

# Inverse Geometric Locomotion

QUENTIN BECKER\*, EPFL, Switzerland

OLIVER GROSS\*, University of California, San Diego, USA and EPFL, Switzerland

MARK PAULY, EPFL, Switzerland

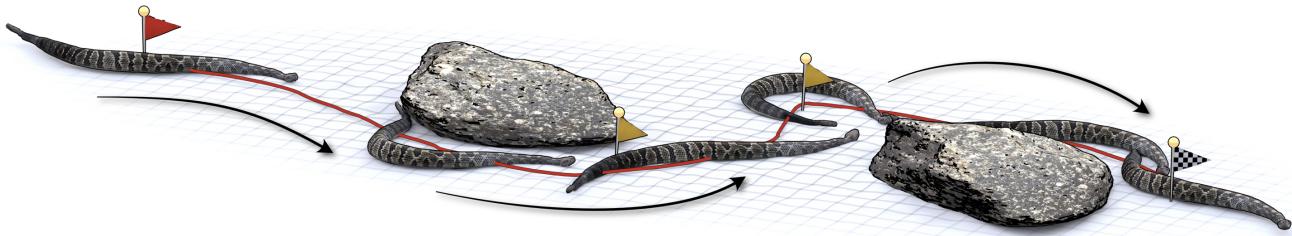


Fig. 1. Our inverse geometric locomotion optimization enables the discovery of, e.g., optimal snake undulation patterns for efficient slithering locomotion from a starting position (red flag), passing through landmarks (yellow flags), and reaching a prescribed target position (checkered flag), all while avoiding obstacles. See video at 00:05.

Numerous tasks in robotics and character animation involve solving combinations of inverse kinematics and motion planning problems that require the precise design of pose sequences to achieve desired motion objectives. Accounting for the complex interplay between body deformations and resulting motion, especially through interactions with the environment, poses significant challenges for the design of such pose sequences. We propose a computational framework to address these challenges in scenarios where the motion of a deformable body is entirely determined by dynamic changes of its shape. Complementing recent methods on the forward problem—mapping shape sequences to global motion trajectories based on a geometric formulation of locomotion—we address the inverse problem of optimizing shape sequences to achieve user-defined motion objectives. We demonstrate the effectiveness of our method through a diverse set of examples, producing realistic shape sequences that result in desired motion trajectories.

CCS Concepts: • Applied computing → Physics; Engineering; • Computing methodologies → Modeling methodologies; Physical simulation; Motion processing;

Additional Key Words and Phrases: Inverse design, inverse kinematics, geometric locomotion, shape change, spacetime optimization, gait, character dynamics, animation

#### ACM Reference Format:

Quentin Becker, Oliver Gross, and Mark Pauly. 2025. Inverse Geometric Locomotion. *ACM Trans. Graph.* 44, 4 (August 2025), 17 pages. <https://doi.org/10.1145/3731187>

\*Both authors contributed equally to this research.

Authors' addresses: Quentin Becker, EPFL, Lausanne, Switzerland, quentin.becker@gmail.com; Oliver Gross, University of California, San Diego, CA, USA and EPFL, Lausanne, Switzerland, ogross@ucsd.edu; Mark Pauly, EPFL, Lausanne, Switzerland, mark.pauly@epfl.ch.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM. 0730-0301/2025/8-ART \$15.00 <https://doi.org/10.1145/3731187>

## 1 INTRODUCTION

Countless organisms at all scales use patterns of self-deformation to achieve locomotion. Researchers have long been fascinated by falling cats whose self-righting reflex seems to defy the laws of conservation of angular and linear momentum. Similarly intriguing is the locomotion of tiny bacteria that wiggle their flagella, or the powerful, slithering motions of snakes. What all these locomotion strategies have in common is that translations and rotations of the moving body are achieved solely through changes in the body's shape.

For instances of this *geometric locomotion*, the relationship between dynamic changes in shape and the resulting displacement in the surrounding space can be reduced to conserved quantities, so-called *geometric momenta*, which are given by the Euler-Lagrange equations of appropriate variational principles. Prominent examples include Euler's principle of least action [1744] and Helmholtz's principle of least dissipation [1882].

Although based on a strict coupling between shape change and the resulting dynamics, geometric locomotion is applicable to a wide range of scenarios modeling unforced inertia-dominated motion of shape-changing bodies in negligible media or the locomotion of organisms in dissipation-dominated environments. Examples include a snake's slithering on sand, as well as abstract and bio-inspired anthropomorphic and non-anthropomorphic robotic systems [Shapere and Wilczek 1989a; Shammas et al. 2007; Zhao et al. 2022; Gross et al. 2023].

In general, geometric locomotors, such as organisms in highly viscous fluids, require sophisticated strategies to generate displacements within the constraints imposed by the distinctive relationship between their shape and motion. While a scallop can move in water by a sequence of slow opening and fast closing, such an endeavor is futile in low Reynolds number environments. According to Purcell's scallop theorem [1977], no cyclic shape change controlled by a single degree of freedom can result in a net displacement. The situation changes, when two scallops are connected, e.g., by a stick. With then two degrees of freedom, a coordinated collaborative effort permits

the system to achieve a net displacement, thus overcoming the restriction by Purcell's theorem—even if only in a single direction (Figure 2).

The skillful locomotion strategies we observe in biological systems have been refined over centuries of evolution to gain a competitive advantage over peers and predators. However, manual design of geometric locomotion for character animation or robotics is highly challenging. Consider, for example, a snake turning in a confined space (see also Figure 8). The objective here is to achieve a net rotation of the snake's body solely through deformations of its shape. The unknown pose sequence needs to be compatible with the morphology of the snake's body, respect the corresponding least-dissipation principle of motion, and avoid non-physical configurations due to collisions with the environment. This leads to a challenging inverse design problem that necessitates advanced computational tools.

The problem is aggravated by the sensitive coupling between shape change and motion, exemplified by the precise sequences of poses required, for example, by gymnasts or platform divers to perform somersaults and twists, which highlight the narrow margin of error for the input data of a simulation [Hodgins et al. 1995; Wooten and Hodgins 1996].

### 1.1 Contributions

The present paper addresses the above issues by proposing a method to solve the following problem:

*Efficiently find shape sequences that yield desired motion trajectories from geometric locomotion.*

Our core technical contribution is an efficient optimization scheme to solve inverse geometric locomotion problems. The presented framework, based on the concept of *geometric mechanics*, explicitly takes advantage of geometric symmetries and structure of the configuration space that give rise to conserved quantities. These integrals of motion allow us to transform the inverse problems into optimization problems defined solely over the shape evolution through time. Contrasting Newtonian physics simulations, the corresponding motion trajectories are fully determined by the condition of momentum conservation, a first order condition, which is equivalent to the equations of motion and can be strictly enforced by employing a variational integrator. We define a variety of high-level design objectives, such as multiple targets for position, orientation, and efficiency, incorporate collision avoidance and confinement constraints, and leverage reduced models to define the underlying shape space of a given deformable body.

To enable inverse design, we implement a differentiable simulation framework based on the geometric variational integrator of Gross et al. [2023]. We analytically derive the sensitivities of our design objectives with respect to a body's shape using adjoint state vectors, which facilitates efficient numerical solutions using standard optimization methods. Moreover, we show how our inverse formulation allows for the refinement of existing motion sequences to better adhere to a variety of user-specified targets, as well as the discovery of new gait patterns for complex locomotion tasks and arbitrary body shapes. Our framework thus enables novel applications in character animation and robotic design.

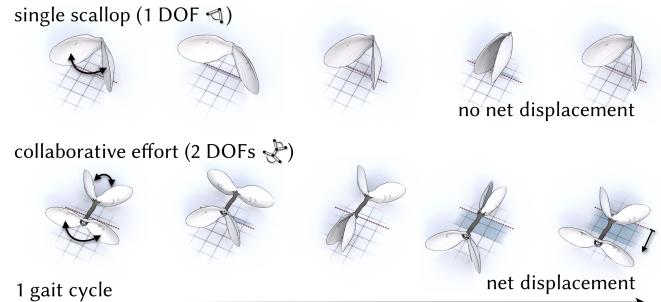


Fig. 2. According to Purcell's scallop theorem, cyclic shape change controlled by a single degree of freedom cannot produce net motion in highly viscous media [Purcell 1977]. However, with two degrees of freedom, two scallops attached to a stick can cooperate to produce a desired net displacement. See video at 00:38.

All data, the full source code of our method, and scripts to reproduce all examples in the paper are available for download at <https://go.epfl.ch/igl/>.

### 1.2 Related work

Understanding how morphological and kinematic properties influence key quantities such as locomotion speed, energy consumption, and turning capabilities for the geometric locomotion of, e.g., biological organisms or robotic locomotors remains an open challenge [Chong et al. 2023; Rieser et al. 2024]. To this day, optimal displacement strategies for even for highly simplified geometric locomotors remain elusive and analytical results are scarce [Purcell 1977; Becker et al. 2003; Tam and Hosoi 2007; Giraldi et al. 2015]. Therefore, character animation, optimal gait design and optimal control for scenarios governed by geometric locomotion remain a challenging tasks.

Despite steady progress in highly specialized numerical methods for studying geometric motion strategies [Shammam et al. 2007; Ramasamy and Hatton 2019; Bing et al. 2022; Chong et al. 2023], there is still a lack of an effective, broadly applicable tool for tackling inverse problems in geometric locomotion—a gap that we address with our proposed method. Our computational treatment faces several challenges. First, the need for a meaningful and expressive representation of character deformations to effectively model shape-changing bodies. In addition, a model for the dynamics of geometric locomotion must be suited to serve in a variety of applications ranging from character animation as well as the exploration and optimization of locomotion strategies. This requires not only an accurate representation of the distinct interplay between animates and their motion, but also efficient methods for interpolation and optimization of both shapes and trajectories.

*Shapes and deformations.* When modeling shape-changing characters, methods grounded in physical principles achieve realistic deformations by assigning material properties, typically assuming some degree of rigidity [Alexa et al. 2000] or elasticity [Martin et al. 2011; Kavan and Sorkine 2012]. These approaches have also been combined with modal actuations [Hildebrandt et al. 2011, 2012;

Bencherkroun et al. 2024]. Moreover, shape interpolations determined as intrinsic, locally shortest paths obtained from exploiting the geometric structures of the “space of shapes” yield realistic sequences of shape deformations [Heeren et al. 2014; Hartman et al. 2023a,b; Bauer et al. 2024]. As demonstrated by Sassen et al. [2024], the geometry of these shape spaces can be modified to naturally account for collision avoidance.

*Simulation and control of motion trajectories.* For motion planning tasks, methods that provide direct control over trajectories in the form of motion objectives such as targets, checkpoints or semantics [Ratliff et al. 2009; Aristidou et al. 2018] are typically preferred over frameworks which rely on the exploration of large numbers of scenes in a “needle in a haystack” scenario [Goel and James 2022].

There are a number of approaches to character animation based on the interpolation of a sparse set of keyframes, including but not limited to space-time constraints [Witkin and Kass 1988], physics-based and FEM-based interpolation [Barbić et al. 2009], and deep learning frameworks [Harvey et al. 2020; Qin et al. 2022]. Force-based controllers based on Newtonian physics simulations have been successfully implemented by Hodgins et al. [1995]; Wooten and Hodgins [1996] for complex animation tasks such as athletes running, cycling, vaulting, and platform diving, while Brandt et al. [2018] perform simulations in a reduced linear shape space using projective dynamics. Coros and colleagues [2012] implicitly model muscle actuation of elastic bodies: The rest shapes are optimized as a time sequence so that the motion trajectories resulting from interactions with the environment and induced deformations minimize user-defined objective functions. Jia et al. [2023] demonstrate how seamlessly looping physical animations can be synthesized. More recently, Kang et al. [2024] approached trajectory optimizations with reinforcement learning, while, e.g., differentiable simulation frameworks have been successfully employed for the inverse design of various animation tasks [Geilinger et al. 2020; Du et al. 2021b; Huang et al. 2024; Newbury et al. 2024] and have found application in biomechanics [Du et al. 2021a; Zhang et al. 2022].

*Coupling with ambient media.* In certain scenarios, the influence of the surrounding medium cannot be neglected. To address this, Kwatra et al. [2010] use a data-driven approach to create realistic simulations of aquatic creatures, while Tan et al. [2011] optimize their swimming gaits by considering a two-way coupling between the animate and the fluid. Computation times can be drastically reduced by means of highly simplified models which do not require the simulation of an ambient fluid. For example, methods based on a localized treatment of the applied forces [Gray and Hancock 1955] have been successfully applied for underwater simulation [Weißmann and Pinkall 2012; Soliman et al. 2024a]. These simple models have been incorporated in controller-based animations of birds [Wu and Popović 2003] or optimal control methods for underwater animals [Min et al. 2019].

*Geometric locomotion.* Contrasting the above scenarios, Gross et al. [2023] build on the fact that, in dissipation- or inertia-dominated settings locomotion can be modeled based on a geometric problem formulation [Shapere and Wilczek 1989a,c]. We provide a more detailed discussion of the formalism in Section 2, and simply note

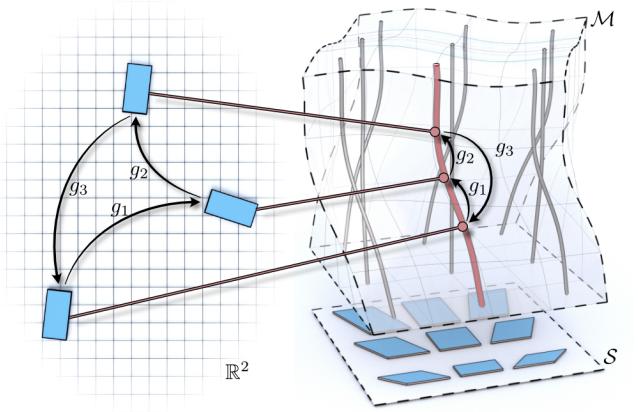


Fig. 3. Illustration of shape space  $S$  and configuration space  $M$  for the simple deformable model of a 2D parallelogram with one fixed edge length. The shape space  $S$  can be parameterized by two parameters, e.g., the second edge length and a shear value. Different positions of any given parallelogram in world space differ by a rigid body motion  $g_i$ . All of these motions make up the fiber in configuration space  $M$  corresponding to this shape.

here that the concept of *geometric mechanics* [Marsden and Ratiu 1999] has found successful application for the discovery of optimal and novel locomotion strategies for legged, undulating [Shapere and Wilczek 1989b; Ostrowski and Burdick 1998; Kobilarov et al. 2009; Ramasamy and Hatton 2019; Chong et al. 2023], abstract mechanical systems [Li et al. 2022] and fluid simulations [Nabizadeh et al. 2024].

