quad (1)$$

In this paper, we consider a deterministic policy ($\mathbf{u}_t = \pi_\theta(\mathbf{x}_t)$) and deterministic dynamics ($\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t)$). Nevertheless, the general form of the problem in (1) remains unchanged since a distribution $p(\tau)$ can still be induced by $p(\theta)$ and $p(\mathbf{x}_0)$. $p(\theta)$ arises in the context of parameter space exploration [31], which is less common than action space exploration. Parameter space exploration allows exploration with deterministic policies.

B. Stability in RL

Lyapunov stability analysis deals with the study of time evolution of non-linear control systems such as robotic manipulators [32]. An equilibrium point of the system is stable if the state trajectories that start close enough are bounded around it and asymptotically stable if the trajectories

converge to it. Such properties are necessary to ensure that the system does not fail catastrophically in a wide range of circumstances. Stability is guaranteed by proving the existence of a certain Lyapunov function $V(\mathbf{x}_t)$ —a radially unbounded, positive definite and scalar function of the state \mathbf{x}_t —that has the properties $V(\mathbf{x}^*) = 0$ and $\dot{V}(\mathbf{x}) < 0 \forall \mathbf{x} \neq \mathbf{x}^*$ and $\dot{V}(\mathbf{x}^*) = 0$, where \mathbf{x}^* is the equilibrium point.

In an RL context, stability corresponds to a guarantee that any rollout is bounded in state space and tends to the goal position demanded by the task. Ideally, the policy should ensure the goal position as the unique equilibrium point, in which case we can refer to stability of the system and not just an equilibrium point.

C. Cross Entropy Methods

Cross Entropy Methods (CEM) [13] conducts sampling-based optimization to solve problems like (1). The optimization relies on a sampling distribution $q(\theta|\Phi)$ to generate the samples $\{\theta_n\}_{n=1}^{N_s}$. The performance of each θ_n is evaluated according to $J(\theta_n)$ and used to compute updates for the distribution parameter Φ . Formally, it iteratively solves:

$$\Phi^{i+1} = \underset{\Phi}{\operatorname{argmax}} \sum_n \mathbb{I}(J(\theta_n^i)) \log q(\theta_n^i|\Phi) \quad \theta_n^i \sim q(\theta|\Phi^i) \quad (2)$$

where $\mathbb{I}(\cdot)$ denotes an indicator function that selects only the best $N_e (< N_s)$ samples or *elites* based on individual performance $J(\theta_n)$. The computation of Φ^{i+1} from the samples $\{\theta_n^i\}_{n=1}^{N_e}$ is done by MLE. Very often $q(\theta|\Phi)$ is modeled as Gaussian, for which an analytical solution exists [13]. As the iteration index i grows, $q(\theta|\Phi^i)$ is encouraged to converge to a narrow solution distribution with high performance, thereby approximately solving (1).

D. i-MOGIC

Integrated MOtion Generator and Impedance Controller (i-MOGIC) was proposed as a modeling framework for discrete interaction motions [12]. The controller has the form of a weighted mixture of several spring-damper components:

$$\mathbf{u} = -\mathbf{S}^0 \mathbf{s} - \mathbf{D}^0 \dot{\mathbf{s}} - \sum_{k=1}^K w^k(\mathbf{s}) [\mathbf{S}^k (\mathbf{s} - \mathbf{s}^k) + \mathbf{D}^k \dot{\mathbf{s}}] \quad (3)$$

The stiffness and damping matrices are denoted by \mathbf{S} and \mathbf{D} , respectively. The superscript 0 is used to indicate the base spring-damper and k for a mixture component. \mathbf{s} and $\dot{\mathbf{s}}$ denote position and velocity of the manipulator. Note that \mathbf{s} is defined relative to the goal position, implying that the global attractor point is the origin. The attractor points for the remaining components are denoted by \mathbf{s}^k . The mixing weight $w^k(\mathbf{s})$ is a function of \mathbf{s} and parameterized by $\mathbf{S}^k, \mathbf{s}^k, l^k$, where l^k is a scalar quantity.

Equation (3) can be seen as a VIC policy [12] of the form:

$$\mathbf{u} = \pi_\theta(\mathbf{x}) = -\bar{\mathbf{S}}(\mathbf{x})(\mathbf{s} - \bar{\mathbf{s}}(\mathbf{x})) - \bar{\mathbf{D}}(\mathbf{x})\dot{\mathbf{s}}, \quad (4)$$

where $\mathbf{x} = [\mathbf{s}^T, \dot{\mathbf{s}}^T]^T$, \mathbf{u} is force/torque and $\bar{\mathbf{s}}$, $\bar{\mathbf{S}}$ and $\bar{\mathbf{D}}$ are state-dependent position reference, stiffness matrix and damping matrix, respectively. The parameter set of i-MOGIC

