

Emergent Hand Morphology and Control from Optimizing Robust Grasps of Diverse Objects

Xinlei Pan^{1,2}, Animesh Garg^{1,3}, Animashree Anandkumar^{1,4}, Yuke Zhu^{1,5}

Abstract— Evolution in nature illustrates that the creatures’ biological structure and their sensorimotor skills adapt to the environmental changes for survival. Likewise, the ability to morph and acquire new skills can facilitate an embodied agent to solve tasks of varying complexities. In this work, we introduce a data-driven approach where effective hand designs naturally emerge for the purpose of grasping diverse objects. Jointly optimizing morphology and control imposes computational challenges since it requires constant evaluation of a black-box function that measures the performance of a combination of embodiment and behavior. We develop a novel Bayesian Optimization algorithm that efficiently co-designs the morphology and grasping skills jointly through learned latent-space representations. We design the grasping tasks based on a taxonomy of three human grasp types: power grasp, pinch grasp, and lateral grasp. Through experimentation and comparative study, we demonstrate the effectiveness of our approach in discovering robust and cost-efficient hand morphologies for grasping novel objects. https://xinleipan.github.io/emergent_morphology/

I. INTRODUCTION

Natural life provides ample evidence of a history of the body, the mind, and the environment all evolving together in cooperation with, and in response to each other [1]. An agent not only learns a policy with immutable morphology, but the morphology itself adapts (slowly) to the distribution of tasks it is exposed to. This shows up in many instances such the co-emergence of adducted thumb along with larger cognitive processing ability in humans. As humans developed rare abilities, such as tool use, throwing, and clubbing [2], both the hand and the ability to use it emerged. Likewise, for a robot to solve a task, say manipulating objects, the interplay between the robot embodiment (hardware) and its control algorithm (software) has an integral role in task success.

Meanwhile, recent advances in learning for robotics have yielded studies on data-driven techniques for robot grasping [3], [4]. Most of them have focused on grasp planning and selection [5], i.e., the control algorithm of grasping, with fixed and often simple parallel-jaw grippers. In contrast, customizing the robot embodiment to a particular task simplifies the computational burden of decision making, a phenomenon referred to as *morphological computation* in robot control [6]. For the task of grasping with different modes, robot hands with sophisticated mechanical mechanisms, such as passive compliance [7], [8], suction end-effectors [3], [9], and actuated rolling fingertips [10] have been carefully engineered to simplify grasp planning problems and improve performance. Though the data-driven approaches have given

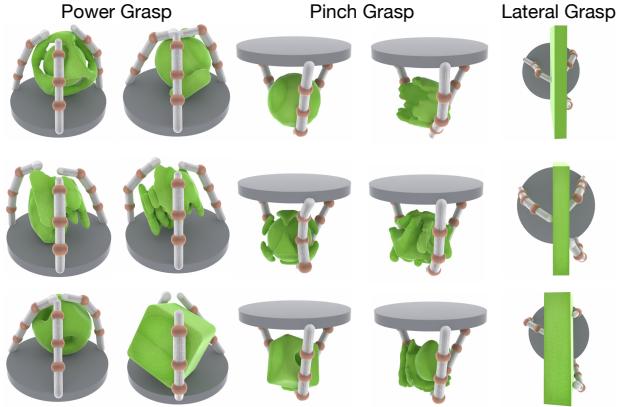


Fig. 1: **Emergent hand morphology and behavior.** We optimize hand design and control on grasping a diverse set of objects, aiming at maximizing the success rate and minimizing the hand complexity. Here are examples of the emergent hand design and grasping behavior from our model in three types of grasps. Our optimized hand is both effective and cost-efficient in its design.

rise to more robust grasps on various objects from raw sensory data, their effectiveness has been fundamentally handicapped due to the neglect of the impact of the robot embodiment, where failure modes are often attributed to the limitation of the gripper designs.

In this work, we introduce a data-driven approach to jointly optimizing robot embodiment and behavior for object grasping, with the aim of maximizing grasp performance. While optimizing for a specific task may lead to ill-formed hand designs that specialize at the task but fails on others, we demonstrate that robust and cost-efficient hand designs emerge from grasp optimization on a diverse of objects in three distinct grasp types (power, precision, and lateral), as classified under the taxonomy of human grasps [11]. In Figure 1 we show example hands optimized considering the performance and cost all together by optimizing the design on a diverse set of objects.

The key challenge of this problem resides in the high dimensionality of the optimization problem, involving a co-adaptation of the morphology and control parameters. The quality of the parameters can only be faithfully assessed through empirical grasping trials. However, the high costs incurred for conducting grasping trials, even in physics-based simulation, make it prohibitive for conventional black-box optimization techniques [12], [13]. Our intuition is that only a subset of design parameters have a strong impact on the grasping performance. Therefore, we can reduce the problem size by extracting these important factors through learning.

¹NVIDIA, ²University of California, Berkeley; ³University of Toronto, Vector Institute; ⁴Caltech; ⁵The University of Texas at Austin

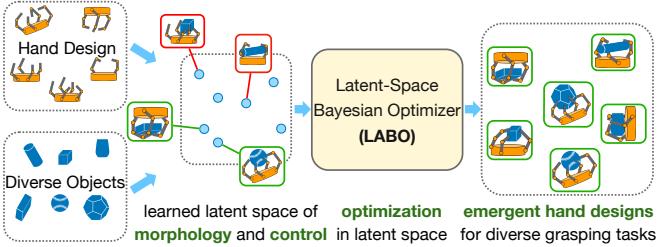


Fig. 2: High-level overview of our optimization framework. We jointly optimize the hand design and the control policy to solve diverse object grasping tasks. With representation learning of the design parameters in a latent space, our Bayesian optimizer (LABO) finds a robust and cost-efficient hand design for various grasp types.