### 1.3 Overview

The remainder of the paper is organized as follows: In Section 2, we review the mathematical background of a geometric formulation of locomotion that forms the basis of our inverse design optimization. We define a general inverse design problem for geometric locomotion in Section 3 and present a set of design objectives that enable precise user control over the resulting pose sequences. We show with a series of examples how combinations of these objectives enable effective exploration of complex motion trajectories. In Section 4 we discuss how the resulting continuous optimization problem can be discretized and solved efficiently using reduced models based on standard numerical techniques. We also present derivations of the required gradients and provide further implementation details. Finally, in Section 5 we evaluate our method, identify limitations of the current approach, and discuss directions for future work.

## 2 PRELIMINARIES

By treating the *shape* of a body in  $\mathbb{R}^3$  as decoupled from its respective *position* in world space, the concept of *geometric mechanics* allows us to examine theoretical and practical aspects of locomotive systems based on the geometric structure of their configuration space [Marsden and Ratiu 1999].

### 2.1 Rigid body motions

Although this paper is focused on the dynamics of shape-changing bodies, *rigid body transformations* will play an important role in our

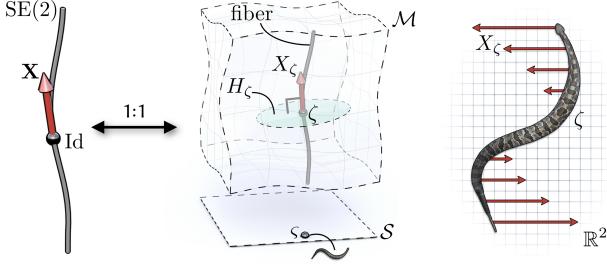


Fig. 4. Elements of the Lie algebra  $X \in \mathfrak{se}(3)$ , the tangent space of  $\text{SE}(3)$  at the identity  $\text{Id}$ , are in a one-to-one correspondence with infinitesimal rigid motions  $X_\zeta \in T_\zeta \mathcal{M}$ , i.e., vertical tangent vectors at a configuration  $\zeta \in \mathcal{M}$  that do not affect the shape  $\zeta \in \mathcal{S}$ .

method. These transformations are composed of a combination of a rotation and a translation on  $\mathbb{R}^3$ , and take the form

$$x \mapsto g(x) = Ax + b$$

for a rotation matrix  $A \in \text{SO}(3)$  and vector  $b \in \mathbb{R}^3$ . The composition of rigid body transformations satisfies the axioms of an algebraic group with the identity map  $\text{Id}$  as its neutral element. The collection of all rigid body motions is the *Euclidean group*  $\text{SE}(3)$ .

In addition to its structure as an algebraic group,  $\text{SE}(3)$  is also a smooth manifold, making it a so-called *Lie group* [Marsden and Ratiu 1999]. For a Lie group, the tangent space at the identity  $\text{Id} \in \text{SE}(3)$ , or *Lie algebra*, takes on a special role and is therefore given a dedicated notation  $\mathfrak{se}(3) := T_{\text{Id}}\text{SE}(3)$ . For the Euclidean group, the Lie algebra consists of infinitesimal rigid body motions  $X \in \mathfrak{se}(3)$ .

## 2.2 Shapes and their position

Arguably, two bodies in  $\mathbb{R}^3$  have the same *shape* if they only differ by a rigid body motion  $g \in \text{SE}(3)$ . This defines an equivalence relation on the *configuration space*  $\mathcal{M}$  of all shapes in all positions (Figure 3). The configuration space  $\mathcal{M}$  is a smooth manifold and we denote the set of smooth vector fields in its tangent bundle  $T\mathcal{M}$  by  $\Gamma T\mathcal{M}$ . The projection  $\pi: \mathcal{M} \rightarrow \mathcal{S}$  onto the *shape space*  $\mathcal{S} = \mathcal{M}/\text{SE}(3)$  decomposes the configuration space  $\mathcal{M}$  into six-dimensional submanifolds

$$\{\zeta = g(\xi) \mid g \in \text{SE}(3)\} \subset \mathcal{M},$$

which consist of all possible positions  $\zeta$  of a shape  $\xi$  (Figure 3). Therefore, each of these *fibers* can naturally be identified with  $\text{SE}(3)$  (Figure 4).

Throughout the paper we will restrict our attention to the case where  $\mathcal{M}$  is *trivial*<sup>1</sup>, i.e.,  $\mathcal{M} = \mathcal{S} \times \text{SE}(3)$ , since in practice, shapes provided by, e.g., an animator are positioned in a common reference frame. In that case, the first component of any positioned shape  $\zeta = (\xi, g) \in \mathcal{M}$  determines the shape  $\xi \in \mathcal{S}$ , while the second component  $g \in \text{SE}(3)$  describes the Euclidean transformation needed to move the shape from the common reference frame to its position in world space.

<sup>1</sup>For general *fiber bundles*,  $\mathcal{M}$  is only required to be locally isomorphic to  $\mathcal{S} \times \text{SE}(3)$  [Frankel 2011].

*The vertical distribution.* Any repositioning of the positioned shape  $\zeta \in \mathcal{M}$  can be described as the action

$$h(\zeta) = (\zeta, h \circ g)$$

of an element of the Euclidean group  $h \in \text{SE}(3)$ . Differentiating a one parameter family  $t \mapsto h_t \in \text{SE}(3)$  with  $h_0 = \text{Id}$  and  $h'_0 = X \in \mathfrak{se}(3)$  at configurations  $\zeta \in \mathcal{M}$  unveils a one-to-one correspondence between elements of the Lie algebra  $X \in \mathfrak{se}(3)$  with so-called *vertical vector fields*  $X \in \Gamma T\mathcal{M}$  given by

$$X_\zeta = \frac{d}{dt}|_{t=0} h_t(\zeta).$$

The collection of all vertical vector fields spans the *vertical distribution*  $V \subset T\mathcal{M}$ . For excellent in-depth discussions of the concepts of Lie groups, Lie algebras, and so-called *fiber bundles* in the context of physics and mechanics, we refer the reader to, e.g., [Marsden and Ratiu 1999] or [Frankel 2011].

## 2.3 Physical motion

When given a shape sequence  $\zeta: [0, T] \rightarrow \mathcal{S}$ , simulation of its dynamics can be thought of as repositioning the shapes from their common reference frame to positions in world space that represent a physically meaningful motion. From a geometric point of view, this amounts to determining a *lift*, i.e., a map  $\zeta: [0, T] \rightarrow \mathcal{M}$  with  $\pi(\zeta) = \zeta$ . *A priori*, there are many possible choices of such lifts, most of which carry no physical meaning (Figure 5).

In line with, for example, Euler's principle of least action [1744] or Helmholtz's principle of least dissipation [1882], which are satisfied by stationary points of total kinetic energy or total energy dissipation, respectively, we postulate that physical motions are extrema of appropriate variational principles. Therefore, we consider variational energies of the form

$$\mathcal{E}(\zeta) = \frac{1}{2} \int_0^T \langle \zeta', \zeta' \rangle_{\mathcal{M}} dt. \quad (1)$$

Here,  $\langle \cdot, \cdot \rangle_{\mathcal{M}}$  denotes an  $\text{SE}(3)$ -invariant Riemannian metric on the configuration space  $\mathcal{M}$ , that is, a Riemannian metric  $\langle \cdot, \cdot \rangle_{\mathcal{M}}$  such that for any two vectors  $\langle X, Y \rangle_{\mathcal{M}} = \langle g(X), g(Y) \rangle_{\mathcal{M}}$  for all  $g \in \text{SE}(3)$ . The specific choice of Riemannian metric determines the physics of the system we model. If the sequence of shapes is given, we seek lifts that are stationary points of Equation 1 under perturbations by vertical vector fields.

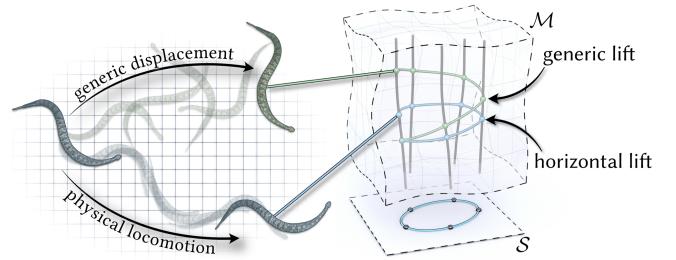


Fig. 5. A lift from a shape space  $\mathcal{S}$  to the configuration space  $\mathcal{M}$  assigns a position to any shape of a shape sequence. While generic lifts are generally not physically meaningful, geometric locomotion can be characterized by horizontal lifts.

*Horizontal distributions.* The assignment of an  $\text{SE}(3)$ -invariant metric allows us to define the *horizontal distribution*  $H := V^\perp$ , so that

$$T\mathcal{M} = V \oplus H.$$

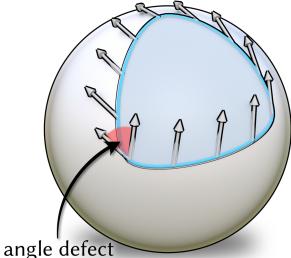
That is, any tangent vector  $\zeta' = (\zeta', g') \in T_\zeta \mathcal{M}$  can be orthogonally decomposed into an infinitesimal shape change  $\zeta' \in H_\zeta$  and an infinitesimal displacement  $g' \in V_\zeta$ . A vector field that completely lies in the horizontal distribution is said to be *horizontal*.

For bodies initially at rest, the Euler-Lagrange equations are satisfied whenever the tangent vector field  $\zeta'$  of the lift  $\zeta$  is horizontal<sup>2</sup> [Gross et al. 2023] (Figure 6). Intuitively, horizontal vector fields point as directly as possible to other fibers of the bundle, so that the configuration change does not contain any unnecessary rigid body motion components, thus minimizing the energy.

But how does horizontally traversing the fibers lead to a net displacement? The key to this *geometric locomotion* lies in the fact that the horizontal vector fields determine the notion of *parallel transport* [Frankel 2011]. Specifically, the parallel transport of a positioned shape  $\zeta^* \in \mathcal{M}$  along a shape sequence  $\zeta: [0, T] \rightarrow \mathcal{S}$  with  $\pi(\zeta^*) = \zeta^0$  is given by horizontal lifts  $\zeta: [0, T] \rightarrow \mathcal{M}$ , which are determined by

$$\pi(\zeta) = \zeta, \quad \zeta' \in \Gamma H \subset \Gamma \mathcal{M}. \quad (2)$$

*Geometric locomotion.* In Riemannian geometry, we observe the effects of curvature for example as angle defects when we parallel transport vectors along a closed path (see inset).



Similarly, for geometric locomotion, the curvature of the configuration space manifests when we parallel transport a configuration  $\zeta^* \in \mathcal{M}$  along periodic shape sequences, or *gaits*  $\zeta: [0, T] \rightarrow \mathcal{S}$ . Then, despite the periodicity of  $\zeta$ , its horizontal lifts  $\zeta$  are in general aperiodic (Figure 5). With the identification  $\pi^{-1}(\{\zeta^0\}) \cong \text{SE}(3)$ , the *geometric phase* is the resulting net displacement of the body after one gait cycle given by  $g_0^{-1} g_T \in \text{SE}(3)$  (Figure 5).

*The momentum map.* We can measure the component of a tangent vector  $Y \in TM$  along a provided rigid-body motion  $\mathbf{X} \in \text{se}(3)$  by means of the *momentum map*

$$\mu: TM \rightarrow \text{se}(3)^*, Y \mapsto \mu_Y.$$

A *geometric momentum*  $\mu_Y$  is evaluated on Lie algebra elements  $\mathbf{X} \in \text{se}(3)$  with corresponding vertical vector field  $X \in \Gamma V$  by

$$\mu_Y(\mathbf{X}) := \langle Y, X \rangle_{\mathcal{M}}.$$

This expression vanishes on the horizontal distribution *i.e.*,  $\mu|_H \equiv 0$ , while it identifies the vertical distribution  $V$  with the trivial bundle  $M \times \text{se}(3)$ .

The *momentum of a lift*  $\zeta$  is given by  $\mu_{\zeta'}$  and measures the rigid-body motion component of the infinitesimal shape change  $\zeta'$ . For

<sup>2</sup>Notably, a variety of other problems relevant to computer graphics can be formulated from a similar point of view [Montgomery 2002], *i.e.*, as finding horizontal sections of suitable fiber bundles (see, *e.g.*, [Padilla et al. 2019; Palmer et al. 2024; Nabizadeh et al. 2024]).

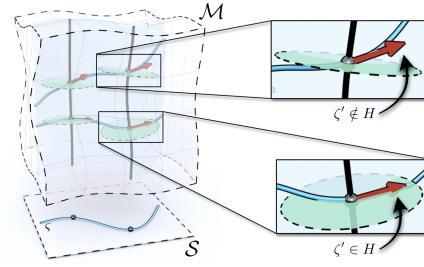


Fig. 6. Two lifts of the same curve  $\zeta$  in the shape space  $\mathcal{S}$  to the configuration space  $\mathcal{M}$ . While tangent vectors  $\zeta'$  of the upper one have vertical components, the bottom lift is always tangent to the horizontal distribution  $H$  making it a horizontal lift.

the present paper, the importance of the momentum map lies in the fact that we can state an equivalent definition of parallel transport (Equation 2) in terms of the momentum of the lift  $\zeta$  as

$$\pi(\zeta) = \zeta, \quad \mu_{\zeta'} \equiv 0.$$

Since we can identify  $\text{se}(3)^* \cong \mathbb{R}^6$ , this condition<sup>3</sup> is favorable to enforce computationally when compared to Equation 2, since it reduces to 6 equality constraints. In particular, from this point of view and for the special case that  $\mathcal{E}$  is the kinetic energy, the components of the  $\mu_{\zeta'} \in \mathbb{R}^6$  recover the common angular and linear momentum vectors (see, *e.g.*, [Gross et al. 2023, App. A]).

### 3 INVERSE PROBLEMS

Choosing an appropriate Riemannian metric on the configuration space  $\mathcal{M}$  allows lifting given shape sequences to physically meaningful motions as shown in [Gross et al. 2023]. Here, we aim to invert this process and solve for a body’s shape deformations such that the corresponding motion best matches user-given objectives.

More precisely, we optimize for a shape sequence  $\zeta: [0, T] \rightarrow \mathcal{M}$  such that the motion represented by a horizontal lift to configuration space minimizes an objective function  $\mathcal{J}$  of the form

$$\mathcal{J}: (\zeta: [0, T] \rightarrow \mathcal{M}) \rightarrow \mathbb{R}_{\geq 0}. \quad (3)$$

The resulting inverse design optimization is then formulated as

$$\underset{\zeta: [0, T] \rightarrow \mathcal{S}}{\operatorname{argmin}} \mathcal{J}(\zeta) \quad \text{s.t. } H(\zeta) = \zeta. \quad (4)$$

As a function of  $\zeta = (\zeta, g)$ , objectives of this form can promote or penalize shapes  $\zeta$ , the transformation  $g$  positioning them, or both. Note that determining whether an animator’s goal of transitioning between two keyframes, *i.e.*, “moving from point A to point B” is always possible typically requires extensive analysis [Montgomery 2002]. By solving our inverse problems (Equation 4) in a “least squares” sense, we avoid this necessity.