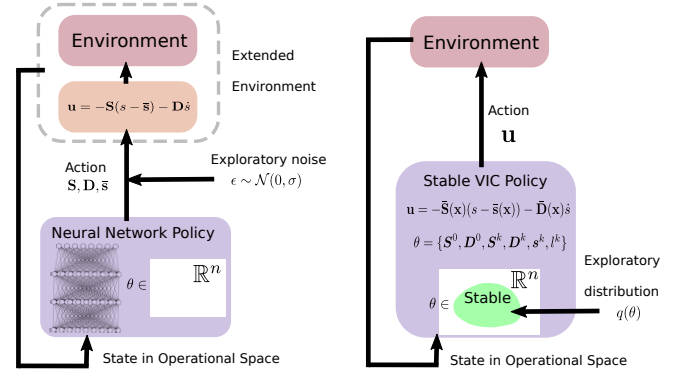


Fig. 2: RL with VIC policy structure. (Left): stability-unaware neural network policies outputting unconstrained impedance gains (Right): *all-the-time-stability* policy search with interpretable and constrained parameters.

is given by:

$$\theta = \{\mathbf{S}^0, \mathbf{D}^0, \mathbf{S}^k, \mathbf{D}^k, \mathbf{s}^k, l^k\} \text{ for } k = 1, \dots, K. \quad (5)$$

Khansari et al. [12] showed that, with gravity compensated, (3) is globally asymptotically stable (GAS) at the origin if:

$$\begin{aligned} \mathbf{S}^0 &= (\mathbf{S}^0)^T \succ 0 \quad \mathbf{D}^0 \succ 0 \\ \mathbf{S}^k &= (\mathbf{S}^k)^T \succeq 0 \quad \mathbf{D}^k \succeq 0 \quad l^k > 0 \quad \forall k = 1, \dots, K \end{aligned} \quad (6)$$

GAS and *passivity* is proven for free motion using only the manipulator dynamics while considering any interaction forces as persistent disturbance. When the interaction is with a *passive* environment, asymptotic stability may be lost but stability is retained [12]. A simple example is when an obstacle is preventing the manipulator from moving towards the goal; here, the contact forces constitute the persistent disturbance. A *passive* environment corresponds to the common case where objects in the environment are not actuated. Note that some terms in the original i-MOGIC are omitted in (3) since they are set to zero.

IV. APPROACH

As we have seen, i-MOGIC is a parameterized policy with a VIC structure and stability guarantees. To attain our goal of an RL algorithm with *all-the-time-stability*, we need a policy search algorithm that also features stable exploration and convergence guarantees. Our main contribution is a model-free RL algorithm that meets these requirements. We adopt i-MOGIC policy parameterization and propose a CEM-like policy search algorithm. i-MOGIC being a deterministic policy and its stability determined solely by its parameter values, the parameter space exploration strategy of CEM is ideal. Figure 2 provides further perspective by juxtaposing our solution with a deep RL approach with VIC structure and the more common action space exploration [3].

In our novel CEM-like algorithm, the constraints in (6) are guaranteed by designing a sampling distribution that makes it impossible to sample an unstable parameter set. A feasible parameter set of i-MOGIC is a mixed set of real-valued vectors, positive scalars, and matrices with symmetry and positive (semi)definiteness. Considering positive numbers as a special case of symmetric positive definite (SPD) matrices and enforcing SPD for all the matrices in (5), all parameters

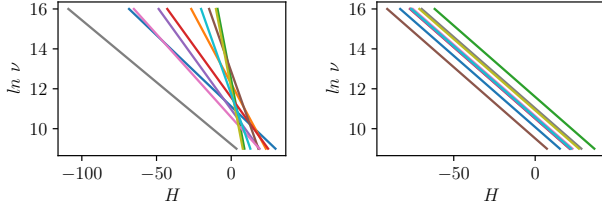


Fig. 3: The (apparent) linear relationship between $\ln \nu$ and entropy H of a Wishart distribution. $\mathbb{E}[S] = \nu W$ is fixed and $\nu > D + 1$. Each plot represents a different distribution. **Left:** random dimensions (D between 2 and 10) and random W parameters. **Right:** $D = 7$ and random W .

except s^k can be modeled by a distribution of SPD to guarantee constraint satisfaction. We focus on this aspect first and then develop the complete solution subsequently. Note that enforcing SPD for all matrices in (5) implies that our approach is slightly more conservative than it is required.

A. Optimizing Positive-definite Matrix Parameters

We model the sampling distribution for an SPD matrix with the Wishart distribution, which is fully defined by two parameters ν and W and denoted by $S \sim \mathcal{W}_D(S|W, \nu)$. $S \in \mathbb{R}^{D \times D}$ and $W \in \mathbb{R}^{D \times D}$ are SPD and $\nu > D - 1$. The expected value is given by $\mathbb{E}[S] = \nu W$. The variance of a random matrix is not easy to define but is controlled by ν . MLE-based update of Wishart distribution is possible if we adopt a numerical optimization approach for (2). Often, it is desirable to avoid such an approach because it may entail specification of parameter intervals, gradient computation or lack of convergence guarantees. Instead, we derive a simple update rule that, in spite of not targeting to solve (2), has the general property of continuously refining the sampling distribution towards an approximate solution. Although ours is no longer a faithful CEM method, it still conforms to the Evolution Strategy paradigm.