To this end, our method learns latent representations of morphology and control parameters that capture the importance and redundancy of design parameters and thereby facilitate knowledge transfer across similar morphologies. Figure 2 shows a high-level overview of our latent-space optimization framework for hand design.

Summary of Contributions. (1) We develop a novel Bayesian Optimization (BO) method that harnesses a learned latent space of morphology and control parameters for efficient co-adaptation of both. (2) We design an auxiliary loss function which predicts the performance of the morphology and control design along with variational auto-encoder (VAE) to train an encoder that casts raw parameters into a latent feature space where important factors are magnified, which accelerates the optimization process by around 100 iterations. (3) Extensive experiments show that our approach outperforms standard black-box optimization baselines by around 20% of success rate. We also demonstrate that the emergent design with simple but effective three-fingers hand, trained on a diverse set of 3D shapes, generalizes well to novel objects unseen during training.

II. RELATED WORK

Optimization-Based Morphological Design. The physical embodiment impacts a robot’s potentials and limitations of dexterous behaviors. Carefully engineered robot designs have been shown to simplify the downstream computational problems of grasping and manipulation [6]–[10]. While robot designs have drawn direct inspirations from real-life organisms [14]–[16], it remains an immense challenge in dealing with numerous design parameters and practical constraints. Optimization-based methods have been used for automating the co-design of morphology and control. These include gradient-free methods, such as evolutionary algorithms [17]–[19], and gradient-based methods, such as constrained optimization [20], [21], differentiable simulation [22], [23], and policy optimization [24]. However, these methods require either great amount of samples to search in the high-dimensional design space or manual specification of constraints. Our method efficiently co-optimizes the robot morphology and the control policy for grasping.

Knowledge Sharing Across Embodiments. Most of the existing studies for embodied agents focus on generalizing towards environmental variations for a specific agent.

Sharing knowledge across agents of different embodiments has gained an increased attention [22], [25]–[28]. Knowledge transfer is typically achieved by either exploiting the compositional structures of embodiment [25], [26], [28] or by learning shared representations of the embodiment [22], [28]. This has enabled effective transfer of robot skills [25], fast adaptation to test-time changes [26], and efficient policy training with new morphologies [27]. Most relevant to ours is learning-in-the-loop optimization [22], which learns a latent space for designing soft robots. However, they have to rely on a fully-differentiable physical simulation for evaluating and updating the embodiment.

Data-Driven Grasping Methods. Recent advances in data-driven grasping methods [3], [4], [29]–[32] have substantially improved the performance of robotic grasping in unstructured environments from raw sensory input. To cope with the data-hungry nature of learning grasping models, especially with deep networks, simulated data has been extensively used as the training sources [33]–[35]. While some works have recently explored grasping and manipulation of dexterous or soft hands [36], [37], the main body of the data-driven grasping literature has centered on simple and fixed end-effectors, often the parallel-jaw grippers, where the grasp prediction problem can be reduced to computing the gripper pose. Closest to our work is UniGrasp [5], which learned a gripper representation in order to transfer grasps across a set of N -fingered robotic hands. However, it only considers a fixed set of grippers rather than optimizing the hand morphologies.

Latent Space Bayesian Optimization. Latent space BO has been explored in chemistry and medical domains, where prior approaches have explored unsupervised learning methods, such as variational auto-encoder (VAE) [38], [39], to perform dimension reduction of the search space for BO. In contrast, we introduce auxiliary supervision of grasping success labels to expedite task-oriented representation learning, and use empirical simulated grasping trials with the latent-space Bayesian optimizer for data-driven hand design.

III. MORPHOLOGY DESIGN WITH LATENT SPACE BAYESIAN OPTIMIZATION

We introduce a data-driven approach to emergent morphological and policy design. Inspired by the morphological evolution of human hands [40] and the task-oriented optimization of robot design [41], we study the problem of jointly optimizing the morphology and control through grasp trials on a diverse set of objects. We examine how joint optimization on these diverse tasks could generate a robust hand morphology and control design that generalizes to unseen scenarios.

The co-optimization of morphology and control imposes significant challenges due to the high-dimensional parameter space, where naïve sampling methods fail to discover effective designs. Our intuition is to exploit the correlation and redundancy in the design parameters – some parameters have a stronger influence in task performance than others, and very

often several of them work in synergy to shape the robot behaviors. For this reason, learning latent representations of the design parameters could capture such relationships among the parameters, leading to efficient search in the design space. Therefore, we use a representation learning module that transforms the raw parameter space into a reduced latent space. The learned latent space can be seamlessly integrated with distributed Bayesian optimization algorithms for black-box optimization of grasp quality metrics. We name our proposed approach LAtent-space Bayesian Optimization (LABO), as shown in Figure 3. Next, we first give a formal problem formulation and then introduce our LABO model.