We now present different optimization objectives and illustrate their application for specific inverse design tasks. In Section 3.2 we will then discuss different strategies to represent shape spaces using reduced models to specify deformation semantics and simplify the inverse design optimization.

<sup>3</sup>By conservation of the vertical component a similar statement remains true for scenarios in which the momentum of the lift  $\zeta$  is non-zero (see, *e.g.*, [Gross et al. 2023]).

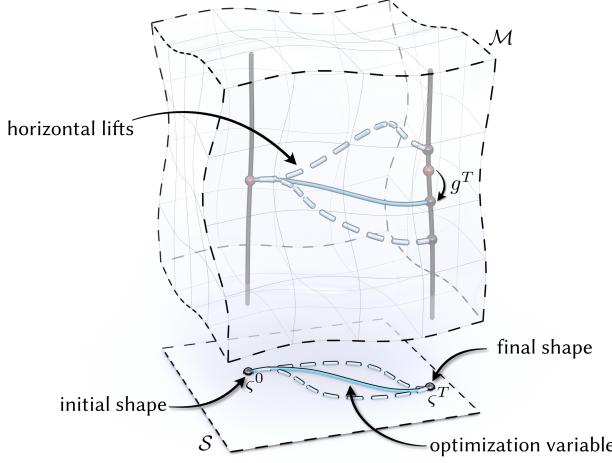


Fig. 7. The objects of the inverse problem Equation 4 for reaching a given target transformation (Equation 5) with the additional constraint of fixed start end shapes visualized in the geometric framework.

### 3.1 Inverse Design Objectives

*Position and orientation targets.* The most basic objective is to provide a desired relative target position and/or orientation for the final pose in the shape sequence. This objective can be expressed as the squared distance of the resulting net transformation to some target transformation  $g^* \in \text{SE}(3)$ , i.e.,

$$\mathcal{J}_{\text{pos}}(\zeta) = \frac{1}{2} |g^* - g_0^{-1} g_T|_{\text{SE}(3)}^2, \quad (5)$$

for a suitable norm  $|\cdot|_{\text{SE}(3)}$  on  $\text{SE}(3)$ . For example, we can specify a net displacement of the shape's centroid to steer the shape to a desired target location and/or control its final orientation at the target. Figures 1, 2, 8, 9, 11, 12, 14, 18, 17, 19, 20 and 21 apply this objective to illustrate how our inverse design optimization automatically finds non-trivial shape deformations that lead to natural motion sequences without any need for user intervention.

We can provide more control to the user by specifying a series of checkpoint transformations that the resulting trajectory should stay close too:

$$\mathcal{J}_{\text{cpt}}(\zeta) = \sum_{k=1}^{\ell} \min_{t \in [0, T]} \frac{1}{2} |g_k^* - g_0^{-1} g_t|_{\text{SE}(3)}^2. \quad (6)$$

This formulation comes with the additional benefit that the time at which the target is reached is not fixed *a priori*, but is also part of the optimization. Figure 1 illustrates a complex motion with multiple checkpoints.

We can generalize discrete checkpoints to continuous functions on the entire motion sequence. Consider the example of a snake turning by 180°. Without additional constraints on the motion, when initialized with a forward slithering trajectory, our algorithm finds a shape sequence that results in a “U-turn” motion (Figure 8 top left). This motion exhibits a significant displacement of the snake's center along the path of motion. If an animator would rather prefer a rotation “on the spot,” we can add a penalty term for the

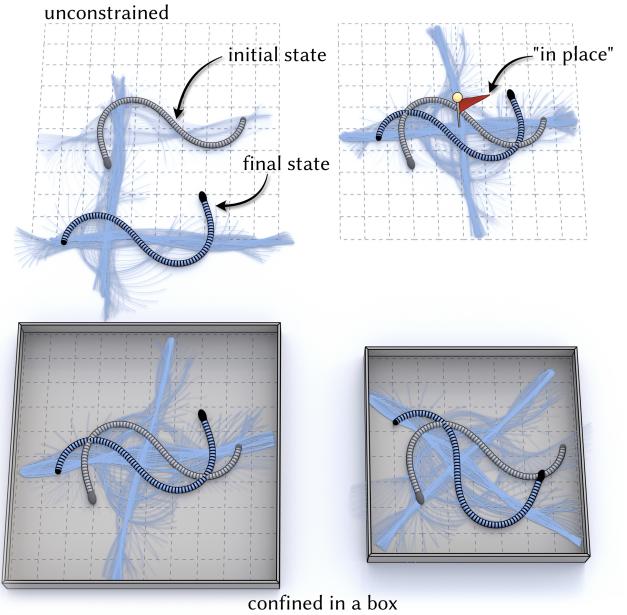


Fig. 8. Gait discovery for a turning snake. The objective is to find a sequence of non-self-intersecting body deformations such that the corresponding motion leads to a net rotation of the snake by 180°. Significantly different pose sequences are found when using no additional constraints (top left), when penalizing displacement of the centroid (top right), or when confining the snake to a box (bottom left). If the box is too small, no solution is found that does not violate the collision penalty (bottom right). See video at 00:58.

displacement of the snake's center of mass com such as

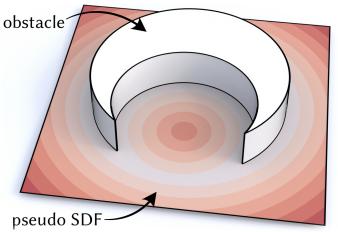
$$\mathcal{J}_{\text{com}}(\zeta) = \frac{1}{2} \int_0^T |\text{com}(\zeta^0) - \text{com}(\zeta^t)|^2 dt.$$

As illustrated in Figure 8 (top right), adding this penalty leads to a qualitatively different motion matching the user's preference.

*Collision avoidance.* In addition to satisfying the user's design intent, we also define objective functions to ensure the physical validity of the resulting motion.

In particular, we want to avoid self-intersections of the locomotor. Figure 8 showcases a rudimentary self-avoidance objective term, where pairwise vertex distances are penalized using an Incremental Potential Contact (IPC) barrier [Li et al. 2020]. While IPC requires Continuous Collision Detection (CCD) to guarantee self-intersection free trajectories, our optimization can find solutions that are collision-free by sufficiently increasing the range of action of the IPC barrier.

Moreover, we want to avoid collisions with obstacles in the environment. We account for obstacles by means of an implicit potential barrier  $c: \mathbb{R}^3 \rightarrow \mathbb{R}$  whose 0-level set represents the boundary of the obstacle. We also assume that  $c$  satisfies the eikonal equation almost everywhere, making it



a *pseudo Signed Distance Function* (pseudo SDF). The admissible space for the motion trajectory is now given by the set  $c^{-1}(\mathbb{R}^+)$ , while  $c^{-1}(\mathbb{R}^-)$  represents the obstacle.

A collision occurs when the positioned shape intersects the obstacle, that is, when  $c(g_t(x)) \leq 0$  for some  $x \in \varsigma^t$  at a time  $t$ . Then, the average constraint violation of a positioned shape at instantaneous time  $t$  is given by

$$C_t(\zeta) = \frac{1}{\text{Vol}(\varsigma_t)} \int_{\varsigma^t} \max(0, -c(g_t(x))) d\varsigma^t(x),$$

whose time average

$$\mathcal{J}_{\text{col}}(\zeta) = \frac{1}{T} \int_0^T C_t(\zeta) dt$$

is added as a penalty term to the optimization problem. Our method's ability to incorporate collision avoidance in this way is illustrated in Figures 1 and 8.

*Pose control.* In some inverse design scenarios, we not only care about the final motion trajectory, but also require direct control over the pose sequence. For example, when optimizing for the complex sequence of shape deformations for a cat's "self-righting reflex," it is essential that the animal lands on its legs in a specific pose to cushion the landing (Figure 9). To this end, we can constrain the start and points of the map  $\varsigma: [0, T] \rightarrow \mathcal{S}$ , thus fixing the start and end pose of the shape-changing body (Figures 7 and 9).

Optimizing Equation 4 for cyclic gaits, constitutes an important special case of this constraint. Here, we constrain the shape sequence  $\varsigma: [0, T] \rightarrow \mathcal{S}$  to have  $\varsigma^0 = \varsigma^T$ , thus giving identical start and end poses (Figure 5).

Moreover, we can provide additional control over the resulting shape sequence by specifying a reference sequence that encodes certain pose semantics that should be preserved by the optimization. In order to guide the optimization with the given shape sequence, we define a soft penalty term

$$\mathcal{J}_{\text{guide}}(\zeta) = \frac{1}{2} \int_0^T |\varsigma^{*,t} - \varsigma^t|_{\mathcal{S}}^2 dt, \quad (7)$$

where  $|\cdot|_{\mathcal{S}}$  denotes a norm on the shape space. In Figure 9 we demonstrate the efficacy of this approach, where our optimized solution retains the semantics of a given input sequence, while achieving the desired target transformation.

### 3.2 Parametrizing shape spaces

In order to solve for shape sequences that satisfy the user-specified pose and motion targets, we need to parametrize the underlying shape space. In general, standard geometry representations such as triangle meshes do not directly provide a suitable parameterization, since no notion of deformation semantics is encoded in the vertex coordinates. To make our inverse design optimization more well-posed, shape deformations should be encoded in a suitable rig, where ideally all combinations of free parameters lead to semantically correct shapes [Loper et al. 2023].

Conceptually, we parametrize the set of admissible shapes by a differentiable map  $\phi: \mathcal{P} \rightarrow \mathcal{S}$  from a parameter space  $\mathcal{P} \subset \mathbb{R}^p$  to the shape space  $\mathcal{S}$  (Figure 10). The corresponding inverse problem

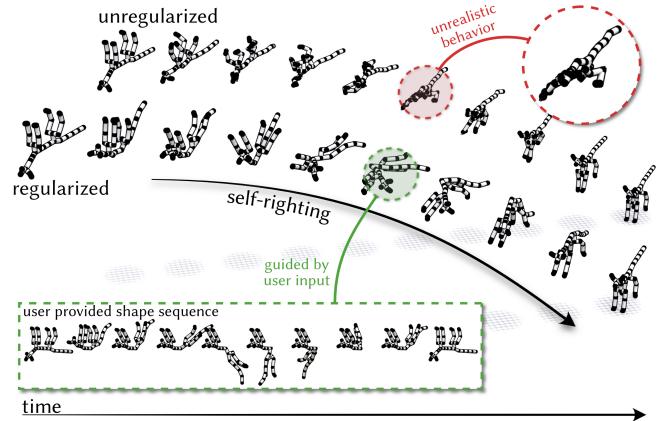


Fig. 9. Modeling a cat's self-righting reflex requires a shape sequence that achieves a desired target rotation in a forward simulation. Our method allows users to guide the optimization output with semantics provided in the form of an unregistered shape sequence to mitigate unrealistic behavior stemming from, e.g., insufficiently regular shape spaces. For the presented experiments, the vertex positions as the degrees of freedom are paired with an isometry constraint on the edge lengths. See video at 01:49.

then takes the form

$$\underset{\rho: [0,T] \rightarrow \mathcal{P}}{\text{argmin}} \mathcal{J}(\zeta) \quad \text{s.t. } H(\phi \circ \rho) = \zeta. \quad (8)$$

In the following, we briefly discuss specific choices of shape space parameterizations that we employed for the experiments throughout this paper.

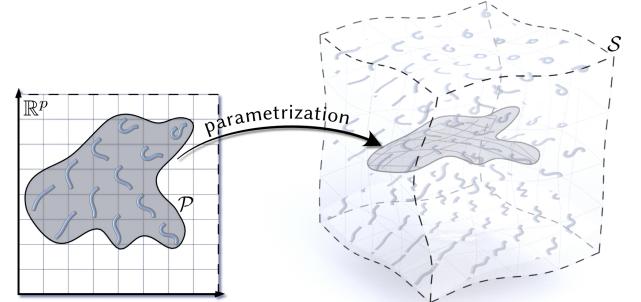


Fig. 10. Parametrizing semantically meaningful or regular parts of the shape space allows reducing the dimensionality of the inverse problems, thus stabilizing the optimization and improving the results.

*Serpentoid shape space.* The space of *serpentoid curves* [Hirose 1993] captures a remarkably large fraction of the shapes observed in undulating locomotors such as snakes and nematodes [Rieser et al. 2024] (Figures 1, 8 and 17). Serpentoid curves are plane curves, determined in terms of their curvature function

$$\kappa(s, t) = w_1(t) \sin(2\pi\xi s) + w_2(t) \cos(2\pi\xi s), \quad (9)$$

which, by the fundamental theorem of plane curves, uniquely determines the curves' shape up to a rigid body transformation [Pinkall

and Gross 2024]. Here, the coordinate  $w(t) = [w_1(t), w_2(t)]$  of the shape in the shape space determines the coefficients, while  $\xi > 0$  is the spatial frequency of body undulation.

**Linkage mechanisms.** Certain types of mechanisms that are composed of linkages of rigid pieces typically come with a natural parameterization of their shape spaces. For example, the hinged mechanisms of Figures 11, 16, 18 and 19 can be parameterized by the dihedral angles of the hinge joints, which automatically ensures that shape elements remain rigid. However, such a parameterization does not exclude self-intersecting configurations, which would need to be treated separately, e.g., through appropriately coupled bound constraints.

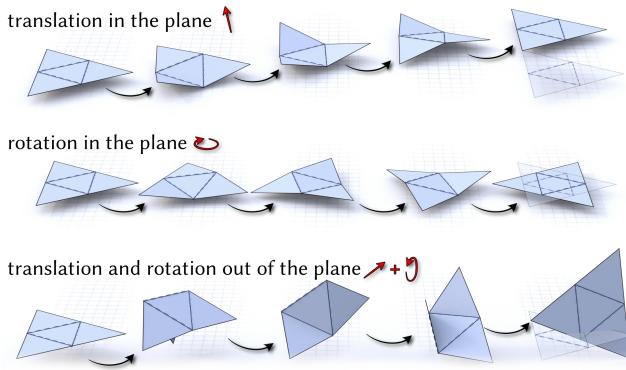


Fig. 11. Cyclic motion discovery on a very simple “Origami” shape composed of four rigid triangles that are connected by rotational hinges along their shared edges. See video at 02:12.

**Point handles.** A common strategy for describing shape deformations is to manipulate a sparse set of point handles, from which the deformed shape is then obtained by interpolation. These methods come in many facets, typically minimizing the variational energies of the deformed geometry [Botsch and Kobbelt 2004; Sorkine and Alexa 2007; Soliman et al. 2024b] or in cells of deformation cages [Joshi et al. 2007; Dodik et al. 2023]. In Figure 12, we demonstrate the compatibility of these approaches with our proposed method.

