arXiv:2609.21138v1 [cs.RO] 17 Sep 2026

Diverse and Adaptable Arm Coordination for Octopus-Crawling via Diffusion-Based Uncertainty-Aware Optimization

Seung Hyun Kim[1], Heng-Sheng Chang[*,1,2], Kimia Kazemi[1], Prashant G. Mehta[1,2], Mattia Gazzola[1,3,4]

***Abstract*— Octopus crawling motivates soft robots that exploit redundancy, yet discovering and organizing diverse coordination modes for adaptation remains challenging. To address this, we introduce a Diffusion-based Uncertainty-aware Optimization (DUO) algorithm that learns demonstration-free crawling controllers for a simulated, muscle-actuated *CyberOctopus*. This work represents the first application of diffusion-based control to soft multi-arm robots in contact-rich simulations. By embedding a variety of locomotion behaviors within a shared control distribution, this approach enables the simulated octopus to navigate dynamic physical constraints, demonstrating that learned coordination diversity inherently facilitates robust adaptation. The main contributions include: (i) a symmetry-structured policy representation that folds radially equivalent controllers into a canonical directional sector, (ii) an online black-box optimization strategy, the DUO algorithm, that discovers and retains diverse coordination modes, and (iii) a control editing technique that adapts existing controllers to novel actuator constraints without retraining. These results show how learned coordination diversity makes motor abundance a practical resource for adaptation in soft multi-arm robots.**

I. INTRODUCTION

Octopus crawling offers both inspiration and a challenge for soft-arm robotics. Through body compliance, distributed muscular actuation, and coordinated recruitment of redundant arms [1], [2], octopuses crawl without a fixed gait or rhythmic motor pattern [3], [4]. They appear to exploit their excess physical degrees of freedom as a resource for diverse behaviors and survival in natural environments [5], [6]. This interplay between motor abundance and compliant body mechanics [7] motivates supporting a range of coordination strategies for adaptation under unexpected actuation and environmental constraints.

Realizing this requires exploring a high-dimensional control space, an already challenging endeavor, which is further complicated by the nonlinear mechanics and contact dynamics of soft arms. Existing approaches pursue behavioral diversity through explicit archives [8], [9], novelty-driven exploration [10], and hierarchical structures [11], [12]. As system dimensionality scales, engineering these behavioral descriptors becomes a major bottleneck [13]. The central challenge, therefore, is to learn an implicit representation that natively encapsulates behavioral diversity without requiring explicit parameterization.

[1]Mechanical Science and Engineering, [2]Coordinated Science Laboratory, [3]Carl R. Woese Institute for Genomic Biology, [4]National Center for Supercomputing Applications, University of Illinois Urbana-Champaign. [*]CA email: hschang@illinois.edu. Code repository will be available soon.

This work is supported by the Office of Naval Research (N00014–22–1–2569), and NSF Expedition "Mind in Vitro" award (IIS-2123781), and used TACC Frontera HPC allocation at the Texas Advanced Computing Center through Allocation (IBN22011).

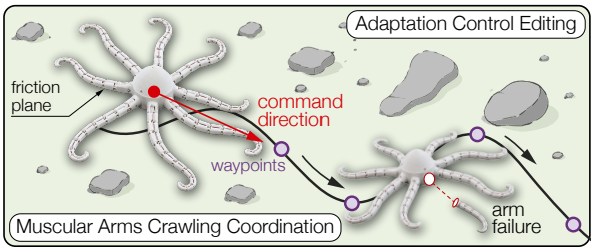

Fig. 1: **CyberOctopus.** The eight-arm soft-body model crawls on a frictional plane following waypoint sequences, requiring coordinated distributed actuation in a hyper-redundant control space. A diffusion model is chosen to handle a variety of feasible coordination modes, and the trained model can then be leveraged for adaptive control editing when an arm fails or surface friction changes, without requiring retraining.

To learn such a representation directly from data [14], we use a conditional diffusion model that captures a multimodal distribution within a shared latent space [15]. While such models usually demand extensive prior datasets, this limitation is bypassed by constructing a dataset dynamically via online black-box (simulator-based) optimization [16]. However, evaluating candidate controls in simulation is costly; therefore, we guide our exploration of the control space by integrating predictive uncertainty [17], [18], ensuring our limited budget is spent testing only the informative candidates. This acquisition strategy underpins our Diffusion-based Uncertainty-aware Optimization (DUO) algorithm.

We evaluate the proposed algorithm using a simulated muscular *CyberOctopus* (Figure 1) with eight radially arranged continuum arms [11] and distributed muscle actuation [19], [20]. Because no prior demonstration data exists for this complex morphology, we rigorously test the algorithm's ability to learn full-direction crawling entirely from dynamic simulation. We then assess the robustness of the resulting controller by subjecting the system to previously unseen physical constraints, such as actuator failures or sudden changes in surface friction, to evaluate its capacity for adaptive locomotion. Our contributions are threefold:

- **Symmetry folding.** We reduce the search direction to a single forward canonical sector by exploiting the system's rotational symmetry, utilizing interpolation and boundary evaluation to enable full-direction crawling.
- **Diversity-oriented search.** We introduce the DUO algorithm, which actively guides exploration to discover and retain a diverse family of effective coordination modes within a limited simulation budget.
- **Constrained reuse.** We demonstrate zero-shot adaptation through control editing. By leveraging the retained coordination modes, the system successfully recovers locomotion under novel physical limitations without requiring any retraining of the diffusion model.

II. RELATED WORK

Controller discovery for soft multi-arm crawling can be formulated as black-box optimization, with the locomotion performance objective obtained through costly simulation evaluations. The redundancy of multi-arm systems further motivates discovering distinct coordination patterns that achieve comparable performance. Accordingly, we review quality–diversity methods for realizing diverse, high-performing solutions and diffusion-based optimization methods for representing multimodal solution distributions.

Quality–diversity (QD). These methods seek collections of high-performing solutions across a *behavior space* defined by descriptors [8], [9]. Classical approaches use handcrafted descriptors to organize an archive; grid-based archives retain the highest-performing solution in each occupied cell. This organization supports the discovery and reuse of behavioral repertoires in robot learning and adaptation [12].

The choice of descriptors dictates which behavioral variations are preserved within the archive. For multi-arm crawling, final-displacement descriptors can group distinct coordination patterns into the same cell, while detailed arm-deformation descriptors increase representation and archive–design complexity. AutoQD [13] addresses the need for manual descriptor design by embedding policy occupancy measures and projecting them into a low-dimensional behavior space for QD optimization. These learned descriptors guide archive-based search toward behaviorally distinct solutions without predefined behavioral descriptors.

Diffusion-based optimization. Diffusion models learn complex, multimodal distributions through iterative denoising [21], enabling expressive action distributions in robotic control [15]. In optimization, Diffusion Evolution [22] connects denoising with evolutionary optimization, while DDOM [16] learns objective-conditioned distributions from fixed datasets. Building on these offline methods, online extensions can actively evaluate candidates during search.