A. Problem Formulation

We define the morphology and control parameters as θ , where $\theta \in \mathbb{R}_{\geq 0}^d$ and d is the total dimension of the parameter space. A parameter θ can be further partitioned into two components (θ_m, θ_a) , where θ_m specifies the robot hand model, such as the number of fingers and their lengths, and θ_a defines the action command that actuates the hand for grasping, and we use torque control for each joint in the robot. Our goal is to maximize a black-box objective function F by searching for a specific combination of morphology and control parameter θ^* such that

$$\theta^* = \arg \max_{\theta} F(\theta), \quad (1)$$

where F is a score function that evaluates the quality of individual morphology and control designs. In practice, F can be a simulator that instantiates the hand model and performs grasp trials to evaluate empirical performances of the design. Unlike prior work that used differentiable simulation [22], [23], we only assume black-box access to this function, and each call of the function incurs non-trivial computational cost, for instance, running physical simulation of grasping diverse objects.

We define the grasping task space as \mathcal{T} . Each unique task $t \in \mathcal{T}$ corresponds to a specific grasp type (power, pinch, or lateral) and a specific object instance. The task t will take in the morphology and policy parameter θ and return the episodic reward obtained: $\mathcal{R}(\theta)$. The reward function is related with the quality and success rate of the grasping, also related with how robust it is against perturbations. In addition, we want to encourage a cost-effective design of the hand, such that non-functioning hand parts will be suppressed. The cost is task independent, and each hand design corresponds to a cost value $C(\theta)$, which corresponds to the morphology complexity. The score function is thus a weighted combination of the rewards and the cost:

$$F(\theta) = w_1 C(\theta) + w_2 \sum_{i=1}^{|\mathcal{T}|} \mathcal{R}_i(\theta), \quad (2)$$

where w_1 is the weight for the cost, and w_2 is the weight for the episodic rewards on all tasks, and $|\mathcal{T}|$ is the number of all tasks.

Reward and Cost Functions. The reward per step is defined per grasping task. Denote finger self-collision as c_s

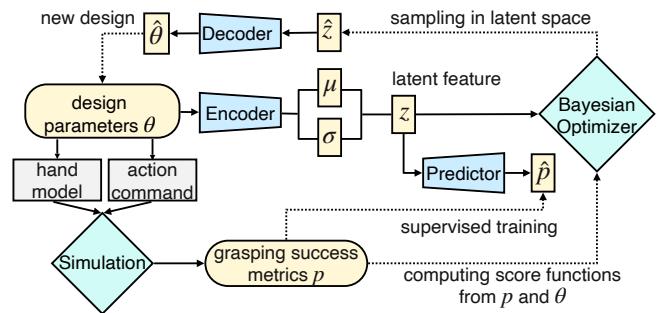


Fig. 3: **Schematic diagram of our LABO model.** We learn a representation learning module, consisting of the encoder, decoder, and predictor to cast raw morphology and control parameters θ into a latent representation z . Then we perform Bayesian optimization in the latent space. We rely on simulated grasp trials to evaluate the generated hand morphology and control on success metrics p .

(the value is 0 or 1), and the proportion of finger tips that are in contact with the object as r_o , the difference between the initial object position and orientation and the current object position and orientation as d_{pos}, d_{orn} , we have the following rewards definition:

$$\mathcal{R}_{power} = -0.01c_s + 0.1r_o - 0.1d_{pos}, \quad (3)$$

$$\mathcal{R}_{pinch} = -0.01c_s + 0.1r_o - 0.1d_{pos}, \quad (4)$$

$$\mathcal{R}_{lateral} = -0.01c_s + 0.1r_o - 0.05d_{pos} - 0.05d_{orn}. \quad (5)$$

Define the number of fingers as n_f and the number of finger segments as $n_{s,i}$, for $i \in \{1, 2, \dots, n_f\}$, the cost function is

$$C = \frac{n_f - 2}{4} + \sum_{i=1}^{n_f} \frac{n_{s,i} - 3}{3}. \quad (6)$$

The grasp success is checked by applying external adversarial perturbation forces from random directions at the center of mass of the object. If the object is held tightly in hand without dropping, the grasp is deemed a success; otherwise, a failure. The action space is consisted of the joint torques applied on all finger joints.

B. Representation Learning of Design Parameters

Directly optimizing in the raw parameter space suffers from the curse of dimensionality. To reduce the problem complexity and encourage knowledge transfer among different hand designs, we develop an encoder-decoder architecture to learn a latent space of design parameters, and use auxiliary losses to facilitate the representations to capture task-oriented information.

Denote the encoder network as ϕ_E and decoder network as ϕ_D . The encoder network casts raw parameters θ into a probabilistic lower-dimensional latent space characterized by the mean μ and the standard deviation σ , where we use the reparameterization trick [42] to draw a sample in the latent space $z = \mu + \sigma \odot \epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$. To ensure that the latent space representation is constricted within a bounded region for the ease of the subsequent optimization, we further take a sigmoid function on z : $f = \text{sigmoid}(z) = \frac{1}{1+e^{-z}}$ so that $f \in \mathbb{R}_{\geq 0}^{d'}$, where $d' \ll d$ and $\|f\|_\infty = 1$. The decoder

reconstructs the original input from the latent representation. This can be written as

$$\theta \xrightarrow{\phi_E} (\mu, \sigma) \xrightarrow{\epsilon \sim \mathcal{N}(0, 1)} z = \mu + \sigma \odot \epsilon \rightarrow f = \text{sigmoid}(z) \xrightarrow{\phi_D} \hat{\theta}. \quad (7)$$