We propose to update the W parameter of the Wishart distribution by the empirical average of the elite samples $\{S_m\}_{m=1}^{N_e}$. This is valid since scaling and addition preserves symmetry and positive definiteness. Our update rule for the ν parameter is based on two modeling assumptions: $\Delta H \propto -\frac{(R_e - R_b)}{R_b}$ and $\Delta \ln \nu \propto -\Delta H$, where H is the entropy of the Wishart distribution; and R_b and R_e are the average rewards of all samples and elites, respectively. The first linear model is our design choice and is motivated by the intuition that there should be a reduction in entropy of the sampling distribution commensurate with a relative gain between R_b and R_e . The second model actually matches closely with reality as evidenced by Fig. 3. After combining the linear models and rearranging terms we obtain our update law:

$$W^{i+1} = \frac{1}{N_e} \sum_{m=1}^{N_e} \bar{S}_m^i, \quad (7a)$$

$$\nu^{i+1} = \nu^i \exp \left(\gamma \beta \left(\frac{R_e^i - R_b^i}{R_b^i} \right) \right) \quad (7b)$$

where \bar{S}_m^i is the elite sample scaled by $\frac{1}{\nu^i}$, i is the iteration variable (see (2)), and $\beta > 0$ and $\gamma > 0$ are the respective proportionality constants of the linear models. Since Fig.

Algorithm 1 Policy search with stability guarantees

- 1: Initialize $i = 0$;
- 2: $\Phi^0 = \{\Phi_s^0, \Phi_{S^0}^0, \Phi_{D^0}^0, \Phi_{S^k}^0, \Phi_{D^k}^0, \Phi_{l^k}^0\}$ for $k = \{1, \dots, K\}$
- 3: **while** not converged **do**
- 4: Do rollouts with $\theta_n^i \sim q(\theta_n^i | \Phi^i)$, $n = 1 \dots N_s$
- 5: Extract N_e samples from N_s based on performance measure $J(\theta_n^i)$ ▷ Sec. III-C
- 6: Compute Φ_s^{i+1} using MLE on $\{s_m^i\}_{m=1}^{N_e}$
- 7: Compute Φ_M^{i+1} using (7) on $\{M_m^i\}_{m=1}^{N_e}$,
- 8: $M \in \{S^0, D^0, S^k, D^k, l^k\}$ for $k = 1, \dots, K$
- 9: Increment i

3 also indicates that γ is dependent only on D , γ can be estimated once D is known (for example, $D = \dim(s)$). The constant β , on the other hand, is a hyperparameter and its tuning corresponds to the trade-off between exploration and exploitation. Higher values can achieve updates that favour exploitation instead of exploration because the variance of Wishart distribution is a decreasing function of ν .

We now focus on the convergence behavior of the proposed method. Notice that $R_e \geq R_b$ and therefore $\nu^{i+1} \geq \nu^i$. The boundary condition $R_e = R_b$ is satisfied only at convergence where all samples from the sampling distribution return identical rewards. In all other cases the distribution shrinks because of the decreasing nature of variance with increasing ν . This guarantees eventual convergence. Indeed, the converged solution may be a local optimum, as it would be the case for any policy search method. It is worth noting that although our modeling assumption, $\ln \nu \propto -H$, is most appropriate, any decreasing function ν of H is sufficient.

B. Stability-Guaranteed Policy Search

Our goal is to derive a model-free policy search method that guarantees the stability constraints of i-MOGIC. More specifically, we seek an algorithm that solves (1) using the general strategy in (2), if possible, but subject to the stability constraints in (6). The policy parameterization π_θ is (3), with the parameter set θ in (5). The only missing piece is the sampling distribution q and the exact strategy for the iterative update of its parameter Φ .

We define a sampling distribution of the form:

$$q(\theta | \Phi) = q(s | \Phi_s) q(S^0 | \Phi_{S^0}) q(D^0 | \Phi_{D^0}) \dots \prod_{k=1}^K q(S^k | \Phi_{S^k}) q(D^k | \Phi_{D^k}) q(l^k | \Phi_{l^k}), \quad (8)$$

where $q(\theta | \Phi)$ is the joint probability distribution of all the parameters in (5). After introducing a notational simplification $s = [(s^1)^T, \dots, (s^K)^T]^T$, $q(s | \Phi_s)$ is a multivariate Gaussian distribution with parameters $\Phi_s = \{\mu_s, \Sigma_s\}$; and $q(M | \Phi_M)$ represents Wishart distributions with parameters $\Phi_M = \{W_M, \nu_M\}$ where $M \in \{S^0, D^0, S^k, D^k, l^k\}$ for $k = 1, \dots, K$. While the matrix cases have dimension $D = \dim(s)$, $q(l^k)$ is a one dimensional Wishart distribution of positive numbers $l^k \in \mathbb{R}^+$ (also equivalent to Gamma distribution). Since we have modeled the individual elements

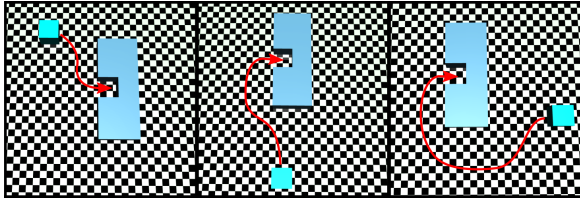


Fig. 4: **2D block-insertion tasks.** From left to right: Task1 (insertion clearance: 0.5 mm, execution time: 1s), Task2 and Task3 (insertion clearance: 2 mm, execution time: 2s). Block size is $50 \times 50 \times 50$ mm and weighs 2Kg. Rough illustrative paths are indicated by red arrows.