DiffBBO [17] and DiBO [18] alternate candidate generation, evaluation, and model updates for online black-box optimization. DiffBBO uses ensemble uncertainty to select target objective values for conditional generation, whereas DiBO combines surrogate-guided posterior inference with local refinement. These approaches highlight the importance of acquisition design under limited evaluation budgets: emphasizing predicted performance can concentrate search, while exploration is needed to discover alternative solutions.

III. PROBLEM FORMULATION

We formulate planar crawling control as an optimization problem: discover a family of controls that achieves stable crawling across the full range of directions, while penalizing net body rotation to preserve heading alignment.

Control evaluation. Let $u \in \mathcal{U}^M$ denote a control for a robot with M arms (Figure 2b). The control to the m-th arm is $u^{(m)} \in \mathcal{U} = [-1,1]^U$, and

$$u := \left[u^{(1)}, \ldots, u^{(M)}\right] \in \mathcal{U}^M = [-1,1]^{M\times U}.$$

Applying a control u yields a body rotation $\Delta\Phi(u)$ and a planar displacement $\Delta\mathbf{p}(u) = [\Delta x(u), \Delta y(u)]$ with $\vartheta(u) = \operatorname{arctan2}(\Delta y(u), \Delta x(u))$. For a commanded direction $\phi \in [0, 2\pi]$, the crawling performance is quantified by

$$\begin{aligned}
\mathsf{R}_{\mathrm{rad}}(u;\phi) &:= \cos\phi\,\Delta x(u) + \sin\phi\,\Delta y(u),\\
\mathsf{J}_{\mathrm{ang}}(u;\phi) &:= \left|\operatorname{arctan2}(\sin(\vartheta(u)-\phi), \cos(\vartheta(u)-\phi))\right|,\\
\mathsf{J}_{\mathrm{rot}}(u) &:= \left|\Delta\Phi(u)\right|.
\end{aligned}$$

$\mathsf{R}_{\mathrm{rad}}$ is progress along the direction ϕ, $\mathsf{J}_{\mathrm{ang}}$ is the angular deviation, and $\mathsf{J}_{\mathrm{rot}}$ is the magnitude of the body rotation.

Control discovery as an optimization. For a fixed commanded direction ϕ, control discovery is formulated as optimization of the following objective:

$$\max_{u\in\mathcal{U}^M} \mathsf{R}_{\mathrm{rad}}(u;\phi) - \frac{\lambda}{2}\left[\mathsf{J}_{\mathrm{ang}}(u;\phi) + \mathsf{J}_{\mathrm{rot}}(u)\right], \qquad (1)$$

where $\lambda > 0$ scales the angular penalties into the displacement units of $\mathsf{R}_{\mathrm{rad}}$. The objective rewards translation along the commanded direction ϕ while discouraging directional error and net body rotation. While incorporating commanded body rotation is a natural extension, limiting the formulation to pure translation provides a clearer baseline for evaluating crawling performance. Thus, we scope our study to translational motion and defer rotational control to future work.

IV. METHOD

A. Radial Symmetry Reduction

Rotational equivariance. Solving (1) independently for every ϕ would require a continuum of direction-specific searches. We instead use the rotational symmetry of the robot to reduce the search space. Let $\triangle\phi := {}^{2\pi}/_{M}$ be the angle between adjacent arms, P the cyclic permutation:

$$P := \begin{bmatrix} \mathbf{0} & 1 \\ I_{M-1} & \mathbf{0} \end{bmatrix},\ u_{\gamma\triangle\phi} := P^{\gamma}u,\ \forall\gamma \in \{0,1,\ldots,M-1\}.$$

Thus, P maps $[u^{(1)},\ldots,u^{(M)}]$ to $[u^{(M)},u^{(1)},\ldots,u^{(M-1)}]$ and satisfies $P^M = I_M$ and $P^{-1} = P^{M-1}$. The shifted control $u_{\gamma\triangle\phi}$ rotates a control assignment u by γ arms.

Exploiting this symmetry, the permutation rotates crawling direction while preserving direction-relative performance:

$$\begin{aligned}
\mathsf{R}_{\mathrm{rad}}(u_{\gamma\triangle\phi};\phi-\gamma\triangle\phi) &= \mathsf{R}_{\mathrm{rad}}(u;\phi),\\
\mathsf{J}_{\mathrm{ang}}(u_{\gamma\triangle\phi};\phi-\gamma\triangle\phi) &= \mathsf{J}_{\mathrm{ang}}(u;\phi), \quad \forall\gamma\in\{0,\ldots,M-1\}.\\
\mathsf{J}_{\mathrm{rot}}(u_{\gamma\triangle\phi}) &= \mathsf{J}_{\mathrm{rot}}(u),
\end{aligned}$$

It is therefore sufficient to search the canonical sector $[-\triangle\phi/2, \triangle\phi/2]$; the other sectors follow by cyclic permutation. This gives exact controls at the discrete symmetry directions, and intermediate directions require interpolation.

Full-direction control. We write the commanded direction as $\phi = \gamma\triangle\phi + \delta$ with $\gamma \in \{0,\ldots,M-1\}$ and $\delta \in [-\triangle\phi/2, \triangle\phi/2]$, blending the adjacent arm assignments using

$$u_\phi := A_\delta P^\gamma u, \quad A_\delta := \left(1 - \tfrac{|\delta|}{\triangle\phi}\right) I_M + \tfrac{|\delta|}{\triangle\phi} P^{\operatorname{sgn}(\delta)}.$$

Because both P and A_δ map $\mathcal{U}^M$ to itself, u_ϕ remains a valid control. This construction produces a directional control based on one canonical control for all $\phi \in [0, 2\pi]$.