We further regularize the latent space by restricting the KL divergence between $\mathcal{N}(\mu, \sigma)$ and $\mathcal{N}(0, I)$. However, the simple auto-encoding objectives are insufficient to inform the representations about the quality of the parameters in terms of task performance. We thus propose to use the grasp success metrics p as an additional supervision for learning the latent representation f , where p is a binary vector of which each dimension corresponds to the grasp success label on each training object. We train another prediction network ϕ_P that predicts the values of p from the representation f with binary cross-entropy losses. The information flow of the auxiliary task can be expressed as

$$\theta \xrightarrow{\phi_E} z \rightarrow f \xrightarrow{\phi_P} \hat{p}. \quad (8)$$

To sum up, the overall loss function for training the representation module is,

$$L_{rep}(\theta, p, \phi_E, \phi_D, \phi_P) = \frac{1}{d} \|\theta - \hat{\theta}\|^2 + \text{CELoss}(p, \hat{p}) + D_{KL}(\mathcal{N}(0, I) || \mathcal{N}(\mu, \sigma)), \quad (9)$$

where $\hat{\theta} = \phi_D(f)$, $\hat{p} = \phi_P(f)$, CELoss denotes the binary cross entropy loss, and D_{KL} denotes the KL-divergence of two distributions. Note that, in our optimization, the reward function $\mathcal{R}(\theta)$ in Equation (2) is a dense reward function more informative than the binary grasp labels p .

C. Latent Space Bayesian Optimization

Bayesian Optimization (BO) [13] is a classical family of methods to solve black-box optimization problems. Here we assume the function to be optimized is an unknown (and often costly) function with only inquiry access. Thus, solving the optimization problem with as few samples as possible is vital. This is particularly important for us, as evaluating a hand on a set of hundreds of tasks takes millions of simulation steps, and fabricating a new hand design in the real world can be even more costly. Therefore, LABO harnesses the learned latent representations to accelerate the Bayesian optimization process.

Pretrain the Representation Learning Module. Before the optimization process starts, we pretrain the latent space by drawing random design parameters and train the encoder and decoder with the VAE objective [42]. The pretraining loss function can be expressed as

$$L_{pre}(\theta, \phi_E, \phi_D) = \frac{1}{d} \|\theta - \hat{\theta}\|^2 + D_{KL}(\mathcal{N}(0, I) || \mathcal{N}(\mu, \sigma)). \quad (10)$$

As we have not tested these designs in simulated trials, we do not have the grasp labels p for supervising the auxiliary losses as in Equation (9). Nonetheless, this procedure yields a good initialization of the latent representations to be further fine-tuned over the course of optimization.

Joint Bayesian Optimization with Representation Learning. After the initial pretraining, we start to use BO to

Algorithm 1 Algorithmic Framework for Morphology and Control Optimization (LABO)

```

1: Input:
2:   BO iterations:  $N_{BO}$ 
3:   Number of pretrain data points:  $N_{pre}$ 
4:   pretrain iterations:  $N_1$ 
5:   score function (simulator) with input  $\theta$ :  $F(\theta)$ 
6:   UCB acquisition function hyperparameter:  $\beta$ 
7:   representation learning iterations per BO iteration:  $N_2$ 
8:   number of samples to draw for each BO iteration:  $N_3$ 
9: Initialize:
10:   $\phi_E, \phi_D, \phi_P, A(\xi, \beta)$ 
11:   $\mathcal{D} = \{\}$                                 ▷ Pretraining dataset
12:   $Q = \{\}$                                 ▷ BO dataset
13: Start:
14:   $\mathcal{D} \sim Uniform[0, 1] \in \mathbb{R}^d$  ▷ sample  $N_{pre}$  pretraining data
15:  For  $i = 1$  to  $N_1$ :
16:    update  $\phi_E$  with  $\nabla_{\phi_E} L_{pre}(\mathcal{D}, \phi_E, \phi_D)$ 
17:    update  $\phi_D$  with  $\nabla_{\phi_D} L_{pre}(\mathcal{D}, \phi_E, \phi_D)$ 
18:  For  $i = 1$  to  $N_{BO}$ :
19:     $f \sim A(\xi, \beta)$  and  $\hat{\theta} = \phi_D(f)$     ▷ sample new data point
20:     $q = (f, F(\theta))$                       ▷ evaluate the new data in simulation
21:     $Q = Q \cup \{q\}$ 
22:    Fit  $G(\xi)$  with  $Q$  once every  $N_3$  steps    ▷ Equation (11)
23:  For  $j = 1$  to  $N_2$ :
24:     $\nabla_{\phi_E} = \nabla_{\phi_E} L_{rep}(Q, \phi_E, \phi_D, \phi_P)$ 
25:    +  $\nabla_{\phi_E} L_{pre}(\mathcal{D}, \phi_E, \phi_D)$ 
26:    update  $\phi_E$  with  $\nabla_{\phi_E}$ 
27:     $\nabla_{\phi_D} = \nabla_{\phi_D} L_{rep}(Q, \phi_E, \phi_D, \phi_P)$ 
28:    +  $\nabla_{\phi_D} L_{pre}(\mathcal{D}, \phi_E, \phi_D)$ 
29:    update  $\phi_D$  with  $\nabla_{\phi_D}$ 
30:     $\nabla_{\phi_P} = \nabla_{\phi_P} L_{rep}(Q, \phi_E, \phi_D, \phi_P)$ 
31:    update  $\phi_P$  with  $\nabla_{\phi_P}$ 