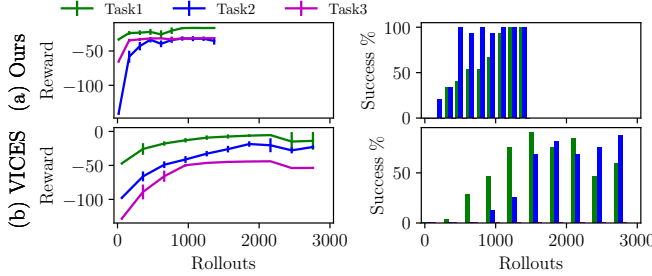


Fig. 5: **RL of 2D block-insertion tasks.** (a) Our method for 50 iteration with $N_s = 15$. (b) VICES for 50 iteration with $N_s = 30$. Success rate is among all rollouts per iteration.

in θ as independent random variables, each of them can be maintained and updated independently. The update of the Gaussian $q(s|\Phi_s)$ is performed by the standard practice of MLE. For the remaining Wishart cases we employ our novel strategy in (7). The complete algorithm is shown in Alg. 1.

V. EXPERIMENTAL RESULTS

Two experimental environments are defined: 2D block-insertion and robot peg-in-hole. VICES [3], a deep RL method based on Proximal Policy Optimization that features VIC and no stability guarantee, was chosen as the baseline. We used [16,16] and [32,32] as the network structures for the first and second tasks respectively. The platforms consisted of the simulator MuJoCo and the robot YuMi, both updated at 100 Hz.

A. Simulated 2D Block-insertion

In this experiment, three variations of the simulated 2D block-insertion tasks are set up by varying the initial position of the block and also the insertion clearance (Fig. 4). The policy controls the block by exerting an orthogonal 2D force (rotation not allowed) with the goal of inserting it into the slot. The task executions are expected to generate contacts between the block and the environment.

Other aspects include hyperparameter tuning, initialization and reward model. Through a process of grid search, we determined the number of spring-damper components (K) as 1 and 8 for Task1 and Task2 respectively. Task3 was not successful. Learning rate, $\beta = 1$, was found to work well. We fixed $N_s = 15$ and $N_e = 3$ for all our experiments. This choice is motivated by the real-world sample complexity of RL for our final experiment, where an excess of 15 rollouts per iteration may be undesirable. We initialized $q(\theta|\Phi)$ in an uninformative way as follows. All Wishart parameters W were set to the identity matrix (or 1 in the case of scalar)

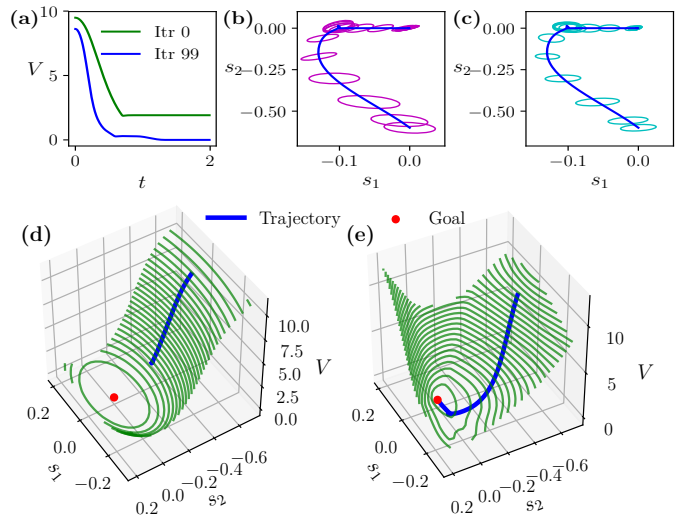


Fig. 6: **VIC policy is learned while maintaining stability (Task2).** Two rollouts are considered: τ_0 from iteration 0 (unsuccessful) and τ_{99} from iteration 99 (successful). (a) Lyapunov energy ($V(s, \dot{s})$) plots of the rollouts. Combined stiffness (b) and damping (c) matrices at regular intervals along τ_{99} . (d-e) Contour plots of $V(s, \dot{s})|_{\dot{s}=0}$ for τ_0 and τ_{99} , respectively, with their trajectories overlaid on them.

and the ν parameters were set to the minimum value $D + 1$. For the Gaussian distribution, we set the mean to zero vector and the covariance to identity. Finally, we adopted a reward model that consists of the Euclidean distance to the goal and a quadratic cost term for the actions.