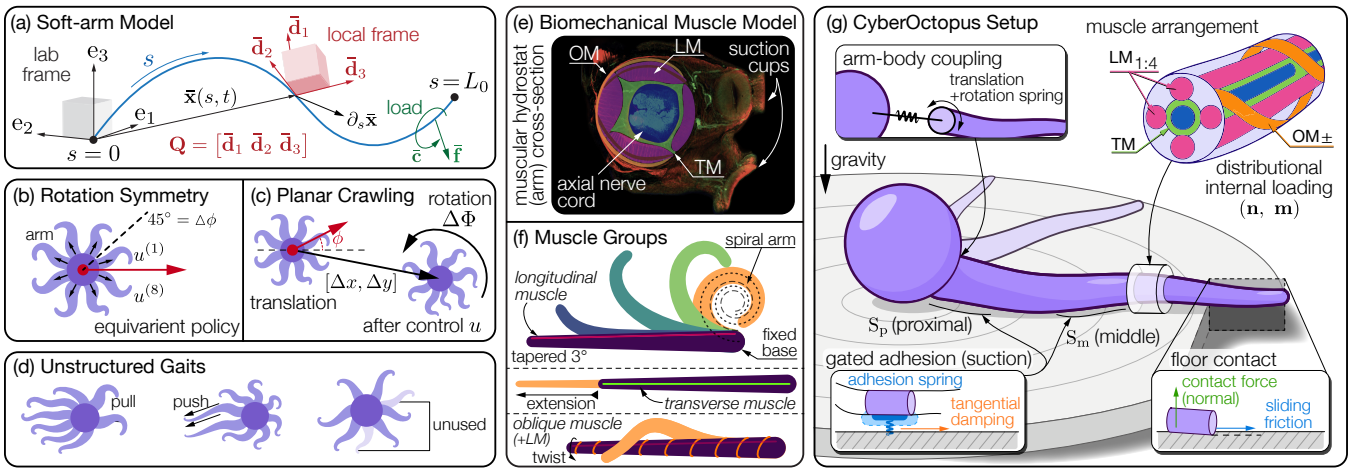

Fig. 2: **Cyber-octopus model and actuation.** (a) Cosserat-rod soft-arm model. (b) Eightfold radial symmetry. (c) Control-induced translation and body rotation. (d) Arm coordination through pulling, pushing, and alternating recruitment. (e) Muscular-hydrostat cross-section: longitudinal (LM), transverse (TM), and oblique (OM) muscles, axial nerve cord, and suction cups. (f) Muscle-driven bending (LM), extension (TM), and twisting (OM and LM); tapered arms and muscle paths follow a spiral construction [23]. (g) Muscle arrangement, arm-body coupling, internal loading, and floor contact. Within the capture distance, sucker adhesion attracts the arm to the floor while tangential damping suppresses slip.

Boundary-robust objective. Rotational equivariance does not determine performance between symmetry-related directions. We therefore evaluate each control both at the canonical direction $\delta = 0$ and at $\delta = {}^{\triangle\phi}/_{2}$, the boundary of the canonical sector. The boundary at the other side $-{}^{\triangle\phi}/_{2}$ is equivalent by the cyclic symmetry, so one additional evaluation suffices.

Define the canonical and boundary radial objectives as

$$f_0(u) := \mathsf{R}_{\mathrm{rad}}(u;0), \quad f_{\mathrm{b}}(u) := \mathsf{R}_{\mathrm{rad}}\left(A_{-\triangle\phi/2}u;\, {}^{\triangle\phi}/_{2}\right).$$

The objective function combines these terms and applies the directional and rotational penalties at both the canonical and boundary directions:

$$\begin{aligned} f(u) := [f_0(u) + f_{\mathrm{b}}(u)] - \frac{\lambda}{2}\left[\mathsf{J}_{\mathrm{ang}}(u;0) + \mathsf{J}_{\mathrm{rot}}(u)\right] \\ - \frac{\lambda}{2}\left[\mathsf{J}_{\mathrm{ang}}(A_{-\triangle\phi/2}u;\, {}^{\triangle\phi}/_{2}) + \mathsf{J}_{\mathrm{rot}}(A_{-\triangle\phi/2}u)\right] \end{aligned} \tag{2}$$

with $\lambda > 0$ as the angular penalty weight. Weighting the canonical and boundary terms equally prevents an optimized control from trading boundary performance for center canonical direction performance. The reformulated optimization problem (1) with the radial-symmetry reduction becomes

$$\max_{u \in \mathcal{U}^M} \; f(u), \tag{3}$$

where $y = f(u)$ is the objective value for control $u \in \mathcal{U}^M$.

Remark 1. *Radial symmetry reduces the search dimensionality to a single canonical sector. However, solving* (3) *still presents two challenges:*

1) *The remaining high-dimensional actuation space $\mathcal{U}^M$.*
2) *The high computational cost of objective f evaluation.*

We address these challenges in the following subsection.

B. Diffusion-based Uncertainty-aware Optimization (DUO)

Diffusion-based inverse model. To address the challenge of searching in a high-dimensional control space $\mathcal{U}^M \cong [-1,1]^d$ with $d = MU$, we propose candidate controls by sampling from a trained inverse model $p_\theta : \mathbb{R} \to \mathcal{P}(\mathcal{U}^M)$, a conditional distribution that predicts control distributions given an objective value from $f : \mathcal{U}^M \to \mathbb{R}$. For diffusion modeling, we identify each control u with its vectorization $\mathrm{vec}(u)$, and reshape generated samples before simulation.

A denoising diffusion probabilistic model (DDPM) [21] learns the conditional distribution $p_\theta(\cdot \mid y)$, where θ denotes its trainable parameters and y the requested objective value of f. For a clean control $u_0 \in [-1,1]^d$, the diffusion process produces a sequence of noisy controls $u_1, \ldots, u_K$ through

$$u_k = \mathrm{clip}\left(\sqrt{\bar{a}_k}\, u_0 + \sqrt{1-\bar{a}_k}\, \epsilon_k\right), \quad \epsilon_k \sim \mathcal{N}(0, I_d),$$

where $k \in \{1, \ldots, K\}$ indexes the K diffusion steps, $\bar{a}_k \in (0,1)$ is the monotonically decreasing cumulative signal-retention factor prescribed by the noise variance schedule. The clipping operator restricts the noisy control to $[-1,1]^d$. The DDPM trains a noise predictor ϵ_θ minimizing

$$\mathcal{L}(\theta; \mathcal{D}) := \mathbb{E}\left[\|\epsilon - \epsilon_\theta(u_k, y, k)\|_2^2\right], \tag{4}$$

where the expectation is over $(u_0, y) \sim \mathrm{Unif}(\mathcal{D})$, a uniform step k, and Gaussian noise ϵ. Here, $\mathcal{D}$ contains the evaluated control-objective pairs $(u, f(u))$.

For a fixed target objective y, sampling begins with $u_K \sim \mathcal{N}(0, I_d)$ and proceeds through reverse steps $k = K, \ldots, 1$. At each step k, the denoiser computes the conditional mean, and the next control is sampled as [21]

$$\bar{u}_{\theta,k} = \sqrt{\frac{\bar{a}_{k-1}}{\bar{a}_k}}\left[u_k - \frac{\tilde{u}_k^2}{\sqrt{1-\bar{a}_k}}\epsilon_\theta(u_k, y, k)\right],$$

$$u_{k-1} = \mathrm{clip}\left(\bar{u}_{\theta,k} + \tilde{u}_k \epsilon_k\right), \quad \tilde{u}_k = \sqrt{1 - \frac{\bar{a}_k}{\bar{a}_{k-1}}},$$

where fresh noise vectors $\epsilon_k \sim \mathcal{N}(0, I_d)$ are drawn independently of one another for $k > 1$, and $\epsilon_1 = 0$ makes the final step return the denoised mean, following Algorithm 2 of [21]. After K reverse steps, the final control $u = u_0$ is a sample from $p_\theta(\cdot \mid y)$. Resampling with fresh noise yields alternative controls for the same target objective y.