**Modal bases.** Another common approach to parameterizing reduced shape spaces is to represent the deformed states of a rest shape in terms of a sparse set of modal basis functions [Hildebrandt et al. 2011; Benchekroun et al. 2024]. To demonstrate the compatibility of our method with this type of approach, we use eigenvectors corresponding to the smallest 12 eigenvalues of the Laplace-Beltrami operator [Pinkall and Polthier 1993] as a basis for each of the three coordinate functions to represent deformed shapes of the sphere (Figure 12).

Alternatively, we can use eigenvectors of the Hessian of a membrane energy around the rest position as the modal basis [Hildebrandt et al. 2012]. This allows localizing the deformations by modifying the associated membrane stiffness, as shown in Figures 14 and 20.

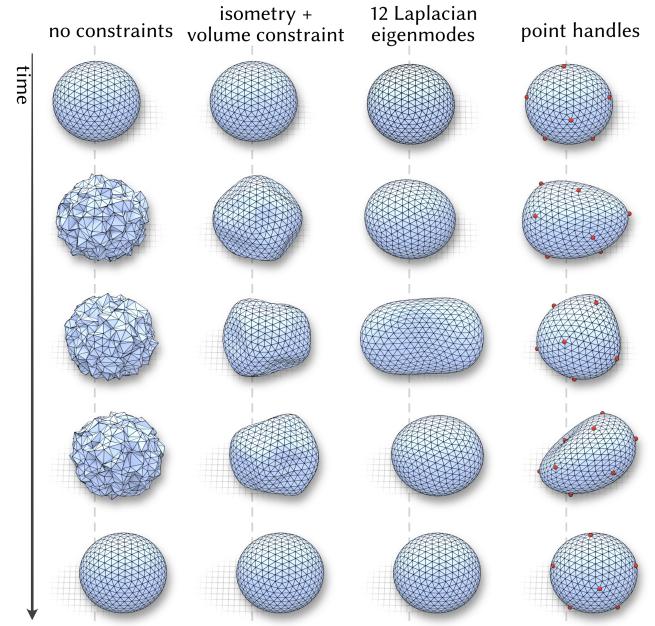


Fig. 12. A comparison of different shape space parameterizations for a spherical object. All examples achieve the same target displacement to the right, but with largely differing pose sequences. With no constraints, all vertex coordinates are free parameters, leading to a rather chaotic motion. We obtain significantly smoother deformations when adding soft constraints that regularize the deformation to remain approximately isometric to the spherical rest state and preserve its initial volume. Laplacian eigenmodes or point handles also lead to more well-behaved deformations with the additional benefit of substantially reducing the dimensionality of the underlying shape space. See video at 02:33.

## 4 METHOD OVERVIEW

For the computational treatment of the inverse geometric locomotion problem we first discretize the continuous equations in space and time (Figure 13), before deriving a practical algorithm to solve them numerically. As we build upon the computational framework of Gross et al. [2023], we briefly review the relevant details. For a more comprehensive exposition we refer to the original reference.

### 4.1 Discretization

We discretize the configuration space  $\mathcal{M}$  of all shapes in all positions as  $\mathcal{M} := \mathbb{R}^{3m}$ , where each instantaneous state

$$(\zeta_0^\top, \dots, \zeta_m^\top)^\top \in \mathcal{M},$$

is represented by a collection of  $m$  vertices  $\zeta_j \in \mathbb{R}^3$ ,  $j \in \{0, \dots, m\}$ . Consequently, the discrete shape space is given by the quotient  $S := \mathcal{M}/SE(3)$ .

For better readability and more concise expressions we will mimic notation from the smooth setup and also use the notation  $\zeta \in S^n$  resp.  $\zeta \in \mathcal{M}^n$  for time-discrete sequences  $(\zeta^1, \dots, \zeta^n) \in S^n$  in the shape space  $S$  resp.  $(\zeta^1, \dots, \zeta^n) \in \mathcal{M}^n$  in the configuration space  $\mathcal{M}$ , whenever the context allows for it.

*Discrete variational energy.* Discrete analogues of the variational energies we consider are given by quadratic forms

$$\mathbb{E}(\zeta) = \frac{1}{2} \sum_{i=1}^{n-1} \langle \Delta\zeta^{(i,i+1)}, \Delta\zeta^{(i,i+1)} \rangle_M, \quad (10)$$

where  $\Delta\zeta^{(i,i+1)} := \zeta^{i+1} - \zeta^i$  and  $\langle \cdot, \cdot \rangle_M$  is an  $SE(3)$ -invariant Riemannian metric on  $M$ . We specify the metric

$$\langle \cdot, \cdot \rangle_M = \langle B^{(i,i+1)}, \cdot \rangle_{\mathbb{R}^{3m}} \quad (11)$$

by a block-diagonal matrix  $B^{(i,i+1)} \in \mathbb{R}^{3m,3m}$ , with  $m$  blocks of the form

$$B_j^{(i,i+1)} := \frac{1}{2}(B_j^i + B_j^{i+1}) \in \mathbb{R}^{3,3},$$

which are assembled from symmetric and positive definite *local metric tensors*  $B_j^i \in \mathbb{R}^{3,3}$  associated to each vertex  $\zeta_j^i$ .

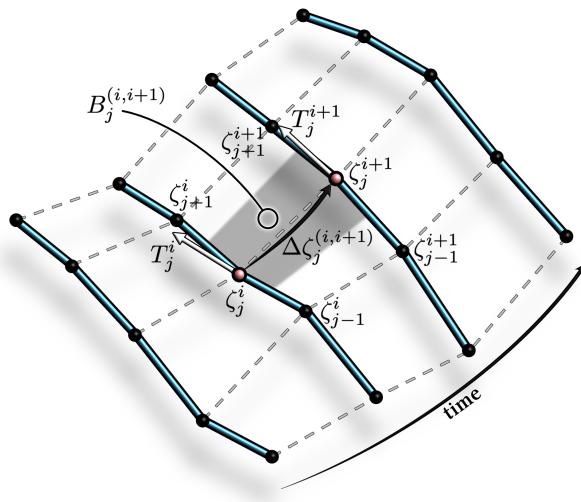


Fig. 13. A schematic overview of the time and space discretization of a lift  $\zeta$ . The tangent vectors (white) are the preferred displacement directions of  $\zeta_j^i$  resp.  $\zeta_j^{i+1}$  (red) with respect to the local dissipation metric Equation 13. By measuring the dissipation caused by displacements  $\Delta\zeta_j^{(i,i+1)}$  in each time step (grey shaded area), these local dissipation energies are aggregated into a total dissipation energy over the shape sequence.

*Physical model parameters.* The choice of local metric tensors determines the physical scenario that is modeled. For example, blocks of the form

$$B_j^i = m_j I \quad (12)$$

model the kinetic energy of a shape-changing body with masses  $m_j > 0$  lumped at its vertex positions. For all experiments involving isotropic local metric tensors, we assign unit mass to each vertex, allowing the mass distribution to be directly inferred from the vertex distribution.

Following *resistive force theory* [Gray and Hancock 1955; Zhang and Goldman 2014], we approximate the total dissipation based on velocity and tangential drag on individual body elements, thus neglecting global interactions. To account for effects of dissipative

anisotropies that, e.g., a swimmer immersed in a viscous medium experiences, we assign a unit vector to each vertex. These vectors are determined solely by the geometry of the shapes. Specifically, for curves, we assign a unit tangent vector  $T_j^i \in \mathbb{S}^2$  and define

$$B_j^i = w_j^i(I + (\epsilon - 1)T_j^i \otimes T_j^i) \in \mathbb{R}^{3,3}, \quad (13)$$

with integration weights  $w_j^i > 0$  and *anisotropy ratio*  $\epsilon \in (0, 1]$ , which models the relative ease of the displacement of a vertex of a polygonal curve in the direction of the tangent vector. Analogously, for surfaces, we assign a unit normal vector  $N_j^i \in \mathbb{S}^2$  to each vertex  $\zeta_j^i$  and define

$$B_j^i = w_j^i(\epsilon I + (1 - \epsilon)N_j^i \otimes N_j^i) \in \mathbb{R}^{3,3}. \quad (14)$$

With this approach, all bodies—including snakes slithering on the ground—are formally modeled as if fully immersed in a viscous medium. While this is a significant simplification, it proves effective in practice, even for capturing the essential characteristics of slithering locomotion. For dissipation-dominated scenarios, we determined the weights  $w_j^i$  proportional to the size of the Voronoi cell around each vertex. Typical choices for the anisotropy parameter were  $\epsilon = 0.1$  for curve-based examples and  $\epsilon \in [0.001, 0.01]$  for surface meshes.

The choice of a Riemannian metric on the total space according to Equation 13 and Equation 14 allows for the modeling of a wide range of physical scenarios [Gross et al. 2023]. An overview of all physical model parameters used in our experiments is provided in Table 1.

*Integrating shapes into motion.* Together with the input shapes and an initial momentum  $\mu_0$ , the physical model parameters uniquely determine motion trajectories up to a global rigid body transformation. Therefore, there is a unique horizontal lift with  $g_0 = \text{Id} \in SE(3)$ , integrating the shape sequence starting from the position of the first shape in the common reference frame.

Analogous to the continuous setup, this lift is determined by the condition that the discrete momentum

$$\mu(\zeta^{i-1}, \zeta^i) := \begin{pmatrix} -\frac{1}{2} \sum_k \zeta_j^i \times (B_j^{i-1} \Delta\zeta_k^{(i-1,i)}) + \zeta_j^{i-1} \times (B_j^i \Delta\zeta_j^{(i-1,i)}) \\ - \sum_k B_j^{(i-1,i)} \Delta\zeta_j^{(i-1,i)} \end{pmatrix} \quad (15)$$

of the lift  $\zeta$  vanishes at each time step. Given two consecutive shapes, this can in practice be achieved by solving for roots of

$$SE(3) \rightarrow \mathbb{R}^6, g_i \mapsto \mu(\zeta^{i-1}, g_i(\zeta^i)) - \mu_0, \quad (16)$$

where  $\mu_0 \in \mathbb{R}^6$  represents an initial momentum (Algorithm 1). By construction, such *variational integrators* exhibit a number of advantageous properties as they are automatically symplectic, exhibit good energy behavior for exponentially long times and are momentum preserving [Marsden and West 2001; Leok 2015], making them highly effective for computing stable trajectories. All our experiments assume the bodies to be initially at rest, which implies  $\mu_0 \equiv 0$ .

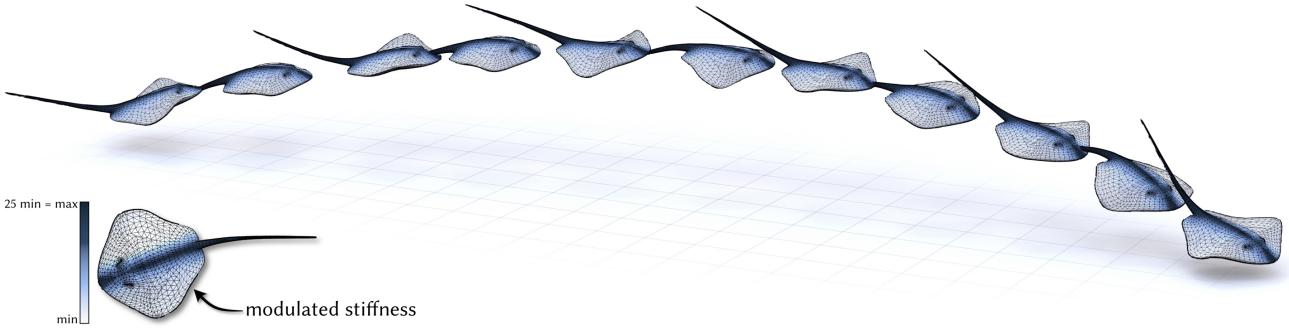


Fig. 14. Optimization of a stingray’s gait. Deformations of the stingray’s body are constrained to a reduced linear modal basis composed of the first 40 vibrational modes of a membrane energy with modulated stiffness as color-coded. The stingray must achieve a net curved trajectory in the most efficient manner as measured by a combination of the membrane energy and the total energy dissipation (Equation 10). The optimized cyclic gait is repeated 7 times and exhibits a natural swimming behavior. See video at 03:24.

---

**Algorithm 1 – IntegrateTrajectory [Gross et al. 2023]**


---

**Input:** shape sequence  $\zeta \in S^n$ , metric tensors  $(B^1, \dots, B^n)$ , initial momentum  $\mu_0 \in \mathbb{R}^6$   
**Output:** horizontal section  $\zeta \in M^n$

- 1:  $\zeta^1 \leftarrow \text{Id}(\zeta^1)$
- 2: **for**  $t = 2, \dots, n$  **do**
- 3:    $g_i \leftarrow \text{solve } \mu(\zeta^{i-1}, g_i(\zeta^i)) - \mu_0 = 0$               ▶ Equation 16
- 4:    $\zeta^i \leftarrow g_i(\zeta^i)$
- 5: **end for**

---

## 4.2 Optimization algorithm

In order to solve inverse problems as in Equation 4, we must compute the derivatives of a generic objective function  $\mathcal{J}$  with respect to the input shapes  $\zeta$ . We consider the physical model parameters to be fixed, which still leaves us with the discrete shape sequence—and a total of  $3nm$  degrees of freedom—as variables.

**THEOREM 4.1.** Let  $\zeta \in M^n$  be a discrete curve in the configuration space  $M$  and  $\mathcal{J}: M^n \rightarrow \mathbb{R}_{\geq 0}$  be a generic objective function (Equation 3). Then, for  $i \in \{1, \dots, n\}$ ,

$$\frac{d\mathcal{J}}{d\zeta^i} = w_{i+1}^\top \frac{\partial \mu_{(i,i+1)}}{\partial \zeta^i} + w_i^\top \frac{\partial \mu_{(i-1,i)}}{\partial \zeta^i} + \frac{\partial \mathcal{J}}{\partial \zeta^i} \frac{\partial \zeta^i}{\partial \zeta^i}, \quad (17)$$

where the adjoint vectors  $w_1, \dots, w_n \in \mathbb{R}^6$  are given by

$$\left( \frac{\partial \mu_{(n-1,n)}}{\partial g_n} \right)^\top w_n = - \left( \frac{\partial \mathcal{J}}{\partial g_n} \right)^\top, \quad (18)$$

respectively, the recursively defined linear systems for  $i \in \{n-1, \dots, 1\}$ ,

$$\left( \frac{\partial \mu_{(i-1,i)}}{\partial g_i} \right)^\top w_i = - \left( \frac{\partial \mu_{(i,i+1)}}{\partial g_i} \right)^\top w_{i+1} - \left( \frac{\partial \mathcal{J}}{\partial g_i} \right)^\top. \quad (19)$$

**PROOF.** See Appendix A. □

Note that, due to the explicit use of configuration space symmetries, the resulting form of our governing ODE—and consequently the definition of the adjoint vectors—differs from those presented in, e.g., Bordalba et al. [2023], Coros et al. [2012], or Pan and Manocha [2018]. Theorem 4.1 allows us to efficiently compute the gradient of

the objective function with respect to the input shapes  $\zeta$  in Equation 17 by solving a sequence of  $6 \times 6$  linear systems in Equation 18 and Equation 19. Initial shape sequences are updated according to an L-BFGS algorithm [Byrd et al. 1995] until sufficiently small relative variations in the objective are observed, or when the gradient norm is lower than a prescribed threshold<sup>4</sup>. We summarize our complete optimization in Algorithm 2.