The RL progress for both our method and VICES are shown in Fig. 5. We see that our method achieves better sample efficiency than VICES in all cases. Note that VICES required at least 30 rollouts per iteration and hence ended up having double the number of total rollouts. Task1 for VICES showed slight decline after initial progress; recall that insertion clearance for Task1 is the tightest so even a slight change can make a difference. Task3 was not successful for both our method (even for $K = 16$) and VICES; we believe this task can only be solved with reward shaping. See [8] for an example of shaping reward functions with multiple waypoints. We avoided this approach because our focus is on discovering close distance interaction behavior rather than complex motion profiles.

Does our method learn variable impedance at all? From the plots of the combined stiffness (Fig. 6b) and damping (Fig. 6c) matrices, along a successful rollout in the final iteration for Task2, we see that both stiffness and damping matrices have larger eigenvalues (higher impedance) at the beginning of the trajectory and gets smaller (lower impedance) at the vicinity and the interior of the slot. This is exactly what one would expect: higher impedance for free motion and lower impedance for contact motion. Note that the trajectory indicates free motion up until it makes contact followed by a smooth insertion. The first contact is visible as a small blip right before the insertion.

To test that stability is indeed maintained throughout the training, we plot the Lyapunov function $V(s, \dot{s})$ of iMOGIC, which shares parameters with the policy, for one rollout each from the first and last iterations (Fig. 6a). We see that

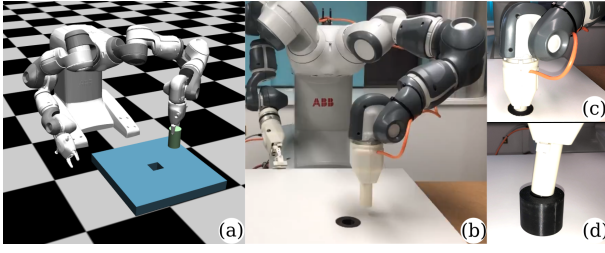


Fig. 7: **Peg-in-hole task** (a) MuJoCo environment: cylindrical peg (24 mm wide, 50 mm long), square hole (25 mm wide, 30 mm deep), and execution time of 2 s (b-d) Real-world environment: cylindrical peg (27 mm wide, 50 mm long), cylindrical hole (27.5 mm wide, 40 mm deep), and 5 s of execution time. (c) Successful insertion position. An insertion depth of 20 mm is considered successful. (d) 3D printed cylindrical peg and hole.

both plots are monotonically decreasing, indicating the main stability guaranteeing property $\dot{V} < 0$. Note that the plot of the first iteration case did not converge to zero indicating that it did not succeed. This is an example of the case where stability is preserved even when a passive environment is preventing convergence to the equilibrium point. In Fig. 6d-e, we see how RL has reshaped an initial Lyapunov function to one that is rich enough to succeed in the task—without compromising stability.

B. Peg-in-hole in Simulation

In this experiment we scale up to a 7-DOF manipulator arm that is expected to insert a cylindrical peg into a square hole (Fig. 7a). The peg-in-hole task has been historically considered as the benchmark problem for contact-rich manipulation [14], [15]. To ensure that the policy learns to exploit and also comply with environmental constraints, the initial position is chosen to be laterally away from the hole. The policy is implemented in the operational space including both translation and rotation according to Sec. 9.2.2. of [33], which uses the goal frame as the reference frame and adopts XYZ Euler angle representation. We restrict the Euler angle ranges to $\pm\pi/2$ with respect to the fixed goal frame to avoid representational singularities. This range is not practically limiting for most applications and we assume the rotation to remain in this range once it is initialized appropriately. Although more sophisticated representations such as rotation matrix and quaternions are interesting, i-MOGIC only supports Euler angles. As mentioned earlier, we fix the population parameters as $N_s = 15$ and $N_e = 3$. Other settings include $K = 2$ and $\beta = 10$. We adopt the reward model suggested in [5].

Unlike the previous experiment, where we did not initialize $q(\theta|\Phi)$ informatively, we now propose the following principled approach for an informative initialization. The base spring-damper distribution parameters W_{S^0}, W_{D^0} are initialized as, but not constrained to, diagonal matrices such that a suitable critically damped pair would bring the manipulator to a close vicinity of the goal in the desired time. For free motion such a value can be calculated analytically, but in the contact-rich case, an experimental approach is best. The parameters ν_{S^0}, ν_{D^0} indicate the level of confidence for W_{S^0}, W_{D^0} and we used an initial value of 30. For the remaining spring-damper components, the parameter W_{S^k}