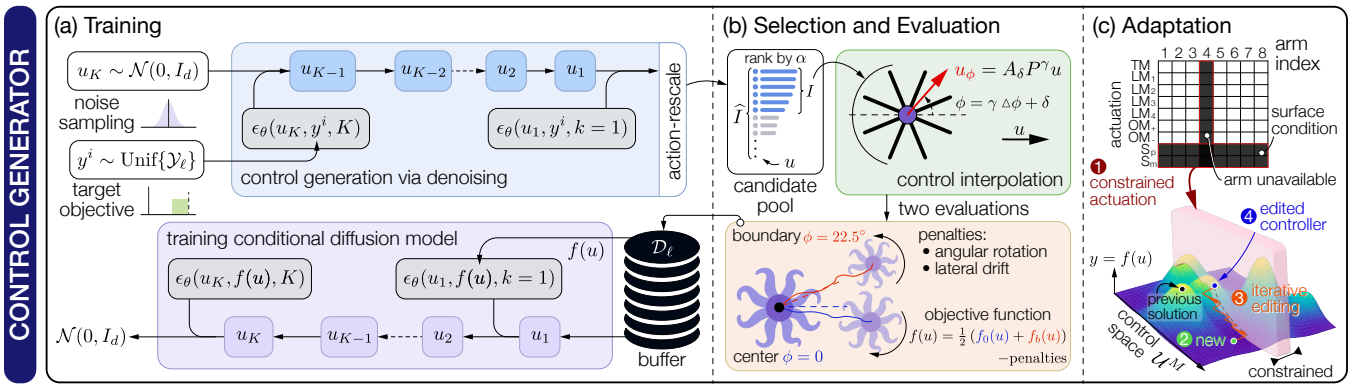

Fig. 3: **Control discovery and constrained reuse.** (a) Control generation and denoiser training. (b) Candidate selection and evaluation. Candidates are ranked by acquisition score α, and the top I candidates are evaluated. Simulating the crawling motion in both the center and boundary directions yields the objective values for the next iteration. (c) Control reuse by editing. Shaded grid entries illustrate constrained actuation coordinates. The landscape depicts a schematic objective, and the pink hyperplane denotes the constrained control subspace. Through repeated noising and denoising, a previous control can be adjusted to satisfy the constraints while improving performance, without retraining.

Algorithm 1: **Diffusion-based Uncertainty-aware Optimization (DUO)**

Input: objective f; initial dataset $\mathcal{D}_0$; iteration count L; model number E, J; batch size I; proposal count $\widehat{I} > I$
Output: trained conditional generator $p_\theta(u \mid y)$

for $\ell = 0, \dots, L-1$ **do**
  Update inverse function generators with $\mathcal{D}_\ell$:
  train diffusion models $\{p_{\theta_e}(u \mid y)\}_{e=1}^{E}$ with (4)
  Update forward objective estimators with $\mathcal{D}_\ell$:
  train surrogates $\{\hat{f}_{\psi_j}(u)\}_{j=1}^{J}$ with MSE regression
  Sample candidate targets $\{y^i \sim \mathrm{Unif}\{\mathcal{Y}_\ell\}\}_{i=1}^{\widehat{I}}$
  Sample $\{u^i \sim p_{\theta_e}(u \mid y^i) : e \sim \mathrm{Unif}\{1, \dots, E\}\}_{i=1}^{\widehat{I}}$
  Rank $\{(u^i, y^i)\}_{i=1}^{\widehat{I}}$ using acquisition score (6)
  Retain and relabel the top candidates as $\{u^i\}_{i=1}^{I}$
  Evaluate $\{f(u^i)\}_{i=1}^{I}$ with the dynamic simulator
  $\mathcal{D}_{\ell+1} \leftarrow \mathcal{D}_\ell \cup \{(u^i, f(u^i))\}_{i=1}^{I}$
end for
$p_\theta(u \mid y) \leftarrow p_{\theta_e}(u \mid y)$, $e \sim \mathrm{Unif}\{1, \dots, E\}$

This inverse model replaces direct search in (3) with target-selection to maximize expected control performance:

$$\max_{y \in \mathcal{Y}} \ \mathbb{E}_{u \sim p_\theta(\cdot|y)}[f(u)], \tag{5}$$

where $\mathcal{Y} \subseteq \mathbb{R}$ is the candidate target objective set. The design of $\mathcal{Y}$ and the remaining hyperparameters is specified in Section V.

Solving (5) requires a representative dataset $\mathcal{D}$. When no pre-existing demonstration data is available, the diffusion model must be trained by iteratively evaluating proposed controls in simulation and updating the dataset with the resulting objective values as $\mathcal{D}_{\ell+1} = \mathcal{D}_\ell \cup \{(u^i, f(u^i))\}_{i=1}^{I}$ with I new evaluations at iteration ℓ. However, evaluating the objective function f is computationally expensive, and therefore simulating every control candidate generated by the inverse model is intractable. To efficiently allocate the evaluation budget, we employ an acquisition strategy that uses predicted performance to select a limited, informative subset of candidates $\mathcal{Y}_\ell$ for simulation.

Uncertainty-aware batch acquisition. To address the challenge of the costly dynamic simulation, we extract a highly informative evaluation batch from a large pool of generated proposals. Candidates are scored by combining their target reliability, estimated via the epistemic uncertainty of a diffusion model ensemble, with their predicted control performance, evaluated through an ensemble of surrogate models of objective function f. We then simulate only the top-ranked batch and append the resulting ground-truth evaluations to the dataset for the next optimization iteration.

Following DiffBBO [17], we train E independently initialized diffusion models with parameters $\{\theta_e\}_{e=1}^{E}$. For $e \sim \mathrm{Unif}\{1, \dots, E\}$, define the target epistemic uncertainty:

$$\Delta_{\mathrm{epi}}(y; \mathcal{D}_\ell) := \mathrm{Var}_e\left[\mathbb{E}_{u \sim p_{\theta_e}(\cdot|y)}\left[\frac{\|u\|_2}{\sqrt{d}}\right]\right].$$

The inner expectation estimates one model's candidate norm, and the outer variance measures disagreement among models. The logarithmic implementation of uncertainty-aware exploration (UaE) [17] assigns the target score

$$\alpha^{\mathrm{target}}(y; \mathcal{D}_\ell) := \log \frac{\bar{y}}{\Delta_{\mathrm{epi}}(y; \mathcal{D}_\ell)}, \quad \bar{y} := \frac{y - y_{\min,\ell}}{y_{\max,\ell} - y_{\min,\ell}},$$

where $y_{\min,\ell}$ and $y_{\max,\ell}$ are the minimum and maximum objective values in $\mathcal{D}_\ell$. This score α^{target} favors high target objectives with low ensemble model disagreement.