```

perform optimization in the latent space. As more effective hand designs are expected to be gradually discovered, we can in turn use these new samples to update the representation module for subsequent optimization steps. For each BO iteration, we perform multiple iterations of representation learning. To speed up the process, rather than using a sequential sampler, we employ the parallelized Bayesian Optimization framework BoTorch [43] for distributed sampling in the latent space. We use a Gaussian Process to model the data distribution, consisting of the morphology and control parameters in the latent space and their corresponding score value $F(\theta)$: $Q = \{q_i = (f_i, F(\theta_i))\}_{i=1}^N$, where N is the number of data points the algorithm has seen so far. Given this dataset, we fit a surrogate distribution model of the data, with parameters ξ :

$$G(q_{1:k}; \xi) \sim \text{Normal}(\mu(q_{1:k}), \Sigma(q_{1:k}, q_{1:k}); \xi), \quad (11)$$

where $\Sigma(q_{1:k}, q_{1:k})$ is the kernel function and we use the Matern kernel

$$\Sigma(x, x') = \alpha \frac{2^{1-v}}{\Gamma(v)} (\sqrt{2v} \|x - x'\|)^v K_v(\sqrt{2v} \|x - x'\|), \quad (12)$$

where K_v is a modified Bessel function with parameter v , and α is another hyperparameter. We use Upper Confidence Bound (UCB) as our acquisition function:

$$A(f; Q, \xi, \beta) = \mu(f; Q, \xi) - \beta \Sigma(f; Q, \xi), \quad (13)$$

where β is a parameter that can be tuned to trade off exploitation and exploration. The UCB algorithm is chosen since it has an additional parameter to tune how much exploration is done, which is important given the dimension of this optimization. We choose the Matern kernel since the changes in the morphology space is non-smooth (changes of numbers of fingers and finger segments), and a traditional Gaussian kernel might fail to model that well. Each time multiple data points are sampled to fit the surrogate model, and new samples are generated by maximizing the acquisition function. We include a complete pseudocode in Algorithm 1.

IV. EXPERIMENTS

To operationalize our idea, we develop a new simulated platform in BulletPhysics [44] that supports procedural generation of parameterized robot hand morphologies and actuating the hand with joint-space controllers. We design our experiments to investigate the following questions: 1) Can the proposed model discover robust hand designs of high success rate and low cost, in comparison to baseline methods? 2) How well does the final hand design generalize to unseen objects in different grasp types? 3) How does the performance change with respect to objects of varying complexity? 4) How is the performance of simpler hand with fewer fingers compared to complex hand with more fingers. To answer these questions, we compare with several baselines and present both quantitative and qualitative results in this section.

A. Experiment Setup

Grasp Types. To facilitate robust morphologies to emerge, we select three types of grasps from the human grasp taxonomy [11], including power, pinch, and lateral grasp. We give definitions of these types of grasping tasks and provide a visualization of these types of grasps in Figure 4a:

- 1) *Power Grasp*: The hand exerts forces to wrap the objects, with the surface of these fingers in a close contact with the object and the palm supporting the object. In this task, the palm faces upward and the object starts within the hand and we let the hand close to perform the grasp.
- 2) *Pinch Grasp*: The fingers mainly rely on the fingertips to pick up the object, and the palm region will not support the object as well. In this task, we keep the palm facing downward and the hand will pick up an object below it and raise the object to a certain height.
- 3) *Lateral Grasp*: The hand uses finger inner surface and tips to support the object. Two set of fingers support opposite sides of an object. In this task, the target objects are mainly shaped like a thin plane. The hands will have the palm facing horizontally and try to grasp a plane object from the side of the object.

Object Datasets. To get a diverse set of objects that demand different grasp strategies, we use the objects from the recently introduced EGAD dataset [45] and select 48 objects of varying grasp difficulties to assist evaluating the performance of the optimized hand on power and pinch

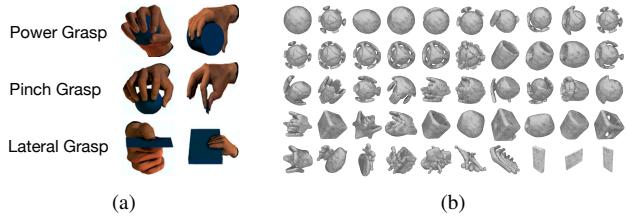


Fig. 4: (a) Three main grasp types of the human grasp taxonomy: power grasp, lateral grasp, and pinch grasp (images are taken from Feix et al. [11]); (b) Example objects we used for evaluating grasps. These objects are either selected from the EGAD dataset [45] or procedurally generated.

grasps. For lateral grasps, we generate thin boxes with different sizes and shapes. Example objects are shown in Figure 4b. The objects range from regular polygons, such as spheres and boxes, to irregular shapes, such as mugs.

Parameterized Design Space. Our original parameter space consists of 185 dimensions, with each dimension ranging between 0 and 1, denoted as $\theta \in \mathbb{R}_{\geq 0}^{185}$, where $\|\theta\|_\infty = 1$. The first 122 dimensions correspond to the parameters θ_m that determine a specific hand design and the rest 63 dimensions determine the control θ_a for the three types of grasping tasks. For simplicity, we use open-loop control and share the same action command for objects in each grasp type. The 122 dimensions include parameters for number of fingers, mass, surface friction, finger shape parameters. The additional dimensions that determines the policy of the robot are related with the torque applied on each of the joints. These parameters are chosen since they affect the structure and the physical characteristics of the hand. The control policy parameters have 63 dimensions, where each grasping task (power, pinch, lateral) get 21 dimensions. These 21 dimensions specified the torque applied on the palm-finger joints, finger-self joints and finger-tip joints for a maximum of 6 fingers and 6 finger segments.