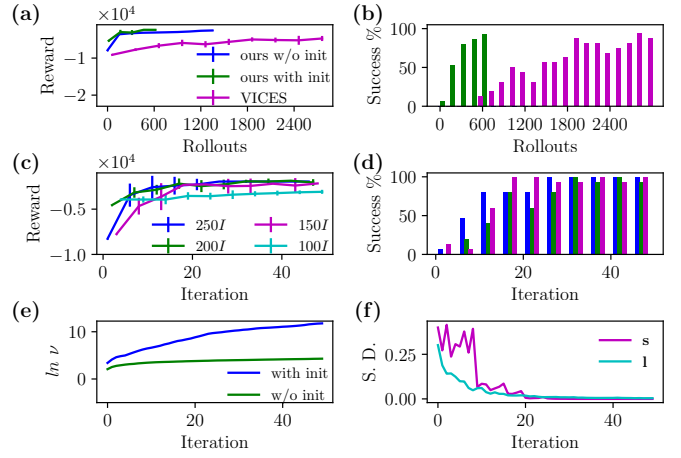


Fig. 8: **RL on simulated peg-in-hole** (a-b) RL progress comparison. Success rate is among all rollouts per iteration. (c-d) Robustness to variation in initial value of W_{S^0} . (e) Growth curve of the ν parameter (d) Average of element-wise standard deviations of s and l parameters (init case).

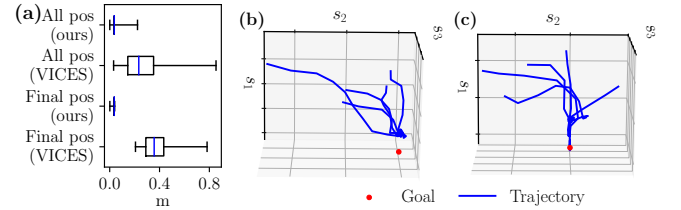


Fig. 9: **Benefits of stable RL** (a) Distributions of end-effector positions relative to the goal during the first 10 iterations of RL. Whiskers of the box plot include all samples (no outliers). Translation trajectories from five random initial positions show convergence before (b) and after (c) RL.

was initialized to a fraction of the base parameter ($W_{S^k} = \frac{1}{4}W_{S^0}$) and W_{D^k} set to the corresponding critically damped value. For each of the local attractor points, its Gaussian distribution mean μ_{s^k} is initialized to the zero vector (goal point) and the covariance Σ_{s^k} is initialized with a diagonal matrix such that the initial position of the manipulator is one unit of Mahalanobis distance away from the goal. Initializing $q(\theta|\Phi)$ with an informed Φ^{init} , such as above, is crucial for ES based methods (see Sec. VI).

Fig. 8a-b shows examples of our method with and without initialization in addition to VICES. The case with initialization converged in 50 iterations but VICES took about 100 iterations. The case without initialization did not succeed at all. As in the previous experiment, VICES required 30 rollouts per iteration for acceptable performance. Further, we can see in Fig. 8e that for the case with initialization, the corresponding ν parameter grows to a high value ($\ln \nu \approx 12$) indicating a corresponding reduction in entropy and consequently variance. In Fig. 8f we compare the average of element-wise standard deviations of the vectors s and l , which contains parameters s^k and l^k , respectively. Note that s is modeled by a Gaussian distribution that is updated by MLE and l is modeled by a Wishart distribution that is updated by our novel update rule. Remarkably, both cases appear to converge to zero at a similar rate. Fig. 8c-d show the result of variations in the initial value of parameter W_{S^0} (rotation component is uninformatively set to I) around the original value of 200I, which was obtained through

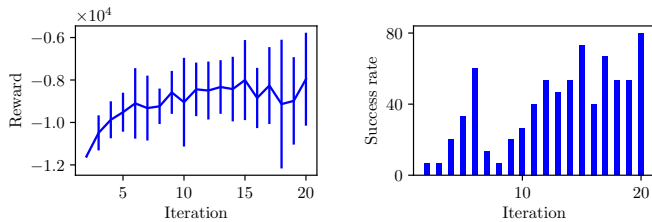


Fig. 10: **RL progress for peg-in-hole with real robot.** Success rate is % success among all rollouts per iteration.

our initialization scheme. It indicates sufficient robustness with only one case (100I) failing. Note that all other initial parameters are derived from W_{S^0} . Since VICES cannot not be informatively initialized, we cannot draw conclusions regarding sample efficiency in this case.

It is also interesting to see the stability property in action. Fig. 9a reveals the real benefit of stability guarantee in RL. We focus on the first 10 iterations of RL because it is during the early stages that stability pays off the most. We see that a lack of stability guarantee can cause potentially unsafe and unpredictable behavior as the robot end-effector covers large distances in the working range of the robot during exploration. It is also able to reach almost any final position. In stark contrast to VICES, our method restricts the robot to a significantly smaller region of the workspace and also forces it to consistently reach the nearest possible final position to the goal that is allowed by the environment. In Fig. 9b-c, we plot some trajectories for particular samples of policy parameters in the first (9b) and last (9c) iterations. Individual trajectories correspond to randomly chosen initial positions. Even without any learning (first iteration) the trajectories converge towards the goal position (9b). Similar behavior is evident for the fully learned final iteration also (9c). In fact, for the latter case, one of the rollouts even succeeded, indicating a potential for generalizing to initial positions.

C. Peg-in-hole with Real Robot

In this experiment we bring the previous simulated experiment of our method into the real world with some minor differences in the geometry and the initial position (Fig. 7b-d). Note that the insertion task is even more challenging with a clearance of only 0.5 mm. We use the same hyperparameter and parameter initialization values as before. A stopping condition for the RL process is introduced: 10 success out of 15 trials—provided that the reward has plateaued—is adopted.