To distinguish control candidates u generated at the same target objective value y, we complement target reliability α^{target} with predictions of individual control performance $\alpha^{\mathrm{control}}$. Following DiBO [18], we train J independently initialized neural-network surrogates $\{\hat{f}_{\psi_j}\}_{j=1}^{J}$ by mean-squared-error regression on $\mathcal{D}_\ell$. Here, ψ_j denotes the trainable parameters of surrogate $\hat{f}_{\psi_j}$, whose output estimates f. For $j \sim \mathrm{Unif}\{1, \dots, J\}$, define the predictive moments

$$\mu(u; \mathcal{D}_\ell) := \mathbb{E}_j[\hat{f}_{\psi_j}(u)], \quad \sigma^2(u; \mathcal{D}_\ell) := \mathrm{Var}_j[\hat{f}_{\psi_j}(u)].$$

The upper-confidence-bound (UCB) score of each candidate control u is then formulated as

$$\alpha^{\mathrm{control}}(u; \mathcal{D}_\ell) := \mu(u; \mathcal{D}_\ell) + \kappa\sigma(u; \mathcal{D}_\ell),$$

where $\kappa \geq 0$ balances the preference for high predicted performance (exploitation) against the preference for uncertain predictions (exploration). Both statistics depend on the current dataset $\mathcal{D}_\ell$, and the UCB score is computed for each candidate control u generated at its assigned target y.

For a candidate pair (u, y), the combined score is

$$\alpha(u, y; \mathcal{D}_\ell) := \alpha^{\text{control}}(u; \mathcal{D}_\ell) + \eta \alpha^{\text{target}}(y; \mathcal{D}_\ell), \quad (6)$$

where $\eta \geq 0$ weights target reliability relative to control performance. This score is used to rank candidates for selecting the most promising ones to evaluate in simulation, balancing the trade-off between high predicted performance and low uncertainty.

At each iteration, we sample $\widehat{I} > I$ candidate targets in $\mathcal{Y}_\ell$, where $\widehat{I}$ is the proposal count and I the new simulation batch size. For each candidate index $i \in \{1, \ldots, \widehat{I}\}$, we draw $u^i \sim p_{\theta_e}(\cdot \mid y^i)$ at its assigned target $y^i \sim \text{Unif}\{\mathcal{Y}_\ell\}$, with e sampled uniformly from the diffusion ensemble. Candidates receive their combined score $\alpha(u^i, y^i; \mathcal{D}_\ell)$ and are ranked accordingly. The top I candidates are selected for simulation, and their resulting objective values supply new pairs $\{(u^i, f(u^i))\}_{i=1}^{I}$ to the dataset for the next iteration. Algorithm 1 summarizes the full iterative procedure, illustrated in Figure 3a–b.

V. EXPERIMENT SETUP

CyberOctopus model. The CyberOctopus models a soft-arm robot driven by distributed internal actuation (Figure 2). Its architecture is inspired by the octopus arm's muscular hydrostat [6], with deformation capabilities relevant to pneumatic, tendon-driven, and embedded soft-actuation systems [24], [25]. Each tapered arm m is represented as a geometrically exact Cosserat rod [26], [27] that captures large bending, twisting, shear, and extension (Figure 2a). The rod is parameterized by arc length $s \in [0, L_0]$ and time $t \in [0, T]$, where L_0 is the arm length at rest and T is the locomotion cycle duration. Each of the M arms connects to a central spherical mantle via translational and rotational restoring springs. In their resting state, the arms are spaced at equal intervals and align with the directional vectors

$$(\cos \phi_m,\ \sin \phi_m), \quad \phi_m = (m - \tfrac{1}{2}) \triangle \phi$$

for all $m \in \{1, 2, \ldots, M\}$.

Let $\bar{\mathbf{x}}(s, t)$ and $\mathbf{Q}(s, t)$ denote an arm's centerline position and material frame. Its linear and angular momentum balances are

$$\begin{aligned} \partial_t(\rho \partial_t \bar{\mathbf{x}}) &= \partial_s(\mathbf{Q}\mathbf{n}) + \bar{\mathbf{f}}, \\ \partial_t(\mathbf{I}\boldsymbol{\omega}) &= \partial_s(\mathbf{Q}\mathbf{m}) + (\partial_s \bar{\mathbf{x}}) \times (\mathbf{Q}\mathbf{n}) + \bar{\mathbf{c}}, \end{aligned} \quad (7)$$

where ρ and $\mathbf{I}$ are the mass and second moment of inertia per unit length, respectively, and $\boldsymbol{\omega}$ is the angular velocity. The pairs $(\mathbf{n}, \mathbf{m})$ and $(\bar{\mathbf{f}}, \bar{\mathbf{c}})$ denote the internal force and couple resultants and the external force and couple densities, respectively. External loads comprise gravity, compliant normal contact, anisotropic kinetic Coulomb friction, arm-to-arm contact, and sucker adhesion [20], [28].

Distributed actuation. Following the octopus arm histology [6], [20], each arm contains four longitudinal muscles $\text{LM}_{1:4}$ for directional bending, one transverse muscle TM for elongation, and two opposing oblique muscles $\text{OM}_\pm$ for twisting (Figure 2e–f). Two groups of ventral suckers, proximal S_p and mid-arm S_m, provide intermittent substrate anchoring. These nine actuation elements form the actuation set $\mathcal{A} = \{\text{LM}_1, \text{LM}_2, \text{LM}_3, \text{LM}_4, \text{TM}, \text{OM}_+, \text{OM}_-, \text{S}_\text{p}, \text{S}_\text{m}\}$. Each element's activation $\nu^a(t) \in [0, 1]$ follows a temporal Gaussian pulse specified by its amplitude, width, and peak phase within the locomotion cycle T. Stacking these parameters gives the control $u^{(m)} \in \mathcal{U} = [-1, 1]^U$ for arm m, with dimension $U = |\mathcal{A}| \times 3 = 27$.

For an actuation element $a \in \mathcal{A}$, let $l^a(s, t) > 0$, $\mathbf{t}^a(s, t) \in \mathbb{S}^2$, and $\mathbf{x}^a(s, t) \in \mathbb{R}^3$ denote its local relative length, unit tangent, and cross-sectional offset, respectively. Omitting the arguments (s, t) for brevity, the contractile force and corresponding force and couple resultants are

$$n^a = \nu^a h(l^a), \qquad \mathbf{n}^a = n^a \mathbf{t}^a, \qquad \mathbf{m}^a = \mathbf{x}^a \times \mathbf{n}^a, \quad (8)$$

where $h : \mathbb{R}_+ \to \mathbb{R}_+$ is the force-length response (Hill's model) at full activation. Adding the muscle contributions to the arm's passive elastic response gives the total internal force and couple resultants,

$$\mathbf{n} = \mathbf{n}^\text{p} + \sum_{a \in \mathcal{A}} \mathbf{n}^a, \quad \mathbf{m} = \mathbf{m}^\text{p} + \sum_{a \in \mathcal{A}} \mathbf{m}^a,$$

where $\mathbf{n}^\text{p}$ and $\mathbf{m}^\text{p}$ are the passive elastic resultants. These internal loads couple muscle actuation (8) to arm deformation through the Cosserat dynamics (7). Together with substrate contact, they produce the displacement $\Delta\mathbf{p}(u)$ and rotation $\Delta\Phi(u)$ used to evaluate control u in objective function (2).

Baselines and metrics. Experiments first assess full-direction control performance after symmetry folding and interpolation, together with path-tracking on a sequence of waypoints. Second, we analyze optimization progress against similar baselines: discovery of high-objective controls and maintaining variability in the generated controls.