To employ this type of quasi-Newton method, the objective  $\mathcal{J}$  for the minimization is ideally required to be a  $C^2$  function. However, in practice our method complies with functions which are merely in  $C^2$  almost everywhere (see, e.g., Figure 1). The optimization timings and associated experimental settings can be found in Table 1.

---

**Algorithm 2 – OptimizeShapeSequence**


---

**Input:** objective  $\mathcal{J}: M^n \rightarrow \mathbb{R}_{\geq 0}$ , initial guess  $\zeta \in S^n$   
**Output:** optimal shape sequence  $\zeta^* \in S^n$

- 1: **while** not Converged( $\mathcal{J}(\zeta)$ ) **do**
- 2:    $\zeta \leftarrow \text{IntegrateTrajectory}(\zeta)$               ▶ Algorithm 1
- 3:    $L_n, r_n \leftarrow \left( \frac{\partial \mu_{(n-1,n)}}{\partial g_n} \right)^\top, - \left( \frac{\partial \mathcal{J}}{\partial g_n} \right)^\top$
- 4:    $w_n \leftarrow \text{Solve}(L_n w_n = r_n)$               ▶ Equation 18
- 5:    $\frac{d\mathcal{J}}{d\zeta^n} \leftarrow w_n^\top \frac{\partial \mu_{(n-1,n)}}{\partial \zeta^n} + \frac{\partial \mathcal{J}}{\partial \zeta^n} \frac{\partial \zeta^n}{\partial \zeta^n}$               ▶ Equation 17
- 6:   **for**  $i = n-1, \dots, 1$  **do**
- 7:      $L_i, r_i \leftarrow \left( \frac{\partial \mu_{(i-1,i)}}{\partial g_i} \right)^\top, - \left( \frac{\partial \mu_{(i,i+1)}}{\partial g_i} \right)^\top w_{i+1} - \left( \frac{\partial \mathcal{J}}{\partial g_i} \right)^\top$
- 8:      $w_i \leftarrow \text{Solve}(L_i w_i = r_i)$               ▶ Equation 19
- 9:      $\frac{d\mathcal{J}}{d\zeta^i} \leftarrow w_{i+1}^\top \frac{\partial \mu_{(i,i+1)}}{\partial \zeta^i} + w_i^\top \frac{\partial \mu_{(i-1,i)}}{\partial \zeta^i} + \frac{\partial \mathcal{J}}{\partial \zeta^i} \frac{\partial \zeta^i}{\partial \zeta^i}$               ▶ Equation 17
- 10:   **end for**
- 11:    $d \leftarrow \text{L-BFGS}(\zeta, \frac{d\mathcal{J}}{d\zeta})$               ▶ descent direction
- 12:    $\zeta^i \leftarrow \text{Linesearch}(\mathcal{J}, \zeta, d)$
- 13: **end while**

---

The stability of our inverse optimization algorithm crucially depends on the accuracy of the derivatives of the objective function,

<sup>4</sup>We refer to the accompanying code for details.

which requires exactly solving the root-finding problem in Equation 16. We found rigidly registering the sequence of input shapes  $\zeta$  to a common reference object (Figures 1, 8, 9, 12, 18, 21, 20) reliably leads the optimization to terminate at a local minimum. Other shape spaces naturally rule out rigid transformation and supply a canonical frame of reference such as the vibrational modes of a membrane (Figure 14) or  $SE(3)$ -equivariant deformations (Figure 2, Figure 11, Figure 16, Figure 19), which performed equally well in our experiments.

#### 4.3 Reducing the degrees of freedom

The number of degrees of freedom for a simulation with Algorithm 2 is generally large, which often makes inverse problems ill-posed (see, e.g., the left most columns of Figure 12 or Figure 20). In this section, we describe two approaches to mitigate this issue and drastically reduce the number of degrees of freedom for inverse problems.

*Discrete reduced shape spaces.* Given a discrete shape space  $S$ , we parametrize a *discrete reduced shape space* by a differentiable map  $\phi: P \rightarrow S$ , where  $P \subset \mathbb{R}^p$  is a low-dimensional parameter space (Figure 10). The corresponding inverse problems then take the form in Equation 8 and we default to PyTorch’s reverse mode automatic differentiation [Paszke et al. 2019] to account for the additional concatenation with  $\phi$  for the gradient computation in Algorithm 2.

As discussed in Section 3.2, a reduced shape space not only significantly lowers the number of degrees of freedom whenever  $p \ll 3N$ , but also acts as a regularization that can encode important shape deformation semantics (Figure 12).

*Space-time interpolations.* We navigate the configuration space leveraging its fibered structure. The global positioning of a deforming object emerges from the conservation of geometric momentum. Therefore, our approach contrasts with methods where trajectories are traced in configuration space directly [Pan and Manocha 2018].

Assuming fairly smooth temporal shape transitions, we can represent a shape sequence in the discrete reduced shape space  $\phi(P)$  using fewer interpolating shapes (see also, e.g., [Heeren et al. 2016]). Consequently, we view the discrete shape sequence  $\phi(\rho) \in \phi(P)^n$  as a sampling of a continuous map  $\phi \circ \rho: [0, T] \rightarrow \phi(P)$  coming

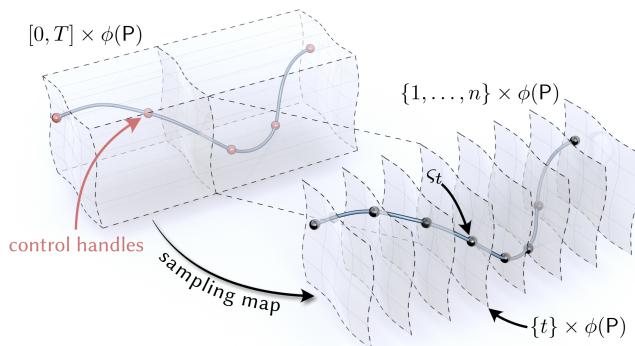


Fig. 15. Splines allow for a continuous description of shape sequences  $\zeta: [0, T] \rightarrow S$  with few degrees of freedom and can be sampled to obtain corresponding discrete sequences  $\zeta \in S^n$ .

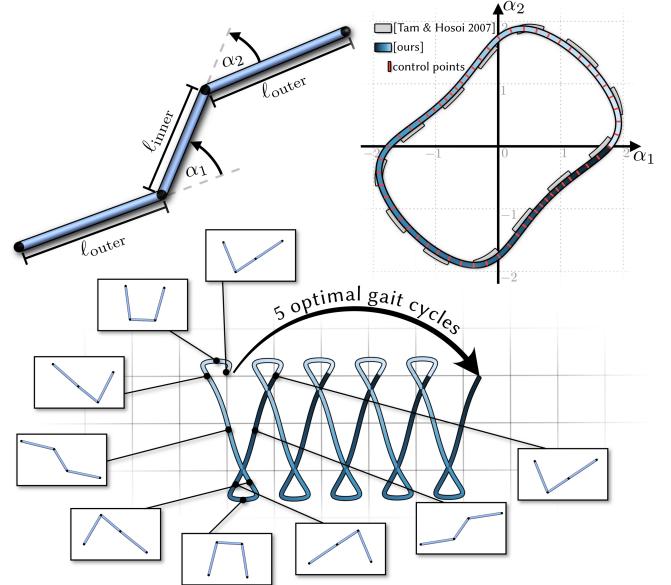


Fig. 16. With two degrees of freedom, the *Purcell’s swimmer* (top left) is the simplest model of a swimmer that can achieve a net displacement by means of periodic shape sequences, or “gaits”. The optimal displacement gait found by our method closely resembles the known optimal solution by Tam and Hosoi [2007] (top right). We show the displacement of the center edge for five gait cycles (bottom). See video at 03:40.

from a *sampling map* (Figure 15)

$$s: ([0, T] \rightarrow S) \rightarrow \{1, \dots, n\} \times S \cong S^n, \zeta \mapsto (\zeta^1, \dots, \zeta^T).$$

We parameterize the underlying continuous section as an interpolating spline with interpolating shapes  $(\phi(\rho^1), \dots, \phi(\rho^\ell))$ , where  $(\rho^1, \dots, \rho^\ell) \in P^\ell$  are control points. These control points serve as design variables for the inverse problem, which reads

$$\operatorname{argmin}_{\rho \in P^\ell} \mathcal{J}(\zeta) \quad \text{s.t. } H(s(\phi \circ \rho)) = \zeta,$$

where we use the short-hand notation introduced in Section 4.1. The map  $\rho: [0, T] \rightarrow P$  is a cubic spline interpolating the control points  $\rho^1, \dots, \rho^\ell \in P$ .

In practice, we use the differentiable cubic spline implementation provided by [Kidger 2017], which can be combined with PyTorch’s reverse mode automatic differentiation to backpropagate the gradient through the interpolation operation. This approach allows reducing the number of degrees of freedom in all our presented experiments, while enabling to solve for smooth closed gaits.

By construction, our setup ensures that the equations of motion are satisfied at every timestep thanks to the condition of geometric momentum preservation. Other methods often only weakly enforce the equations of motions using soft penalty terms [Pan and Manocha 2018], or enforce them at a discrete set of timesteps called collocation points in collocation methods [Bordalba et al. 2023], with no guarantees for the intervals in between.

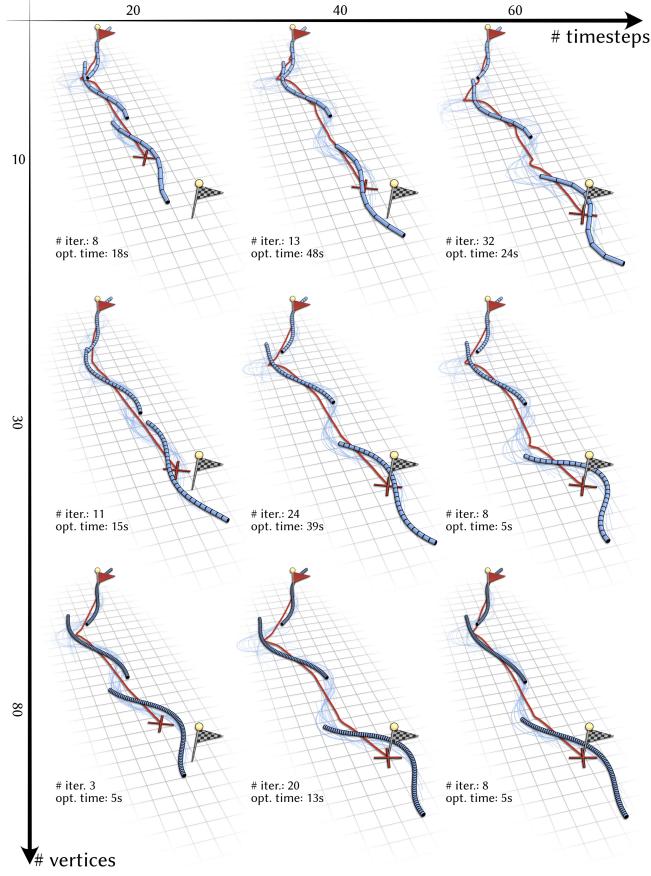


Fig. 17. Effect of the resolution of our space-time discretization on a shape parameterized by the serpenoid function. The red and checkered flags indicate the start position and desired target position, respectively. The red curve shows the resulting net displacement of the shape's centroid, with the cross denoting the final achieved position. See video at 04:03.

## 5 RESULTS

In this section, we analyze and validate our computational pipeline, before presenting several example applications focusing on both the exploration and the optimization of shape sequences and gaits. We also discuss limitations of our approach and suggest directions for future work.

### 5.1 Performance, validation, and consistency

All presented experiments were computed with our Python implementation, which is available for download at <https://go.epfl.ch/igl/>. Performance statistics computed on an Apple Macbook Pro with a M1 Max CPU are given in Table 1.

As analytical results on optimal gait strategies of geometric locomotors are generally scarce, we resort to the simplest<sup>5</sup> model, *Purcell's swimmer*, that can achieve a net displacement by means of

<sup>5</sup>Although the model for two scallops on a stick in Figure 2 (also known as the *symmetrized cousin* of Purcell's swimmer [Kadam and Banavar 2016]) also relies on merely two degrees of freedom, this model consists of more edges than Purcell's swimmer.

periodic shape sequences to validate our optimization algorithm. The optimal displacement gait found by our method closely resembles the known optimal solution by Tam and Hosoi [2007] while achieving 99.7% of the displacement (Figure 16).

Figure 17 summarizes the effects of the space-time discretization on the optimization results, illustrating a trade-off between computational efficiency and accuracy of the optimized motion sequence.

Notably, the energy dissipation rates of optimal trajectories regularized by the variational energy Equation 10 are almost constant, hinting at the fact that our optimization not only by construction returns lifts tangent to the horizontal distributions, but in fact closely approximates *subriemannian geodesics*—locally shortest lifts among all lifts tangent to the horizontal distribution—in the configuration space  $M$  equipped with the dissipation metric given in Equation 11 [Montgomery 2002]. Natural dynamical systems can ubiquitously be described as geodesics [Frankel 2011] and, as shown in Figures 1, 14, 18 and 19, for our applications as well, motion trajectories regularized this way exhibit a natural look and feel.

### 5.2 Motion exploration and discovery

A key benefit of our inverse approach is that we can effectively discover locomotion strategies when only provided with a deformation shape space. Figure 11 illustrates how we can discover gaits that displace an abstract locomotor with three degrees of freedom in different ways. These motions were initialized with small random perturbations of the constant initial state to break symmetry and thus generated effectively “from nothing.”

This ability is particularly beneficial in applications such as robotics, where motion systems can be explored solely through knowledge of kinematic properties. As demonstrated in Figure 18, we can explore gaits of a snake-like robot and automatically adapt pose sequences to account for unforeseen events, such as a broken joint segment.

Our method also allows for gait adaptation based on different constraints on the shape space, such as symmetries. In Figure 2, our optimization successfully exploits collaborative efforts, without which no net displacement is possible. However, when comparing the two scallops on a stick in Figure 2 to the swimming robot in Figure 19 with similar morphology, we note that having four degrees of freedom significantly increases its maneuverability, allowing the displacement along a curved trajectory.