The stopping condition was satisfied in iteration 20 with 11 successes out of 15 trials. The overall progress of RL can be seen in Fig. 10. Interestingly, due to good initialization, there were instances of insertions very early on; but, as it is evident from the success rate plot, it was also unreliable. The parameter ν reached a value of about 2000 at iteration 20. We recommend a final target value for ν in the interval $[1 \times 10^3, 10 \times 10^3]$ as a guiding principle for tuning the learning rate β . Throughout the RL process, the motion exhibited interaction behaviour that relied on environmental constraints for task execution. For example, the spike in success rate at iteration 5 corresponded to the behavior in

which the peg slid (on the horizontal surface) all the way to the hole. Later iterations learned a faster strategy to target the opening of the hole directly which also resulted in complex interactions. Stable behavior, such as represented in Fig. 9, was also evident throughout the RL process.

VI. DISCUSSION

Our results show that it is not only possible to achieve *all-the-time-stability* for RL in the context of contact-rich manipulation but that it also has important benefits. The most important benefit is that stability guarantee can lead to safe and predictable behavior at all stages of RL as evidenced by Fig. 9a. It may be possible to achieve such a behaviour for regular RL methods through prior simulation training or fine tuning of hyperparameters, but these strategies are only best-effort and do not offer any guarantees. In our case, such behaviors are a fundamental property that does not depend on any tuning. It is also worth pointing out that the very definition of stability ensures that the goal position of an episodic task is already built-in to the policy, unlike the case in deep RL methods where the goal is built-in to the reward function and it takes many subsequent iterations for that information to be induced into the policy. It is reasonable to assume that this may have been the reason for the better sample efficiency of our method even with uninformative initialization at times.

A natural concern is regarding the model capacity of the specialized policy i-MOGIC. The authors of [12] do mention that complex motion profiles may not be possible. Our results suggest that learning contact-rich manipulation that focuses more on interaction behavior rather than arbitrary motion profiles is well-suited for the i-MOGIC framework. ANN-based policy can be expected to have better capacity although, as it is usually the case with deep learning, it comes with data inefficiency. One indication of data inefficiency of ANN based VICES was that it required double the number of rollouts in an iteration. Interestingly, we achieved better efficiency for VICES than what was reported in the original work; for example, we achieved ~ 3000 rollouts in peg-in-hole task while most tasks took 5000 to 9000 rollouts (1 rollout = 200 steps) in [3]. A unique benefit of deep RL methods is that it can incorporate complex high dimensional state spaces such as image pixels. Stability guaranteed deep RL methods can have the best of both worlds and will be pursued in our future work.

ES algorithms often require proper initialization of the sampling distribution to succeed. More specifically, the true solution should be likely to be sampled from the very first iteration. If not, the algorithm will converge to any local optimum that is covered in the initial distribution. Thanks to our informative initialization scheme in Sec. V-B, which was possible only because of interpretable structures (spring-damper elements) in our policy, it is possible to effectively initialize our method. Note that in the case of our 2d block-insertion task even an uninformative initialization succeeded. Initialization also undoubtedly contributed to high sample efficiency of our method. Our novel update rule in (7) has an

additional benefit for real-world RL that imposes constraints on values for N_s and N_e . Notice that in Fig. 8f, MLE with few samples of N_e (3 in our case) has a tendency to collapse the sampling distribution rapidly and thus contribute to local optima. On the other hand, our method offers a fine control on the process through the learning rate β .

VII. CONCLUSION

In this paper, we set the ambitious goal of attaining stability-guaranteed reinforcement learning for contact-rich manipulation tasks. To include stable exploration into our notion of stability, we introduced the term *all-the-time-stability*. We built upon a previously proposed stable variable impedance controller and developed a novel model-free policy search algorithm that is inspired by Cross-Entropy Method. We formulated the sampling distribution such that it inherently satisfied stability conditions and then derived an update law for its improvement. The experimental results are significant because *all-the-time-stability* was achieved and its benefits demonstrated on the benchmark task of peg-in-hole without sacrificing sample efficiency.