Comparative Study of Hand Complexity. To analyze how grasping performance varies with respect to the hand complexity, we compare the performance of hands with different numbers of fingers. We fix the number of fingers of the hand to be 2 to 6, and perform optimization with LABO.

B. Experiment Results

Baseline Comparisons. We compare LABO with three baselines for black-box optimization:

- 1) *Uniform Sampling* (Uniform): we randomly sample morphology and control parameters uniformly from [0,1] in the original raw parameter space and run the experiments for multiple trials and select the best design with the highest overall success rate as the final design.
- 2) *CMA-ES*: We use the covariance matrix adaptation evolutionary strategy (CMA-ES) [12] approach and select the design of the highest overall success rate.
- 3) *BO w/ Raw Parameters* (BO): We use the same BO algorithm as our final model, but optimization runs in the original parameter space as opposed to the learned latent space.

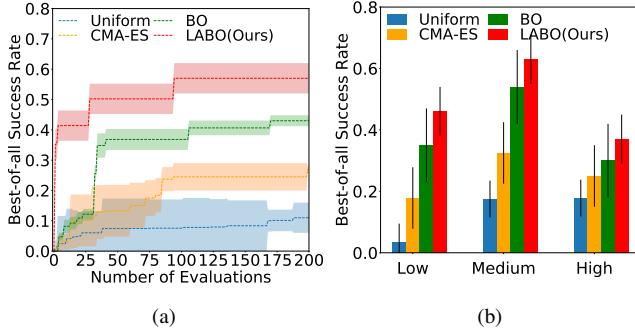


Fig. 5: (a) Best-of-all grasping success rates of the optimal hand designs on the training data set of objects; (b) Grasping success rates on objects of different geometric complexities.

Table I reports the per-task and overall success rate. In terms of the final optimized hand cost values, they are 2.81 ± 1.12 , 2.88 ± 0.19 , 2.63 ± 0.14 , and 1.83 ± 0.67 for Uniform, CMA-ES, BO and LABO, respectively. For all the methods, we sample the same number of data points. The results show that LABO outperforms other baselines on all three types of grasps with the lowest cost of the final hand design. The closest baseline is BO w/ Raw Parameters. On the other hand, the Uniform Sampling and CMA-ES baselines fail to produce robust hand designs, resulting in more than 20% lower success rate than ours and increased costs. Qualitatively speaking, the hand with a higher cost usually have redundant finger segments that are not functioning. The baselines can still find some successful hand designs but fail to remove these non-functioning fingers or segments due to the high dimensionality of the optimization problem. Optimizing in the latent space using LABO effectively find the important factors for design and can find more cost-effective hands.

Figure 5a shows the success rate curves of the all methods over the course of optimization. We found that the our model not only achieved the highest final performance, but also outperformed all baselines with less number of evaluations, and to achieve a success rate of 40%, it is around 100 iterations faster than BO with raw parameters. It implies that the learned latent representations facilitate efficient search of promising samples and knowledge sharing across different design parameters, reducing the sample complexity of the Bayesian Optimization process. We further analyze the impact of object geometry on the grasp performance, and show an ablation study of the success rates of our optimal hand on these tasks with respect to the object geometric complexity in Figure 5b. The geometric complexity is measured by the number of vertices in the 3D mesh models of the objects, where object models of less than 5k vertices are classified as *low* and object models of greater than 7k vertices *high*, and *medium* otherwise. The results show that objects of medium complexity are easiest to grasp. We hypothesize that objects of low complexity, such as spheres, might lack flat surface for stable grasps, and those of high complexity, such as flower shapes, might be difficult to reach contact points. Figure 1 shows the qualitative examples of the optimized hands from our model and the grasping behaviors on objects of the

TABLE I: Success Rate of Optimized Hands on Novel Objects

Methods	Power	Pinch	Lateral	Overall
Uniform	0.13 ± 0.08	0.03 ± 0.05	0.08 ± 0.05	0.07 ± 0.05
CMA-ES [12]	0.19 ± 0.20	0.14 ± 0.09	0.45 ± 0.40	0.26 ± 0.03
BO [13]	0.24 ± 0.12	0.15 ± 0.14	0.72 ± 0.31	0.37 ± 0.19
LABO	0.37 ± 0.19	0.21 ± 0.15	0.92 ± 0.10	0.50 ± 0.12

TABLE II: Ablations Different Numbers of Fingers (Success Rate)

# Fingers	Power	Pinch	Lateral	Overall
2	0.07 ± 0.04	0.01 ± 0.01	0.53 ± 0.02	0.20 ± 0.01
3	0.37 ± 0.07	0.24 ± 0.02	0.97 ± 0.06	0.52 ± 0.04
4	0.38 ± 0.24	0.22 ± 0.18	0.71 ± 0.27	0.44 ± 0.09
5	0.33 ± 0.19	0.20 ± 0.14	0.79 ± 0.35	0.44 ± 0.17
6	0.31 ± 0.24	0.25 ± 0.15	0.95 ± 0.07	0.50 ± 0.13

three grasp types. The optimized hand developed different strategies for grasping objects of various shapes from the three grasp types. With the morphological costs as part of the optimization objective, it constrained the complexity of the design and landed on a three-finger hand.

a) *Ablations on the Hand Complexity*: In Table II, we show the result of the per-task grasping success rate, overall success rate for the optimized hand by fixing the number of fingers. It shows that simple hand with only 2 fingers can hardly succeed at the grasping tasks. We hypothesize that two-fingers hand is hard to form a tight force-closure and these hands are vulnerable to adversarial perturbations. With three or four fingers, the hand are becoming more robust to external perturbations while still keep the morphology cost-effective. Increased number of fingers cause a significantly increased overall success rate variance. This might be that with more fingers to design, there are more choices to make and less easy to make consistent designs.