We compare DUO with DiffBBO and DiBO, the closest iterative diffusion-optimization methods. Multi-CMA-ES [29] is included as an intuitive extension that obtains multiple solutions from a population-based optimization. Lastly, we compare with AutoQD [13], a variant of a quality–diversity method that automatically constructs the behavior descriptors, applicable to our problem. Each baseline is implemented from its manuscript and adapted to our environment setup. For methods using neural networks, we match the architecture and parameter count. All optimization uses the same simulator-evaluation budget.

Adaptation evaluation. Lastly, we demonstrate an essential benefit of adopting a diffusion-based method: adaptation under actuation constraints, without retraining. Using an editing technique similar to *inpainting* [30], [31], additional constraints can be imposed on a previously generated control by iterating denoise-constrain-noise steps [30] (Figure 3c).

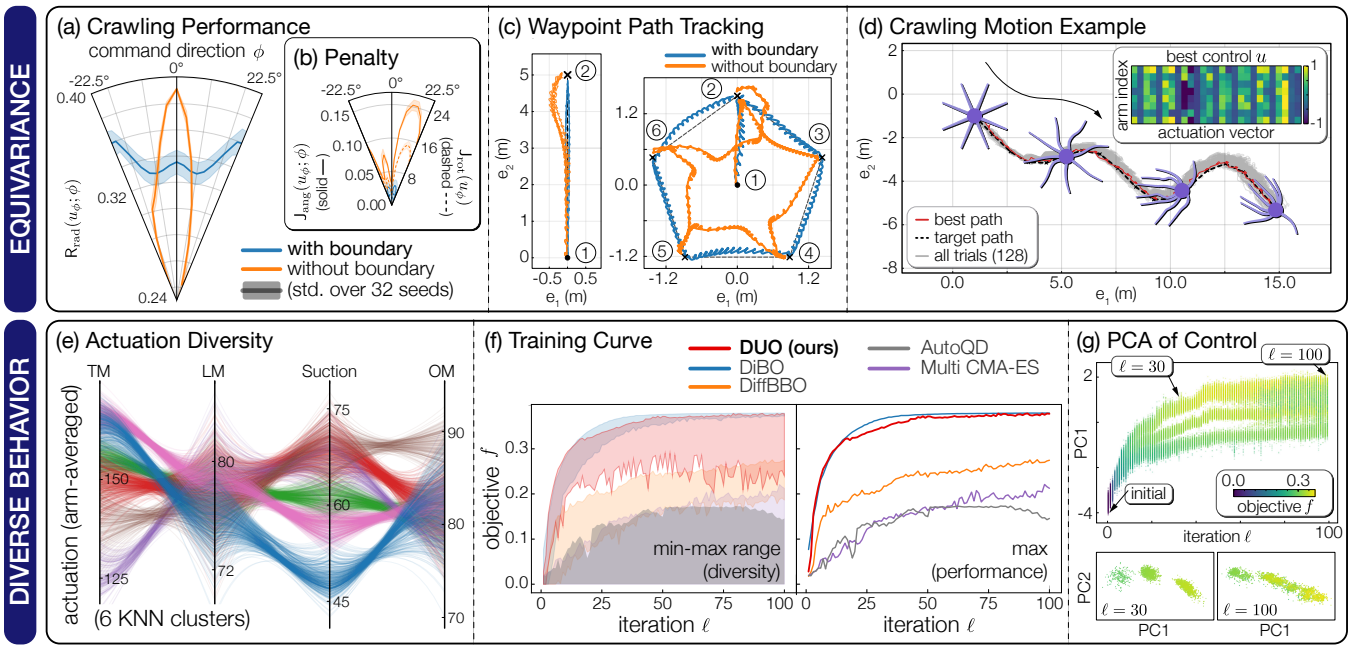

Fig. 4: **Directional crawling and control diversity.** (a) Crawling objective over $\phi \in [-22.5°, 22.5°]$ with (blue) and without (orange) the boundary objective; shading indicates standard deviation over 32 seeds. (b) Angular deviation (solid, left scale) and body rotation (dashed, right scale). (c) Waypoint tracking with the same color convention as (a); black dashed lines mark the commanded paths. (d) Crawling along a curved target path (black dashed), showing successive body configurations, trajectories from 128 controls (gray), and best control (red); inset: best control values. (e) Arm-averaged time-integrated activation of TM, LM, suction, and OM recruitment across the control family. The family is clustered using the k-nearest neighbors (KNN) algorithm (k = 6). (f) Per-iteration objective ranges (left) and maxima (right) for the compared methods. (g) PCA of the generated control distribution across training iterations, $1,800$ samples per iteration. PC1 and PC2 indicate the first and second principal components.

For evaluation, we selected three scenarios that impose a significant, yet recoverable, impact on the controller. First, a 25% reduction in all muscles, simulating a mild, uniformly degrading case that would require a small amount of control reconfiguration to recover. Next, a 75% reduction in suction on all arms, mimicking ineffective suction, as when a surface affords only a weak grip such as a simple granular surface. Lastly, complete suppression of actuation on four alternating arms ($m = 1, 3, 5, 7$) to simulate partial arm availability.

Hyperparameters. All evaluations run for 100 iterations of $I = 1{,}800$ evaluations. For DUO, we use $E = J = 8$ ensemble sizes and generate candidates at a ratio $\widehat{I}/I = 3$, and sample target objective candidates $\{y^i\}_{i=1}^{\widehat{I}}$ uniformly from $\mathcal{Y}_\ell = [0.6y_{\max,\ell}, y_{\max,\ell}]$ for each iteration ℓ. After a hyperparameter sweep, we use $\lambda = 0.5$, $\kappa = 1$, and $\eta = 0.25$, which attained the highest performance while maintaining variation among the controls explored during search.

We train each denoiser for $10{,}000$ gradient steps per iteration, using $K = 64$ diffusion steps with a linear variance schedule [21], conditioning dropout is 0.15, and classifier-free guidance scale is 2 [32]. Remaining hyperparameter details are provided in the code repository.

VI. RESULTS AND DISCUSSION

Full-direction crawling. Figure 4a–b evaluates the controller over the canonical forward sector. Without the boundary objective, interpolation severely degrades in the interior of the sector, in both the crawling objective and the penalty terms. This deviation is more pronounced in waypoint tracking (Figure 4c). Adding the boundary objective, on the other hand, loses a little performance at the center, but yields more uniform crawling in all directions, with reduced angular deviation and body rotation (Figure 4b). Together, these results support including the boundary objective to improve directional consistency between arm-aligned commands.

Figure 4d illustrates crawling along a curved target path, with successive body configurations showing coordinated arm motion as the commanded direction changes. All trajectories from 128 generated controls show successful tracking and reach the end. This example demonstrates how the directional controller construction supports successive changes in crawling direction.