The ability to explore and discover locomotion strategies is not limited to kinematic systems, but also carries over to, e.g., reduced shape spaces, which we demonstrate in the studies shown in Figures 12 and 20, eventually allowing for more complex animation tasks. Although a stingray’s locomotion in water is a corner case for the strict applicability of geometric locomotion, our algorithm managed to discover a realistically looking periodic shape sequence mainly concentrated around the animals pectoral fins, which yields a curved trajectory (Figure 14). The optimization was performed on a reduced shape space spanned by 40 vibrational modes of a membrane energy with modulated stiffness as described in Section 3.2. This animation task combines methods from Figures 12 and 20 as well as Figures 10 and 15.

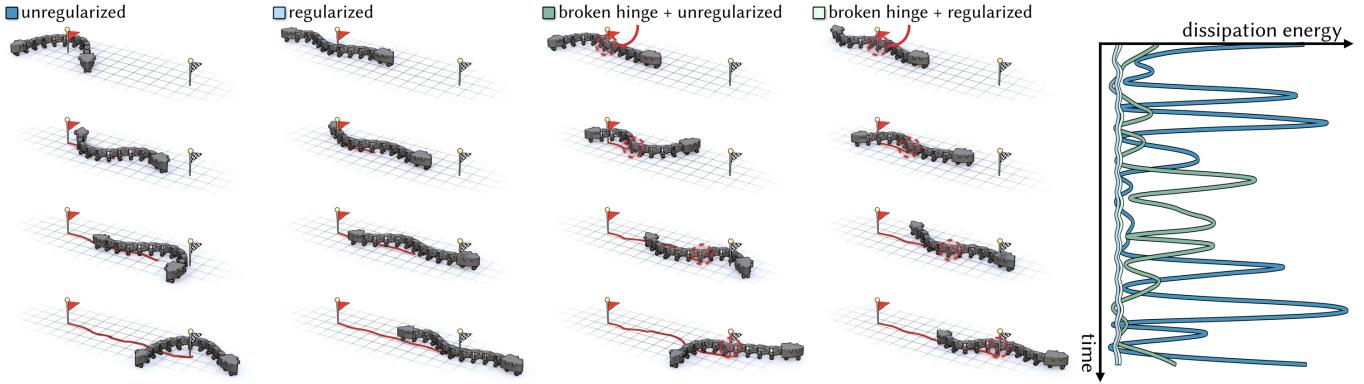


Fig. 18. Motion trajectories of a snake-like robot slithering to a predetermined target position are compared to trajectories obtained with an additional regularization by the total energy dissipation. While the robot in the *first* and *second* column can freely move all its hinge segments, admissible shapes of the robot in the *third* and *fourth* column are limited by a broken hinge segment which remains in a position with a fixed angle of  $\pi/4$ . The graph on the right depicts energy dissipation over time. See video at 04:41.

### 5.3 Shape sequence and gait optimization

A particularly suitable application for our approach is the optimization of given gaits or shape sequences to better satisfy their motion objectives. For example, our approximation of the optimal gait of Purcell’s swimmer (Figure 16) was initialized with a naive circular gait, before our method successfully optimized it to closely approximate its known maximal displacement gait (Section 5.1).

Similarly, Figure 21 shows a skeleton-based animation of an Armadillo astronaut in an inertia dominated zero-gravity scenario, which was reproduced from [Gross et al. 2023]. While maintaining similar motion semantics, our optimization increased the amount of rotation achieved in a single gait cycle by a factor of five. However, since the approach is agnostic of physical poses, the optimization can be pushed beyond the physically possible, which can eventually lead to non-realistic behavior such as self-intersections.

### 5.4 Manipulation and constraints

Avoiding non-realistic behavior typically requires an appropriate choice of reduced shape space, taking into account the semantics of

physical motion [Loper et al. 2023; Sassen et al. 2024]. However, the variety of objectives introduced in Section 3.1 can already act as effective regularizations, allowing us to manipulate motion trajectories either provided by an animator or initially obtained through automatic motion discovery (Section 5.2). For example, objectives of the form of Equation 6 cause the trajectory to pass through certain waypoints (Figure 1), while objectives such as in Equation 7 cause it to conform to certain semantics designed by an animator (Figure 9). Figure 8 shows how a snake adapts its gait to achieve a half-turn while satisfying environmental constraints materialized by a box.

So far we have disregarded how the shape changes are realized in practice. In robotic applications, where battery power is at a premium and force or torque constraints must be met, the efficiency of locomotion strategies is of particular importance. Similar principles apply when modeling a shape-changing body in a highly damped environment, such as a snake slithering on sand. Not only does its displacement through the granular medium dissipate energy, but also its metabolism consumes energy to perform the deformations.

A common approach to account for these notions of internal energy dissipation in the absence of motions is to associate physical properties such as elasticity with shape-changing characters. Geodesics in shape spaces with respect to a Riemannian metric have been successfully computed in the context of shape deformation and character animation [Heeren et al. 2014, 2016; Sassen et al. 2024], based on an elastic membrane energy. Notably, Hartwig et al. [2025] propose a choice of Riemannian metric on the configuration space that intrinsically captures both inner and outer energy dissipation. We demonstrate how an additional regularization with an elastic membrane energy can be applied in the present setup to effectively bias the optimization result (Figure 20). Moreover, in Figure 18, we illustrate how an additional regularization with Equation 10 can significantly reduce the total energy dissipated by a snake-like robot—even when admissible deformations are hindered by a broken hinge.

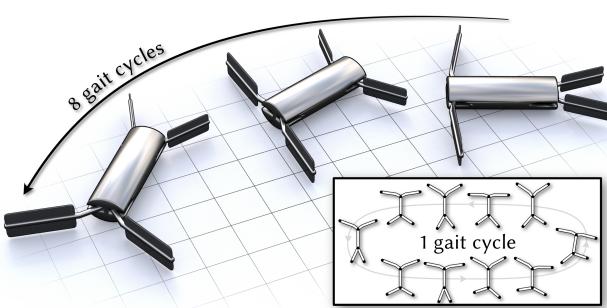


Fig. 19. An abstract robot in a highly damped environment employing its four degrees of freedom to perform an asymmetric swimming gait to achieve displacement and rotation along a curved trajectory. See video at 05:32.

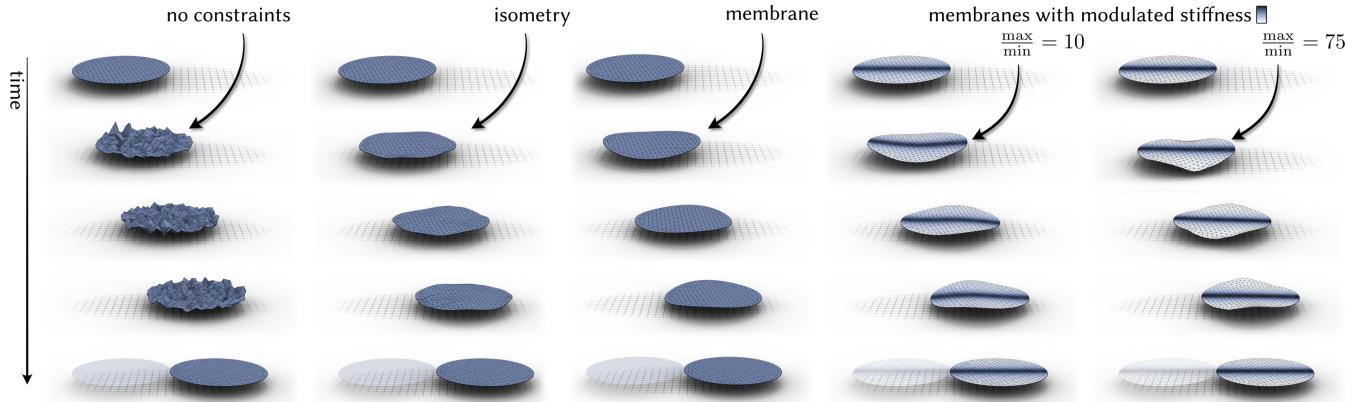


Fig. 20. Different constraints and regularizations allow for the exploration of a variety of gaits which all lead to the discs' displacement by one diameter. For the depicted motion trajectories the degrees of freedom are given by the vertex coordinates, while the optimization was run with no constraints, an isometry constraint and regularizations by a membrane energy using different bending stiffness distributions (from left to right). See video at 05:56.

### 5.5 Limitations and future work

We have demonstrated the effectiveness of our inverse geometric locomotion algorithm on a diverse set of examples and applications. However, our approach still has a number of limitations.

First, building on the framework of Gross et al. [2023], the proposed approach inherits the same benefits and limitations, such as neglecting the dynamics of participating media. As an example of neglected phenomena, vortices that naturally form in water and air are, in many ways, essential for efficient aquatic and aerial locomotion in high Reynolds number regimes.

Moreover, our treatment of constraints in terms of soft penalty terms generally leads to trade-offs between different components of the objective, thus potentially allowing for small violations of, e.g., collision constraints (see, e.g., the bottom right in Figure 8). Additionally, as with all highly nonlinear problems, there is a possibility

that our optimization gets stuck in a local minimum that fails to satisfy the motion objective (Figure 22).

In these cases, it is typically impossible to tell if the optimization failed because of an unrealistic target objective with no feasible solution, or if modeling choices, such as reduced shape space or the design of a penalty term, are too restricting (see, e.g., Figure 17).

In general, the choice of an appropriate shape space parameterization remains a challenging task. A shape space parameterization needs to be sufficiently expressive to allow for a solution to exist. However, too many degrees of freedom may destabilize the optimization and lead to sub-optimal termination. For example, in Figure 22 we explore the influence of different bound constraints on the wavelength for the serpenoid shape space (Section 3.2) on the convergence of our optimization for Figure 1. To mitigate trade-offs introduced by regularizing the shape space and/or enforcing semantics by means of soft constraints, a promising avenue for future work is to extend our method by incorporating semantic constraints for, e.g., collision avoidance directly in the shape spaces as in [Loper et al. 2023; Sassen et al. 2024].

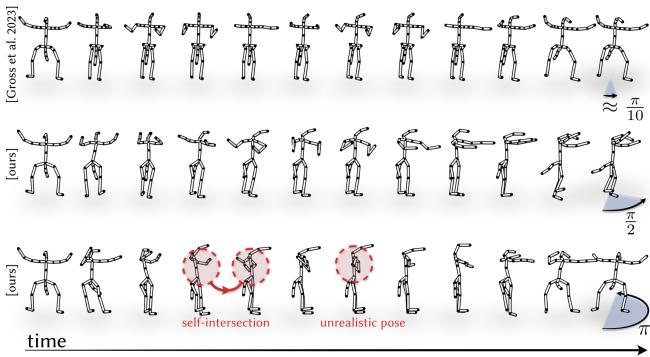


Fig. 21. A naive turning gait of an armadillo astronaut reproduced from [Gross et al. 2023] (top row) is compared to a gait optimized by our method (middle row), showing possible performance differences despite visual similarity. Unrealistic behavior, i.e., configurations that exhibit self-intersections or defy expected motion semantics may occur when the target rotation goal is set too high (bottom row). See video at 06:58.

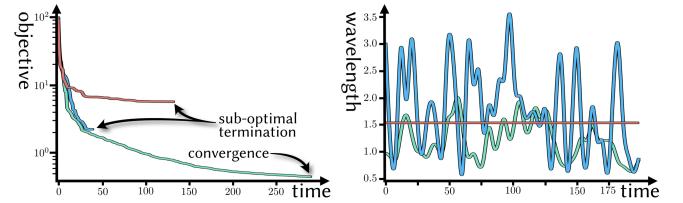


Fig. 22. The values of the objective function over time steps of the optimization of the experiment shown in Figure 1 by our method (left) and the corresponding values of the wavelength parameters (right) are displayed for three different experiments, which are distinguished by constraints on the wavelengths parameter (red: constant, green: bound constraints, blue: unconstrained). Notably, bounds that are too restrictive or too permissive may lead to sub-optimal convergence of the optimization. The result of the green graphs is displayed in Figure 1.

In our current setting, we optimize for shape deformations, but assume that the corresponding morphology, *i.e.*, the rest shape of the moving body is given as input. Fascinating questions arise on how to formulate a co-optimization that simultaneously finds optimal morphologies and pose sequences to best achieve given target objectives (see, *e.g.*, [Wampler and Popović 2009]). Such a method could potentially aid studies in evolutionary biology as well as support robotic designs with new motion capabilities.

## 6 CONCLUSION

We present an efficient computational framework for addressing a wide range of inverse geometric locomotion problems. Our method serves as both a powerful animation tool and a versatile system for exploring locomotion strategies in shape-changing bodies, as demonstrated through a diverse set of examples. It provides users with intuitive controls to design and manipulate motion trajectories for animation, while in robotics it facilitates the discovery and optimization of gaits and enables the simulation and analysis of potential failure scenarios, thereby enhancing robustness and reliability.

## ACKNOWLEDGMENTS

The rattlesnake mesh is courtesy of DigitalLife3D at the University of Massachusetts, Amherst, the robot snake mesh is courtesy of the MiniRo Lab at the University of Notre Dame, and the wood stick mesh is courtesy of 3.99 SHOP. This research was supported by the Swiss National Science Foundation (Grant 200021-231293). Additional support was provided through Houdini software, courtesy of SideFX.