V. CONCLUSION

In this work, we explore the emergence of embodiment and behavior of a robot hand to fulfill grasping tasks. Our goal is to discover robust and cost-efficient hand morphologies capable of grasping a diverse set of objects. To combat the challenge of searching in the high-dimensional design space, we integrate the learning of latent representations with scalable Bayesian optimization algorithms. It validates our hypothesis that morphology and control can arise as emergent phenomena for the purpose of performing tasks. While our optimization method is general, two notable limitations could hinder its real-world use for designing robot hands. First, the optimization objective does not consider all practical constraints and costs of mechanical design. Our model simplified important factors, such as actuation limits, fabrication costs, and energy efficiency, when evaluating the design samples. Second, using a physical simulator for grasp trials might result in vastly different outcomes than real-world experiments. A careful quantification of the simulation quality is necessary to ensure sim-to-real transfer of the grasp behaviors. For future work, we would like to explore the learning of vision-based closed-loop grasp policies in replacement of the open-loop controller. We also hope to take into account hardware constraints and the reality gap in order to extend our method to real-world robot designs.

ACKNOWLEDGMENT

We gratefully acknowledge the feedback from members in NVIDIA AI Algorithms research team. We also acknowledge Jonathan Tremblay from NVIDIA for the support on ViSII rendering.