Multimodal control discovery. The actuation profiles in Figure 4e, averaged across the arms, provide a muscle-coordination view of the discovered controller diversity. The control families follow distinct group across all activation types, rather than differing only by small deviations from a shared profile. Interestingly, LM and OM actuation level remain relatively uniform across controls, while more distinct characteristics emerge in the utilization of TM and suction. Together, these results indicate that the learned distribution contains multiple coordination patterns for the crawling task. They do not, by themselves, establish mechanically discrete gaits, but instead reveal distinct control and recruitment patterns within a common locomotion task.

The progression of the training is visualized through training curve and a principal-component embedding, with color indicating the achieved objective (Figure 4f–g). Initially at $\ell = 0$, sampled controls occupy a compact, low-objective region. By $\ell = 30$, the objective has risen rapidly and the

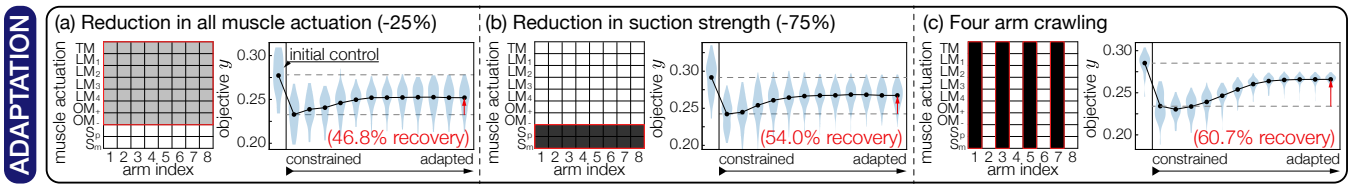

Fig. 5: **Adaptive control editing under actuation constraints.** Performance objective recovery outcomes produced by the trained and fixed DUO denoiser: (a) 25% muscle-amplitude reduction across all arms; (b) 75% suction-amplitude reduction across all arms; (c) zero muscle and suction amplitudes on arms 1, 3, 5, and 7. On the left, dark cells mark constrained channels. On the right, blue violin plots show retained-outcome objective distributions over 64 generated controls; black markers show means. The plotted points illustrate the performance across the adaptation process: starting from the initial control, applying the constraint, and showing the subsequent edited versions across the iterative denoising steps. Red percentages report the mean of recovery, $(y_{\text{adapted}} - y_{\text{constrained}})/(y_{\text{initial}} - y_{\text{constrained}}) \times 100\%$.

distribution has branched into several clusters. As training proceeds, both the PC1–PC2 projection and variation in the training curve indicate continued exploration: generated controls continue to spread outwards while previously discovered solutions are retained and exploited. This spread persists through the end of training ($\ell = 100$) rather than collapsing onto the single best control, with some generated controls remaining in slightly suboptimal regions. Thus, DUO retains high-performing controls while continuing to query nearby, non-maximal regions during online optimization.

Comparison with other methods. Training progress and comparisons with related methods are reported in Figure 4f. The plot shows the min–max range of the objective together with the corresponding batch maximum.

DiBO and DUO reach the highest values, with DiBO converging slightly earlier. DUO continues to evaluate controls over a wider objective range than DiBO. This range implies that the acquisition procedure does not concentrate too quickly, and that evaluation includes a wider range away from the optimum. Once the maximum is reached, the variance increases slowly as the search continues actively in under-explored regions.

DiffBBO improves more gradually and reaches a lower batch maximum of approximately 0.27, and exhibits a smaller min-max range compared to DUO. Additionally, while Multi-CMA-ES demonstrates high diversity, its batch objective remains below that of the diffusion-based methods under the matched evaluation budget.

AutoQD improves more slowly, attaining a retained best-so-far of 0.173 later than the diffusion-based methods, but it searches more thoroughly. Its quality–diversity mechanism ranks candidates by improvement within cells of a learned behavior archive rather than by raw objective alone [13], so emitters continue to probe underrepresented regions—the corners of the archive—after a high-performing control has been found. Periodic updates of the behavior descriptor remap the archive and restart the emitters, introducing discontinuities between proposal batches; the batch mean and maximum can therefore decline while the retained best-so-far is unchanged. That late decline reflects continued coverage of the behavior space and the quality–diversity trade-off, not loss of the archive's best controller. Although AutoQD's batch maximum remains below DUO, the curves indicate broader coverage of AutoQD's archive.

Adaptation. Figure 5 reports adaptive control editing under the three different actuation constraints, with the trained denoiser held frozen. The violin plots show how generated controls perform through editing with positive recovery. In all three cases, editing restores about half or more of the lost objective, including when four alternating arms are unavailable. These results support constrained adaptability in zero-shot: the learned distribution contains compatibility with the new limits, therefore enabling recovery without additional retraining. Further, recovery under constraints never imposed during training—especially the complete inactivation of four arms—shows that the learned denoiser can reconfigure its control strategy, a mechanism distinct from simply retrieving a stored solution from the evaluation data.

Ablation study. Three ablation studies isolate the importance of each mechanism introduced in DUO (Table I). First, we compare against using a single conditioning objective, set to the maximum value in the observed dataset, as commonly used in optimization without a diversity focus or in offline settings. Conditioning exclusively on the maximum is reasonable when seeking a single high-performing control, but limits exploration and can prematurely concentrate the search around local optima. Indeed, this replacement reduces the final objective by 33.3%, indicating that conditioning over a broader objective range promotes effective exploration and better coverage across the y-conditioning space.

We next isolate the two acquisition mechanisms that distinguish DUO from a single-path diffusion query. Removing the term α^{control} produces a 36.7% reduction, while removing the term α^{target} reduces the objective by 14.6%. Together, the three ablations show complementary roles: the ranged target-member pool maintains search coverage, control acquisition directs the simulation budget toward promising controls, and target acquisition keeps queries within reliable objective levels. These mechanisms balance exploration with selective evaluation, providing informative data throughout online training.

TABLE I: Ablation study of DUO components. Values are mean over five independent runs, each of 100 iterations.

Variation	Final best	Δ (%)
DUO	**0.384**	–
- sample u^i from $p_{\theta_e}(\cdot\|y_{\max})$ instead	0.256	−33.3
- without α^{control}	0.243	−36.7
- without α^{target}	0.328	−14.6

VII. CONCLUSION

In soft robotics, control redundancy gives rise to a wide variety of solutions, which diffusion models offer substantial potential to handle. In support of this, we present Diffusion-based Uncertainty-aware Optimization (DUO), an efficient online data-collection procedure for training a denoiser. This method learns to generate high-performing controls without relying on prior demonstrations. Together with radial symmetry reduction, we obtain diverse muscle-arm coordination strategies for planar crawling in all commanded directions for the CyberOctopus. Further, diffusion-based control editing enables partial performance recovery under previously unseen actuation constraints without retraining. These results demonstrate how DUO turn the representational capacity of diffusion models into a practical resource for both control and adaptation.

More broadly, these results suggest the potential of diffusion-based controller for interactive soft-robot operation, beyond isolated experimental settings, and toward open-environment interaction. Future work will validate the generated controllers and adaptation strategies on physical hardware [33], incorporate sensory feedback [34] for more complex tasks, and extend the approach to three-dimensional locomotion [35].