## REFERENCES

- M. Alexa, D. Cohen-Or, and D. Levin. 2000. As-rigid-as-possible shape interpolation. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*. ACM Press/Addison-Wesley Publishing Co., 157–164.
- A. Aristidou, J. Lasenby, Y. Chrysanthou, and A. Shamir. 2018. Inverse Kinematics Techniques in Computer Graphics: A Survey. *Comp. Graph. Forum* 37, 6 (2018), 35–58.
- J. Barbić, M. da Silva, and J. Popović. 2009. Deformable object animation using reduced optimal control. In *ACM SIGGRAPH 2009 Papers* (New Orleans, Louisiana) (*SIGGRAPH '09*). Association for Computing Machinery, Article 53, 9 pages.
- M. Bauer, P. Heslin, and C. Maor. 2024. Completeness and geodesic distance properties for fractional Sobolev metrics on spaces of immersed curves. *J. Geom. Anal.* 34, 7 (2024), 214.
- L. E. Becker, S. A. Koehler, and H. A. Stone. 2003. On self-propulsion of micro-machines at low Reynolds number: Purcell's three-link swimmer. *J. Fl. Mech.* 490 (2003), 15–5.
- O. Benchekroun, K. Xie, Hsueh-Ti Liu, E. Grinspan, S. Andrews, and V. Zordan. 2024. Actuators A La Mode: Modal Actuations for Soft Body Locomotion. In *SIGGRAPH Asia 2024 Conference Papers*. Article 53, 10 pages.
- Z. Bing, L. Cheng, K. Huang, and A. Knoll. 2022. Simulation to Real: Learning Energy-Efficient Slithering Gaits for a Snake-Like Robot. 29, 4 (2022), 92–103.
- R. Bordalba, T. Schoels, L. Ros, J. M. Porta, and M. Diehl. 2023. Direct Collocation Methods for Trajectory Optimization in Constrained Robotic Systems. 39, 1 (2023), 183–202.
- M. Botsch and L. Kobelt. 2004. An intuitive framework for real-time freeform modeling. In *ACM SIGGRAPH 2004 Papers* (Los Angeles, California) (*SIGGRAPH '04*). 630–634.
- C. Brandt, E. Eisemann, and K. Hildebrandt. 2018. Hyper-reduced projective dynamics. *ACM Trans. Graph.* 37, 4, Article 80 (July 2018), 13 pages.
- R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu. 1995. A Limited Memory Algorithm for Bound Constrained Optimization. *SIAM J. Sci. Comp.* 16, 5 (1995), 1190–1208.
- B. Chong, T. Wang, L. Bo, S. Li, P. C. Muthukrishnan, J. He, D. Irvine, H. Choset, G. Blekherman, and D. I. Goldman. 2023. Optimizing contact patterns for robot locomotion via geometric mechanics. *I. J. Rob. Res.* 42, 10 (2023), 859–873.
- S. Coros, S. Martin, B. Thomaszewski, C. Schumacher, R. Sumner, and M. Gross. 2012. Deformable objects alive! *ACM Trans. Graph.* 31, 4, Article 69 (2012), 9 pages.
- A. Dodik, O. Stein, V. Sitzmann, and J. Solomon. 2023. Variational Barycentric Coordinates. *ACM Trans. Graph.* 42, 6, Article 255 (2023), 16 pages.
- T. Du, J. Hughes, S. Wah, W. Matusik, and D. Rus. 2021a. Underwater Soft Robot Modeling and Control With Differentiable Simulation. *IEEE Robot. Autom. lett.* 6, 3 (2021), 4994–5001. <https://doi.org/10.1109/LRA.2021.3070305>
- T. Du, K. Wu, P. Ma, S. Wah, A. Spielberg, D. Rus, and W. Matusik. 2021b. DiffPD: Differentiable Projective Dynamics. *ACM Trans. Graph.* 41, 2, Article 13 (2021), 21 pages. <https://doi.org/10.1145/3490168>
- L. Euler. 1744. *Methodus Inveniendi Lineas Curvas Maximi Minivi Proprietate Gaudentes*. Bousquet, Lausanne & Geneva. English translation at WikiSource.
- T. Frankel. 2011. *The Geometry of Physics: An Introduction* (3 ed.). CUP.
- M. Geilinger, D. Hahn, J. Zehnder, M. Bächer, B. Thomaszewski, and S. Coros. 2020. ADD: analytically differentiable dynamics for multi-body systems with frictional contact. *ACM Trans. Graph.* 39, 6, Article 190 (2020), 15 pages.
- L. Giraldi, P. Martinon, and M. Zoppello. 2015. Optimal design of Purcell's three-link swimmer. *Phys. R. E* 91 (Feb 2015), 023012. Issue 2.
- P. Goel and D. L. James. 2022. Unified Many-Worlds Browsing of Arbitrary Physics-Based Animations. *ACM Trans. Graph.* 41, 4 (2022), 156:1–15.
- J. Gray and G. J. Hancock. 1955. The Propulsion of Sea-Urchin Spermatozoa. *J. Exp. Biol.* 32, 4 (1955), 802–814.
- O. Gross, Y. Soliman, M. Padilla, F. Knöppel, U. Pinkall, and P. Schröder. 2023. Motion from Shape Change. *ACM Trans. Graph.* 42, 4 (2023), 107:1–11.
- E. Hartman, E. Pierson, M. Bauer, N. Charon, and M. Daoudi. 2023a. Bare-esa: A riemannian framework for unregistered human body shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14181–14191.
- E. Hartman, Y. Sukurdeep, E. Klassen, N. Charon, and M. Bauer. 2023b. Elastic shape analysis of surfaces with second-order sobolev metrics: a comprehensive numerical framework. *Int. J. Comput. Vis.* 131, 5 (2023), 1183–1209.
- F. Hartwig, O. Gross, M. Rumpf, and P. Schröder. 2025. Optimal motion from shape change. *Oberwolfach Reports* 5/2025, 15–18.
- F. G. Harvey, M. Yurick, De. Nowrouzezahrai, and C. Pal. 2020. Robust motion in-betweening. *ACM Trans. Graph.* 39, 4, Article 60 (2020), 12 pages.
- B. Heeren, M. Rumpf, P. Schröder, M. Wardetzky, and B. Wirth. 2014. Exploring the Geometry of the Space of Shells. In *Proceedings of the Symposium on Geometry Processing* (Cardiff, UK) (*SGP '14*). Eurographics Association, 247–256.
- B. Heeren, M. Rumpf, P. Schröder, M. Wardetzky, and B. Wirth. 2016. Splines in the Space of Shells. *Comp. Graph. Forum* (2016).
- K. Hildebrandt, C. Schulz, c. von Tycowicz, and K. Polthier. 2012. Modal shape analysis beyond Laplacian. *Comput. Aided Geom. Des.* 29, 5 (2012), 204–218. Geometric Modeling and Processing 2010.
- K. Hildebrandt, C. Schulz, C. von Tycowicz, and K. Polthier. 2011. Interactive surface modeling using modal analysis. *ACM Trans. Graph.* 30, 5, Article 119 (2011), 11 pages.
- S. Hirose. 1993. *Biologically Inspired Robots (Snake-like Locomotor and Manipulator)*. Oxford University Press.
- J. K. Hodgins, W. L. Wooten, D. C. Brogan, and J. F. O'Brien. 1995. Animating human athletics. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*. Association for Computing Machinery, 71–78.
- Z. Huang, D. C. Tozoni, A. Gjoka, Z. Ferguson, T. Schneider, D. Panizzo, and D. Zorin. 2024. Differentiable solver for time-dependent deformation problems with contact. *ACM Trans. Graph.* 43, 3, Article 31 (2024), 30 pages.
- S. Jia, S. Wang, T.-M. Li, and A. Chern. 2023. Physical Cyclic Animations. *Proc. ACM Comput. Graph. Interact. Tech.* 6, 3, Article 45 (2023), 18 pages.
- P. Joshi, M. Meyer, T. DeRose, B. Green, and T. Sanocki. 2007. Harmonic coordinates for character articulation. *ACM Trans. Graph.* 26, 3 (2007), 71–es.
- S. Kadam and R. N. Banavar. 2016. Geometric Controllability of The Purcell's Swimmer and its Symmetrized Cousin. *IFAC-PapersOnLine* 49, 18 (2016), 988–993. 10th IFAC Symposium on Nonlinear Control Systems NOLCOS 2016.
- S. Kang, X. Xu, J. Sarva, L. Liang, and H. Yang. 2024. Fast and Certifiable Trajectory Optimization. arXiv:2406.05846
- L. Kavan and O. Sorkine. 2012. Elasticity-inspired deformers for character articulation. *ACM Trans. Graph.* 31, 6, Article 196 (2012), 8 pages.
- P. Kidger. 2017. torchcubicspline. <https://github.com/patrick-kidger/torchcubicspline> Last accessed 17 January 2025.
- M. Kobilarov, K. Crane, and M. Desbrun. 2009. Lie Group Integrators for Animation and Control of Vehicles. *ACM Trans. Graph.* 28, 2 (2009), 16:1–14.
- N. Kwatra, C. Wojtan, M. Carlson, I. A. Essa, P. J. Mucha, and G. Turk. 2010. Fluid Simulation with Articulated Bodies. *IEEE Trans. Vis. Comp. Graph.* 16, 1 (2010), 70–80.
- M. Leok. 2015. Variational Integrators. In *Encyclopedia of Applied and Computational Mathematics*, Björn Engquist (Ed.). Springer, 1519–1525.
- Minchen Li, Zachary Ferguson, Teseo Schneider, Timothy Langlois, Denis Zorin, Daniele Panizzo, Chenfanfu Jiang, and Danny M. Kaufman. 2020. Incremental potential contact: intersection-and inversion-free, large-deformation dynamics. 39, 4, Article 49 (2020), 20 pages. <https://doi.org/10.1145/3386569.3392425>
- S. Li, T. Wang, V. H. Kojouharov, J. McInerney, E. Aydin, Y. Ozkan-Aydin, D. I. Goldman, and D. Z. Rocklin. 2022. Robotic swimming in curved space via geometric phase. *Proc. Nat. Acad. Sci.* 119, 31 (2022), e2200924119.

Table 1. Experimental data of the gait optimizations (*from left to right*): Anisotropy ratio  $\epsilon$ , whether the optimized gait is periodic, number of timesteps in the simulation, number of vertices of the shape, number of control points in the spline representation, number of reduced degrees of freedom, and the time it took to optimize the shape sequence. If multiple values are provided, the order follows experiments from left to right then top to bottom of the corresponding figure.

Exp. (Figure)	$\epsilon$	closed gait	#timesteps	#vertices	#CPs	#DOFs	optim. time (min:sec)
Snake Parkour (1)	0.08	yes	200	30	50	3	11:13
Two Clams on a Stick (2)	0.01	yes	200	156	15	2	02:34
Turning Snakes (8)	0.1	yes	100	80	20	3	11:36, 42:36, 03:05, 02:26
Falling Cat (9)	1	yes	100	36	15	108	15:37, 15:34
Triangle Origami (11)	0.01	yes	100	66	20	3	00:36, 00:18, 01:22
Moving Spheres (12)	0.01	yes	80	642	15	1926, 1926, 36, 36	01:33, 16:06, 15:42, 19:08
Stingray (14)	0.0001	yes	100	1792	15	40	46:43
Purcell's Swimmer (16)	0.5	yes	200	154	30	2	00:29
Broken Robot (18)	0.1	no	100	11	16	10, 10, 9, 9	00:12, 11:53, 00:13, 08:18
Crawling Robot (19)	0.01	yes	100	190	15	4	07:09
Disk (20)	0.01	yes	80	423	15	1269	00:50, 18:26, 18:07, 17:05, 16:51
Astronaut (21)	1	yes	150	25	25	75	25:10, 07:33

- M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. 2023. *SMPL: A Skinned Multi-Person Linear Model* (1 ed.). Association for Computing Machinery, New York, NY, USA.
- J. E. Marsden and T. S. Ratiu. 1999. *Introduction to Mechanics and Symmetry* (2 ed.). Springer New York, NY.
- J. E. Marsden and M. West. 2001. Discrete Mechanics and Variational Integrators. *Act. Num.* 10 (May 2001), 357–514.
- S. Martin, B. Thomaszewski, E. Grinspun, and M. Gross. 2011. Example-based elastic materials. In *ACM SIGGRAPH 2011 Papers*. Association for Computing Machinery, New York, NY, USA, Article 72, 8 pages.
- S. Min, J. Won, S. Lee, J. Park, and J. Lee. 2019. SoftCon: Simulation and Control of Soft-Bodied Animals with Biomimetic Actuators. *ACM Trans. Graph.* 38, 6 (2019), 208:1–12.
- R. Montgomery. 2002. *A tour of subriemannian geometries, their geodesics and applications*. Number 91 in Mathematical surveys and monographs. Amer. Math. Soc.
- M. S. Nabizadeh, R. Roy-Chowdhury, H. Yin, R. Ramamoorthi, and A. Chern. 2024. Fluid Implicit Particles on Coadjoint Orbits. *ACM Trans. Graph.* 43, 6, Article 270 (2024), 38 pages.
- R. Newbury, J. Collins, K. He, J. Pan, I. Posner, D. Howard, and A. Cosgun. 2024. A Review of Differentiable Simulators. *IEEE Access* 12 (2024), 97581–97604.
- J. Ostrowski and J. Burdick. 1998. The Geometric Mechanics of Undulatory Robotic Locomotion. *I. J. Rob. Res.* 17, 7 (1998), 683–701.
- M. Padilla, A. Chern, F. Knöppel, U. Pinkall, and P. Schröder. 2019. On Bubble Rings and Ink Chandeliers. *ACM Trans. Graph.* 38, 4 (2019), 129:1–14.
- D. Palmer, A. Chern, and J. Solomon. 2024. Lifting Directional Fields to Minimal Sections. *ACM Trans. Graph.* 43, 4, Article 60 (2024), 20 pages. <https://doi.org/10.1145/3658198>
- Z. Pan and D. Manocha. 2018. Active Animations of Reduced Deformable Models with Environment Interactions. *ACM Trans. Graph.* 37, 3, Article 36 (2018), 17 pages.
- A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. 2019. *PyTorch: an imperative style, high-performance deep learning library*. Curran Associates Inc., Red Hook, NY, USA.
- U. Pinkall and O. Gross. 2024. *Differential Geometry: From Elastic Curves to Willmore Surfaces*. Birkhäuser.
- U. Pinkall and K. Polthier. 1993. Computing Discrete Minimal Surfaces and Their Conjugates. *Exp. Math.* 2, 1 (1993), 15–36.
- E. M. Purcell. 1977. Life at Low Reynolds Number. *Amer. J. Phys.* 45, 3 (1977), 3–11.
- J. Qin, Y. Zheng, and K. Zhou. 2022. Motion In-Betweening via Two-Stage Transformers. *ACM Trans. Graph.* 41, 6, Article 184 (2022), 16 pages.
- S. Ramasamy and R. L. Hatton. 2019. The geometry of optimal gaits for drag-dominated kinematic systems. *IEEE Trans. Rob.* 35, 4 (2019), 1014–1033.
- Nathan Ratliff, Matt Zucker, J. Andrew Bagnell, and Siddhartha Srinivasa. 2009. CHOMP: Gradient optimization techniques for efficient motion planning. In *2009 IEEE International Conference on Robotics and Automation*, 489–494.
- J. M. Rieser, B. Chong, C. Gong, H. C. Astley, P. E. Schiebel, K. Diaz, C. J. Pierce, H. Lu, R. L. Hatton, H. Choset, and D. I. Goldman. 2024. Geometric phase predicts locomotion performance in undulating living systems across scales. *Proc. Nat. Acad. Sci.* 121, 24 (2024), e2320517121.
- J. Sassen, H. Schumacher, M. Rumpf, and K. Crane. 2024. Repulsive Shells. *ACM Trans. Graph.* 43, 4 (2024).
- E. A. Shammas, H. Choset, and A. A. Rizzi. 2007. Geometric motion planning analysis for two classes of underactuated mechanical systems. *I. J. Rob. Res.* 26, 10 (2007), 1043–1073.
- A. Shapere and F. Wilczek. 1989a. Geometry of Self-Propulsion at Low Reynolds Number. *J. Fluid Mech.* 198 (1989), 557–585.
- A. Shapere and F. Wilczek. 1989b. Efficiencies of Self-Propulsion at Low Reynolds Number. *J. Fl. Mech.* 198 (1989), 587–599.
- A. Shapere and F. Wilczek. 1989c. Gauge Kinematics of Deformable Bodies. *Amer. J. Phys.* 57, 6 (1989), 514–518.
- Y. Soliman, M. Padilla, O. Gross, F. Knöppel, U. Pinkall, and P. Schröder. 2024a. Going with the Flow. *ACM Trans. Graph.* 43, 4 (2024), 57:1–12.
- Y. Soliman, U. Pinkall, and P. Schröder. 2024b. Conformal surface splines. *Differ. Geom. Appl.* 97 (2024), 102200.
- O. Sorkine and M. Alexa. 2007. As-Rigid-As-Possible Surface Modeling. In *Geometry Processing*, Alexander Belyaev and Michael Garland (Eds.). The Eurographics Association.
- D. Tam and A. E. Hosoi. 2007. Optimal Stroke Patterns for Purcell's Three-Link Swimmer. *Phys. R. Lett.* 98 (Feb 2007), 068105. Issue 6.
- J. Tan, Y. Gu, G. Turk, and C. K. Liu. 2011. Articulated swimming creatures. *ACM Trans. Graph.* 30, 4, Article 58 (2011), 12 pages.
- H. von Helmholz. 1882. Zur Theorie der stationären Ströme in reibenden Flüssigkeiten. In *Wissenschaftliche Abhandlungen*. Vol. I. J. A. Barth, 223–230.
- K. Wampler and Z. Popović. 2009. Optimal gait and form for animal locomotion. *ACM Trans. Graph.* 28, 3, Article 60 (2009), 8 pages. <https://doi.org/10.1145/1531326.1531366>
- S. Weißmann and U. Pinkall. 2012. Underwater Rigid Body Dynamics. *ACM Trans. Graph.* 31, 4 (2012), 104:1–7.
- A. Witkin and M. Kass. 1988. Spacetime constraints. *SIGGRAPH Comput. Graph.* 22, 4 (1988), 159–168.
- W. L. Wooten and J. K. Hodgins. 1996. Animation of Human Diving. *Comp. Graph. Forum* 15, 1 (1996), 3–13.
- J.-C. Wu and Z. Popović. 2003. Realistic Modeling of Bird Flight Animations. *ACM Trans. Graph.* 22, 3 (2003), 888–895.
- J. Z. Zhang, Y. Zhang, P. Ma, E. Nava, T. Du, P. Arm, W. Matusik, and R. K. Katzschmann. 2022. Sim2Real for Soft Robotic Fish via Differentiable Simulation. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 12598–12605.
- T. Zhang and D. I. Goldman. 2014. The effectiveness of resistive force theory in granular locomotion. *Phys. Fluids* 26, 10 (2014), 101308. <https://doi.org/10.1063/1.4898629>
- D. Zhao, B. Bittner, G. Clifton, N. Gravish, and S. Revzen. 2022. Walking is like slithering: A unifying, data-driven view of locomotion. *Proc. Nat. Acad. Sci.* 119, 37 (2022), e2113222119.