REFERENCES

- [1] S. Chiaverini, B. Siciliano, and L. Villani, "A survey of robot interaction control schemes with experimental comparison," *IEEE/ASME Transactions on mechatronics*, vol. 4, no. 3, pp. 273–285, 1999.
- [2] K. J. A. Kronander, "Control and learning of compliant manipulation skills," Ph.D. dissertation, EPFL, 2015.
- [3] 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)*, 2019, pp. 1010–1017.
- [4] M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, "Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8943–8950.
- [5] S. Levine, N. Wagener, and P. Abbeel, "Learning contact-rich manipulation skills with guided policy search," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, 2015, pp. 156–163.
- [6] G. Thomas, M. Chien, A. Tamar, J. A. Ojea, and P. Abbeel, "Learning robotic assembly from cad," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 1–9.
- [7] A. Tamar, G. Thomas, T. Zhang, S. Levine, and P. Abbeel, "Learning from the hindsight planepisodic mpc improvement," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, 2017, pp. 336–343.
- [8] J. Buchli, F. Stulp, E. Theodorou, and S. Schaal, "Learning variable impedance control," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 820–833, 2011.
- [9] 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, no. 1, pp. 45–64, 2018.
- [10] S. S. M. Salehian and A. Billard, "A dynamical-system-based approach for controlling robotic manipulators during noncontact/contact transitions," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2738–2745, 2018.
- [11] S. A. Khader, H. Yin, P. Falco, and D. Kragic, "Data-efficient model learning and prediction for contact-rich manipulation tasks," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4321–4328, 2020.
- [12] S. M. Khansari-Zadeh, K. Kronander, and A. Billard, "Modeling robot discrete movements with state-varying stiffness and damping: A framework for integrated motion generation and impedance control," *Proceedings of Robotics: Science and Systems X (RSS 2014)*, vol. 10, p. 2014, 2014.
- [13] R. Y. Rubinstein and D. P. Kroese, *The Cross Entropy Method: A Unified Approach To Combinatorial Optimization, Monte-Carlo Simulation (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2004.
- [14] V. Gullapalli, R. A. Grupen, and A. G. Barto, "Learning reactive admittance control," in *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, 1992, pp. 1475–1480.
- [15] M. Nuttin and H. Van Brussel, "Learning the peg-into-hole assembly operation with a connectionist reinforcement technique," *Computers in Industry*, vol. 33, no. 1, pp. 101–109, 1997.
- [16] C. Ott, R. Mukherjee, and Y. Nakamura, "Unified impedance and admittance control," in *2010 IEEE International Conference on Robotics and Automation*, 2010, pp. 554–561.
- [17] T. Johannink, S. Bahl, A. Nair, J. Luo, A. Kumar, M. Loskyll, J. A. Ojea, E. Solowjow, and S. Levine, "Residual reinforcement learning for robot control," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6023–6029.
- [18] I. Lenz, R. A. Knepper, and A. Saxena, "Deepmpc: Learning deep latent features for model predictive control," in *Robotics: Science and Systems*, 2015.
- [19] J. Luo, E. Solowjow, C. Wen, J. A. Ojea, A. M. Agogino, A. Tamar, and P. Abbeel, "Reinforcement learning on variable impedance controller for high-precision robotic assembly," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3080–3087.
- [20] S. Calinon, I. Sardellitti, and D. G. Caldwell, "Learning-based control strategy for safe human-robot interaction exploiting task and robot redundancies," in *Proc. IEEE/RSJ Intl Conf. on Intelligent Robots and Systems (IROS)*, 2010.
- [21] K. Kronander and A. Billard, "Learning compliant manipulation through kinesthetic and tactile human-robot interaction," *IEEE transactions on haptics*, vol. 7, no. 3, pp. 367–380, 2013.
- [22] S. M. Khansari-Zadeh and O. Khatib, "Learning potential functions from human demonstrations with encapsulated dynamic and compliant behaviors," *Autonomous Robots*, vol. 41, no. 1, pp. 45–69, 2017.
- [23] P. Kormushev, S. Calinon, and D. G. Caldwell, "Robot motor skill coordination with em-based reinforcement learning," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.
- [24] K. Kronander and A. Billard, "Stability considerations for variable impedance control," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1298–1305, 2016.
- [25] F. Ferraguti, C. Secchi, and C. Fantuzzi, "A tank-based approach to impedance control with variable stiffness," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 4948–4953.
- [26] S. Mannor, R. Y. Rubinstein, and Y. Gat, "The cross entropy method for fast policy search," in *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 2003, pp. 512–519.
- [27] M. Wen and U. Topcu, "Constrained cross-entropy method for safe reinforcement learning," in *Advances in Neural Information Processing Systems*, 2018, pp. 7450–7460.
- [28] F. Berkenkamp, M. Turchetta, A. Schoellig, and A. Krause, "Safe model-based reinforcement learning with stability guarantees," in *Advances in neural information processing systems*, 2017, pp. 908–918.
- [29] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick, "End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks," in *The Thirty-Third AAAI Conference on Artificial Intelligence*, AAAI, 2019, pp. 3387–3395.
- [30] R. Cheng, A. Verma, G. Orosz, S. Chaudhuri, Y. Yue, and J. Burdick, "Control regularization for reduced variance reinforcement learning," in *Proceedings of the 36th International Conference on Machine Learning, ICML, 2019*, pp. 1141–1150.
- [31] P. Matthias, H. Rein, D. Prafulla, S. Szymon, C. Richard Y., C. Xi, A. Tamim, A. Pieter, and A. Marcin, "Parameter space noise for exploration," in *International Conference on Learning Representations (ICLR)*, 2018.
- [32] J.-J. E. Slotine, W. Li *et al.*, *Applied nonlinear control*. Prentice hall Englewood Cliffs, NJ, 1991, vol. 199, no. 1.
- [33] B. Siciliano, L. Sciacivco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control*. Springer Science & Business Media, 2010.