REFERENCES

[1] F. W. Grasso, "Octopus sucker-arm coordination in grasping and manipulation," *American Malacological Bulletin*, vol. 24, no. 1, pp. 13–23, 2008.

[2] E. B. L. Kennedy, K. C. Buresch, P. Boinapally, and R. T. Hanlon, "Octopus arms exhibit exceptional flexibility," *Scientific Reports*, vol. 10, no. 20872, 2020.

[3] G. Levy, T. Flash, and B. Hochner, "Arm coordination in octopus crawling involves unique motor control strategies," *Current Biology*, vol. 25, no. 9, pp. 1195–1200, 2015.

[4] G. Levy and B. Hochner, "Embodied organization of octopus vulgaris morphology, vision, and locomotion," *Frontiers in Physiology*, vol. 8, 2017.

[5] C. Bennice, K. Buresch, J. Grossman, T. Morano, and R. Hanlon, "Octopus arm flexibility facilitates complex behaviors in diverse natural environments," *Scientific Reports*, vol. 15, 09 2025.

[6] L. Margheri, C. Laschi, and B. Mazzolai, "Soft robotic arm inspired by the octopus: I. from biological functions to artificial requirements," *Bioinspiration and Biomimetics*, vol. 7, p. 025004, 05 2012.

[7] H. Hauser, A. J. Ijspeert, R. M. Füchslin, R. Pfeifer, and W. Maass, "Towards a theoretical foundation for morphological computation with compliant bodies," *Biological Cybernetics*, vol. 105, no. 5–6, 2011.

[8] J. K. Pugh, L. B. Soros, and K. O. Stanley, "Quality diversity: A new frontier for evolutionary computation," *Frontiers in Robotics and AI*, vol. Volume 3 - 2016, 2016.

[9] O. Nilsson and A. Cully, "Policy Gradient Assisted MAP-Elites," in *The Genetic and Evolutionary Computation Conference*, Lille, France, Jul. 2021.

[10] J. Lehman and K. O. Stanley, "Abandoning objectives: Evolution through the search for novelty alone," *Evol. Comput.*, vol. 19, no. 2, p. 189–223, Jun. 2011.

[11] C.-H. Shih, N. Naughton, U. Halder, H.-S. Chang, S. H. Kim, R. Gillette, P. G. Mehta, and M. Gazzola, "Hierarchical control and learning of a foraging cyberoctopus," *Advanced Intelligent Systems*, vol. 5, no. 9, p. 2300088, 2023.

[12] M. Allard, S. C. Smith, K. Chatzilygeroudis, B. Lim, and A. Cully, "Online damage recovery for physical robots with hierarchical quality-diversity," *ACM Trans. Evol. Learn. Optim.*, vol. 3, no. 2, Jun. 2023.

[13] S. Hedayatian and S. Nikolaidis, "AutoQD: Automatic discovery of diverse behaviors with quality-diversity optimization," in *The Fourteenth International Conference on Learning Representations*, 2026.

[14] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, p. 1798–1828, Aug. 2013.

[15] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, 2024.

[16] S. Krishnamoorthy, S. M. Mashkaria, and A. Grover, "Diffusion models for black-box optimization," in *Proceedings of the 40th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 202. PMLR, 2023.

[17] D. Wu, N. L. Kuang, R. Niu, Y. Ma, and R. Yu, "Diff-BBO: Diffusion-based inverse modeling for black-box optimization," in *NeurIPS 2024 Workshop on Bayesian Decision-making and Uncertainty*, 2024.

[18] T. Yun, K. Om, J. Lee, S. Yun, and J. Park, "Posterior inference with diffusion models for high-dimensional black-box optimization," in *Proceedings of the 42nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 267. PMLR, 2025, pp. 73 897–73 917.

[19] H.-S. Chang, U. Halder, C.-H. Shih, N. Naughton, M. Gazzola, and P. G. Mehta, "Energy-shaping control of a muscular octopus arm moving in three dimensions," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 479, no. 2270, p. 20220593, 2023.

[20] A. Tekinalp, N. Naughton, S. H. Kim, U. Halder, R. Gillette, P. G. Mehta, W. Kier, and M. Gazzola, "Topology, dynamics, and control of a muscle-architected soft arm," *Proceedings of the National Academy of Sciences*, vol. 121, no. 41, p. e2318769121, 2024.

[21] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 6840–6851.

[22] Y. Zhang, B. Hartl, H. Hazan, and M. Levin, "Diffusion models are evolutionary algorithms," in *International Conference on Learning Representations*, 2025.

[23] Z. Wang, N. M. Freris, and X. Wei, "Spirobs: Logarithmic spiral-shaped robots for versatile grasping across scales," *Device*, 2023.

[24] D. Rus and M. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, pp. 467–75, 05 2015.

[25] M. Cianchetti, C. Laschi, A. Menciassi, and P. Dario, "Biomedical applications of soft robotics," *Nature Reviews Materials*, vol. 3, no. 6, pp. 143–153, Jun 2018.

[26] S. S. Antman, *Nonlinear Problems of Elasticity*. Springer, 1995, vol. 107.

[27] M. Gazzola, L. H. Dudte, A. G. McCormick, and L. Mahadevan, "Forward and inverse problems in the mechanics of soft filaments," *Royal Society Open Science*, vol. 5, no. 6, p. 171628, 06 2018.

[28] S. H. Kim, J. Guo, A. Tekinalp, H.-S. Chang, U. Akcal, T. Wang, D. Biskup, B. Walt, G. Chowdhary, G. Krishnan, P. G. Mehta, and M. Gazzola, "Digital twins for the design, interactive control, and deployment of modular, fiber-reinforced soft continuum arms," 2025.

[29] N. Hansen, "The cma evolution strategy: A tutorial," *ArXiv*, vol. abs/1604.00772, 2016.

[30] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, "RePaint: Inpainting using denoising diffusion probabilistic models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 461–11 471.

[31] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon, "SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations," Jan. 2022.

[32] J. Ho and T. Salimans, "Classifier-Free Diffusion Guidance," in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[33] M. S. Nazeer, Y. T. Ansari, E. Solfiti, B. S. Bal, M. L. Preti, B. Mazzolai, and C. Laschi, "Benchmarking data-driven control of octopus-inspired soft arms in underwater environment," *Advanced Intelligent Systems*, p. e70506, 2026.

[34] T. Wang, H.-S. Chang, S. H. Kim, J. Guo, U. Akcal, B. Walt, D. Biskup, U. Halder, G. Krishnan, G. Chowdhary *et al.*, "A neural network-based framework for fast and smooth posture reconstruction of a soft continuum arm," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 1–8.

[35] M. Wu, W. H. Afridi, J. Wu, R. H. Afridi, K. Wang, X. Zheng, C. Wang, and G. Xie, "Octopus-inspired underwater soft robotic gripper with crawling and swimming capabilities," *Research*, vol. 7, p. 0456, 2024.