REFERENCES

- [1] C. Mautner and R. K. Belew, “Evolving robot morphology and control,” *Artificial Life and Robotics*, vol. 4, no. 3, pp. 130–136, 2000.
- [2] R. W. Young, “Evolution of the human hand: the role of throwing and clubbing,” *Journal of Anatomy*, vol. 202, no. 1, pp. 165–174, 2003.
- [3] J. Mahler, M. Matl, X. Liu, A. Li, D. Gealy, and K. Goldberg, “Dex-net 3.0: Computing robust robot vacuum suction grasp targets in point clouds using a new analytic model and deep learning,” *arXiv preprint arXiv:1709.06670*, 2017.
- [4] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, “Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics,” *arXiv preprint arXiv:1703.09312*, 2017.
- [5] L. Shao, F. Ferreira, M. Jorda, V. Nambiar, J. Luo, E. Solowjow, J. A. Ojea, O. Khatib, and J. Bohg, “Unigrasp: Learning a unified model to grasp with n-fingered robotic hands,” *ArXiv*, vol. abs/1910.10900, 2019.
- [6] R. Pfeifer and G. Gómez, “Morphological computation—connecting brain, body, and environment,” in *Creating brain-like intelligence*. Springer, 2009, pp. 66–83.
- [7] S. B. Backus and A. M. Dollar, “An adaptive three-fingered prismatic gripper with passive rotational joints,” *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 668–675, 2016.
- [8] R. Deimel and O. Brock, “A novel type of compliant and underactuated robotic hand for dexterous grasping,” *The International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 161–185, 2016.
- [9] C. Eppner, S. Höfer, R. Jonschkowski, A. Sieverling, V. Wall, and O. Brock, “Lessons from the amazon picking challenge: Four aspects of building robotic systems,” in *h International Joint Conference on Artificial Intelligence*, 2017.
- [10] S. Yuan, A. D. Epps, J. B. Nowak, and J. K. Salisbury, “Design of a roller-based dexterous hand for object grasping and within-hand manipulation,” in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2020.
- [11] T. Feix, J. Romero, H.-B. Schmidtmayer, A. M. Dollar, and D. Kragic, “The GRASP taxonomy of human grasp types,” *IEEE Transactions on human-machine systems*, vol. 46, no. 1, pp. 66–77, 2015.
- [12] N. Hansen, S. D. Müller, and P. Koumoutsakos, “Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es),” *Evolutionary computation*, vol. 11, no. 1, pp. 1–18, 2003.
- [13] P. I. Frazier, “A tutorial on bayesian optimization,” *arXiv preprint arXiv:1807.02811*, 2018.
- [14] S. Seok, A. Wang, M. Y. Chuah, D. J. Hyun, J. Lee, D. M. Otten, J. H. Lang, and S. Kim, “Design principles for energy-efficient legged locomotion and implementation on the mit cheetah robot,” *Ieee/asmé transactions on mechatronics*, vol. 20, no. 3, pp. 1117–1129, 2014.
- [15] K. Graichen, S. Hentzelt, A. Hildebrandt, N. Kärcher, N. Gaißert, and E. Knubben, “Control design for a bionic kangaroo,” *Control Engineering Practice*, vol. 42, pp. 106–117, 2015.
- [16] K. Jayaram, J. Shum, S. Castellanos, E. F. Helbling, and R. J. Wood, “Scaling down an insect-size microrobot, hamr-vi into hamr-jr,” *arXiv preprint arXiv:2003.03337*, 2020.
- [17] C. Leger *et al.*, *Automated synthesis and optimization of robot configurations: an evolutionary approach*, 1999.
- [18] N. Cheney, R. MacCurdy, J. Clune, and H. Lipson, “Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding,” in *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, 2013, pp. 167–174.
- [19] K. S. Luck, H. B. Amor, and R. Calandra, “Data-efficient co-adaptation of morphology and behaviour with deep reinforcement learning,” in *Conference on Robot Learning*, 2020, pp. 854–869.
- [20] K. Wampler and Z. Popović, “Optimal gait and form for animal locomotion,” *ACM Transactions on Graphics (TOG)*, vol. 28, no. 3, pp. 1–8, 2009.
- [21] S. Ha, S. Coros, A. Alspach, J. Kim, and K. Yamane, “Joint optimization of robot design and motion parameters using the implicit function theorem,” in *RSS*, 2017.
- [22] A. Spielberg, A. Zhao, Y. Hu, T. Du, W. Matusik, and D. Rus, “Learning-in-the-loop optimization: End-to-end control and co-design of soft robots through learned deep latent representations,” in *Advances in Neural Information Processing Systems* 32, 2019.
- [23] Y. Hu, J. Liu, A. Spielberg, J. B. Tenenbaum, W. T. Freeman, J. Wu, D. Rus, and W. Matusik, “Chainqueen: A real-time differentiable physical simulator for soft robotics,” in *International Conference on Robotics and Automation (ICRA)*, 2019.
- [24] T. Chen, Z. He, and M. Ciocarlie, “Hardware as policy: Mechanical and computational co-optimization using deep reinforcement learning,” *arXiv preprint arXiv:2008.04460*, 2020.
- [25] T. Wang, Y. Zhou, S. Fidler, and J. Ba, “Neural graph evolution: Towards efficient automatic robot design,” *arXiv preprint arXiv:1906.05370*, 2019.
- [26] D. Pathak, C. Lu, T. Darrell, P. Isola, and A. A. Efros, “Learning to control self-assembling morphologies: a study of generalization via modularity,” in *Advances in Neural Information Processing Systems*, 2019, pp. 2295–2305.
- [27] T. Chen, A. Murali, and A. Gupta, “Hardware conditioned policies for multi-robot transfer learning,” in *Advances in Neural Information Processing Systems*, 2018, pp. 9355–9366.
- [28] W. Huang, I. Mordatch, and D. Pathak, “One policy to control them all: Shared modular policies for agent-agnostic control,” in *ICML*, 2020.
- [29] J. Bohg, A. Morales, T. Asfour, and D. Kragic, “Data-driven grasp synthesis—a survey,” *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, 2013.
- [30] I. Lenz, H. Lee, and A. Saxena, “Deep learning for detecting robotic grasps,” *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.
- [31] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, “Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection,” *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421–436, 2018.
- [32] L. Pinto and A. Gupta, “Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.
- [33] K. Fang, Y. Zhu, A. Garg, A. Kuryenkov, V. Mehta, L. Fei-Fei, and S. Savarese, “Learning task-oriented grasping for tool manipulation from simulated self-supervision,” *Robotics: Science and Systems (RSS)*, 2018.
- [34] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige, *et al.*, “Using simulation and domain adaptation to improve efficiency of deep robotic grasping,” in *IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 4243–4250.
- [35] U. Viereck, A. t. Pas, K. Saenko, and R. Platt, “Learning a visuomotor controller for real world robotic grasping using simulated depth images,” *arXiv preprint arXiv:1706.04652*, 2017.
- [36] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, *et al.*, “Learning dexterous in-hand manipulation,” *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020.
- [37] A. Gupta, C. Eppner, S. Levine, and P. Abbeel, “Learning dexterous manipulation for a soft robotic hand from human demonstrations,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 3786–3793.
- [38] J. Dhamala, S. Ghimire, J. L. Sapp, B. M. Horáček, and L. Wang, “High-dimensional bayesian optimization of personalized cardiac model parameters via an embedded generative model,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2018, pp. 499–507.
- [39] R.-R. Griffiths and J. M. Hernández-Lobato, “Constrained bayesian optimization for automatic chemical design,” *arXiv preprint arXiv:1709.05501*, 2017.
- [40] J. Napier, “The evolution of the hand,” *Scientific American*, 1962.
- [41] E. Van Hen ten, D. Van’t Slot, C. Hol, and L. Van Willigenburg, “Optimal manipulator design for a cucumber harvesting robot,” *Computers and electronics in agriculture*, vol. 65, no. 2, pp. 247–257, 2009.
- [42] D. P. Kingma and M. Welling, “An introduction to variational autoencoders,” *arXiv preprint arXiv:1906.02691*, 2019.
- [43] M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wil-

- son, and E. Bakshy, “Botorch: Programmable bayesian optimization in pytorch,” *arXiv preprint arXiv:1910.06403*, 2019.
- [44] E. Coumans and Y. Bai, “Pybullet, a python module for physics simulation in robotics, games and machine learning,” 2017.
- [45] D. Morrison, P. Corke, and J. Leitner, “Egad! an evolved grasping analysis dataset for diversity and reproducibility in robotic manipulation,” *IEEE Robotics and Automation Letters*, 2020.