## A PROOF OF THEOREM 4.1

We start out by proving a lemma, which provides us with some helpful identities. For brevity, we will use the notation  $\mu_{(i-1,i)} := \mu(\zeta^{i-1}, \zeta^i)$ .

LEMMA 1. *It holds*

$$\frac{dg_t}{d\zeta^i} = - \left( \frac{\partial \mu_{(t-1,t)}}{\partial g_t} \right)^{-1} \left( \frac{\partial \mu_{(t-1,t)}}{\partial \zeta^i} + \frac{\partial \mu_{(t-1,t)}}{\partial g_{t-1}} \frac{dg_{t-1}}{d\zeta^i} \right).$$

PROOF. Since solutions of the forward problem are characterized by a constant  $\mu_{(t-1,t)}$  ( $(\zeta^{t-1}, g_{t-1}(\zeta^{t-1})), (\zeta^t, g_t(\zeta^t))$ ), we can differentiate to obtain

$$0 = \frac{d\mu_{(t-1,t)}}{d\zeta^i} = \frac{\partial \mu_{(t-1,t)}}{\partial \zeta^i} + \frac{\partial \mu_{(t-1,t)}}{\partial g_{t-1}} \frac{dg_{t-1}}{d\zeta^i} + \frac{\partial \mu_{(t-1,t)}}{\partial g_t} \frac{dg_t}{d\zeta^i},$$

which yields the claim after re-arranging the terms.  $\square$

COROLLARY 1. *In the setup of Lemma 1, we obtain following simplified special cases:*

(1) *When  $2 \leq t \leq n$  and  $i = t - 1$ , it holds*

$$\frac{dg_t}{d\zeta^{t-1}} = - \left( \frac{\partial \mu_{(t-1,t)}}{\partial g_t} \right)^{-1} \left( \frac{\partial \mu_{(t-1,t)}}{\partial \zeta^{t-1}} + \frac{\partial \mu_{(t-1,t)}}{\partial g_{t-1}} \frac{dg_{t-1}}{d\zeta^{t-1}} \right). \quad (20)$$

(2) *When  $1 \leq t \leq n$  and  $i = t$ , it holds*

$$\frac{dg_t}{d\zeta^t} = - \left( \frac{\partial \mu_{(t-1,t)}}{\partial g_t} \right)^{-1} \frac{\partial \mu_{(t-1,t)}}{\partial \zeta^t}. \quad (21)$$

(3) *When  $t = 1$  and  $i = 0$ , it holds*

$$\frac{dg_1}{d\zeta^0} = - \left( \frac{\partial \mu_{(0,1)}}{\partial g_1} \right)^{-1} \frac{\partial \mu_{(0,1)}}{\partial \zeta^0}. \quad (22)$$

(4) *When  $2 \leq t \leq n$  and  $0 \leq i \leq t - 2$ , it holds*

$$\frac{dg_t}{d\zeta^i} = - \left( \frac{\partial \mu_{(t-1,t)}}{\partial g_t} \right)^{-1} \frac{\partial \mu_{(t-1,t)}}{\partial g_{t-1}} \frac{dg_{t-1}}{d\zeta^i}. \quad (23)$$

LEMMA 2. *With the vectors  $w_1, \dots, w_n \in \mathbb{R}^6$  defined as in Theorem 4.1, it holds*

$$\frac{\partial \mathcal{J}}{\partial g_n} \frac{dg_n}{d\zeta^n} = w_n^\top \frac{\partial \mu_{(n-1,n)}}{\partial \zeta^n},$$

$$\frac{\partial \mathcal{J}}{\partial g_n} \frac{dg_n}{d\zeta^{n-1}} = w_n^\top \left( \frac{\partial \mu_{(n-1,n)}}{\partial \zeta^{n-1}} + \frac{\partial \mu_{(n-1,n)}}{\partial g_{n-1}} \frac{dg_{n-1}}{d\zeta^{n-1}} \right),$$

$$\frac{\partial \mathcal{J}}{\partial g_n} \frac{dg_n}{d\zeta^i} = w_n^\top \frac{\partial \mu_{(n-1,n)}}{\partial g_{n-1}} \frac{dg_{n-1}}{d\zeta^i} \text{ for } i \in \{0, \dots, n-2\},$$

and, for  $i \in \{0, \dots, n-2\}$ ,

$$\left( w_{i+2}^\top \frac{\partial \mu_{(i+1,i+2)}}{\partial g_{i+1}} + \frac{\partial \mathcal{J}}{\partial g_{i+1}} \right) \frac{dg_{i+1}}{d\zeta^i} = w_{i+1}^\top \left( \frac{\partial \mu_{(i,i+1)}}{\partial \zeta^i} + \frac{\partial \mu_{(i,i+1)}}{\partial g_i} \frac{dg_i}{d\zeta^i} \right),$$

and, for all  $t \in \{2, \dots, n-1\}$  and  $i \in \{0, \dots, t-2\}$ ,

$$\left( w_{t+1}^\top \frac{\partial \mu_{(t,t+1)}}{\partial g_t} + \frac{\partial \mathcal{J}}{\partial g_t} \right) \frac{dg_t}{d\zeta^i} = w_t^\top \frac{\partial \mu_{(t-1,t)}}{\partial g_{t-1}} \frac{dg_{t-1}}{d\zeta^i}.$$

PROOF. All three equations immediately follow from plugging in the results from Cor. 1 for  $\frac{dg_n}{d\zeta^i}$  resp.  $\frac{dg_t}{d\zeta^i}$  and the definition of the adjoint vectors  $w_1, \dots, w_n \in \mathbb{R}^6$ .  $\square$

With these formulas in place, we are now in a position to prove Theorem 4.1.

PROOF OF THEOREM 4.1. We consider an objective function of the form  $\mathcal{J}(\zeta)$ , where  $\zeta = (\zeta^1, \dots, \zeta^n)$  with  $\zeta^i(g_i(\zeta^i), \zeta^i)$  for each  $i \in \{1, \dots, n\}$ . Its derivative with respect to the  $i$ -th shape is therefore given by

$$\frac{d\mathcal{J}}{d\zeta^i} = \sum_{t=i}^n \frac{\partial \mathcal{J}}{\partial \zeta^t} \frac{d\zeta^t}{d\zeta^i} = \sum_{t=i}^n \left( \frac{\partial \mathcal{J}}{\partial g_t} \frac{dg_t}{d\zeta^i} + \frac{\partial \mathcal{J}}{\partial \zeta^t} \frac{\partial \zeta^t}{d\zeta^i} \right).$$

Notably, for each summand with  $k \neq i$ , the term  $\frac{\partial \zeta^k}{d\zeta^i}$  vanishes, since by definition  $\zeta^k(g_k(\zeta^k), \zeta^k)$ . Therefore, by recursively applying the identities of Lemma 2, when  $i \in \{1, \dots, n-2\}$ , we find

$$\begin{aligned} \frac{d\mathcal{J}}{d\zeta^i} &= \frac{\partial \mathcal{J}}{\partial g_n} \frac{dg_n}{d\zeta^i} + \sum_{t=i}^{n-1} \left( \frac{\partial \mathcal{J}}{\partial g_t} \frac{dg_t}{d\zeta^i} \right) + \frac{\partial \mathcal{J}}{\partial \zeta^i} \frac{\partial \zeta^i}{d\zeta^i} \\ &= \left( w_n^\top \frac{\partial \mu_{(n-1,n)}}{\partial g_{n-1}} \right) \frac{dg_{n-1}}{d\zeta^i} + \sum_{t=i}^{n-2} \left( \frac{\partial \mathcal{J}}{\partial g_t} \frac{dg_t}{d\zeta^i} \right) + \frac{\partial \mathcal{J}}{\partial \zeta^i} \frac{\partial \zeta^i}{d\zeta^i} \\ &= w_{n-1}^\top \frac{\partial \mu_{(n-2,n-1)}}{\partial g_{n-2}} \frac{dg_{n-2}}{d\zeta^i} + \sum_{t=i}^{n-2} \left( \frac{\partial \mathcal{J}}{\partial g_t} \frac{dg_t}{d\zeta^i} \right) + \frac{\partial \mathcal{J}}{\partial \zeta^i} \frac{\partial \zeta^i}{d\zeta^i} \\ &\vdots \\ &= \left( w_{i+2}^\top \frac{\partial \mu_{(i+2,i+1)}}{\partial g_{i+1}} + \frac{\partial \mathcal{J}}{\partial g_{i+1}} \right) \frac{dg_{i+1}}{d\zeta^i} + \frac{\partial \mathcal{J}}{\partial g_i} \frac{dg_i}{d\zeta^i} + \frac{\partial \mathcal{J}}{\partial \zeta^i} \frac{\partial \zeta^i}{d\zeta^i} \\ &= w_{i+1}^\top \frac{\partial \mu_{(i,i+1)}}{\partial \zeta^i} + \left( w_{i+1}^\top \frac{\partial \mu_{(i,i+1)}}{\partial g_i} + \frac{\partial \mathcal{J}}{\partial g_i} \right) \frac{dg_i}{d\zeta^i} + \frac{\partial \mathcal{J}}{\partial \zeta^i} \frac{\partial \zeta^i}{d\zeta^i} \\ &= w_{i+1}^\top \frac{\partial \mu_{(i,i+1)}}{\partial \zeta^i} + w_i^\top \frac{\partial \mu_{(i-1,i)}}{\partial \zeta^i} + \frac{\partial \mathcal{J}}{\partial \zeta^i} \frac{\partial \zeta^i}{d\zeta^i}. \end{aligned}$$

For the remaining cases, we find

$$\begin{aligned} \frac{d\mathcal{J}}{d\zeta^n} &= \frac{\partial \mathcal{J}}{\partial g_n} \frac{dg_n}{d\zeta^n} + \frac{\partial \mathcal{J}}{\partial \zeta^n} \frac{\partial \zeta^n}{d\zeta^n} = w_n^\top \frac{\partial \mu_{(n-1,n)}}{\partial g_n} + \frac{\partial \mathcal{J}}{\partial \zeta^n} \frac{\partial \zeta^n}{d\zeta^n}, \\ \frac{d\mathcal{J}}{d\zeta^{n-1}} &= \frac{\partial \mathcal{J}}{\partial g_n} \frac{dg_n}{d\zeta^{n-1}} + \frac{\partial \mathcal{J}}{\partial g_{n-1}} \frac{dg_{n-1}}{d\zeta^{n-1}} + \frac{\partial \mathcal{J}}{\partial \zeta^{n-1}} \frac{\partial \zeta^{n-1}}{d\zeta^{n-1}} \\ &= w_n^\top \frac{\partial \mu_{(n-1,n)}}{\partial \zeta^{n-1}} + \left( w_n^\top \frac{\partial \mu_{(n-1,n)}}{\partial g_{n-1}} + \frac{\partial \mathcal{J}}{\partial g_{n-1}} \right) \frac{dg_{n-1}}{d\zeta^{n-1}} \\ &\quad + \frac{\partial \mathcal{J}}{\partial \zeta^{n-1}} \frac{\partial \zeta^{n-1}}{d\zeta^{n-1}}, \\ &= w_n^\top \frac{\partial \mu_{(n-1,n)}}{\partial \zeta^{n-1}} + w_{n-1}^\top \frac{\partial \mu_{(n-2,n-1)}}{\partial \zeta^{n-1}} + \frac{\partial \mathcal{J}}{\partial \zeta^{n-1}} \frac{\partial \zeta^{n-1}}{d\zeta^{n-1}}, \\ \frac{d\mathcal{J}}{d\zeta^0} &= \left( w_2^\top \frac{\partial \mu_{(1,2)}}{\partial g_1} + \frac{\partial \mathcal{J}}{\partial g_1} \right) \frac{dg_1}{d\zeta^0} + \frac{\partial \mathcal{J}}{\partial \zeta^0} \frac{\partial \zeta^0}{d\zeta^0} \\ &= w_1^\top \frac{\partial \mu_{(0,1)}}{\partial \zeta^0} + \frac{\partial \mathcal{J}}{\partial \zeta^0} \frac{\partial \zeta^0}{d\zeta^0}, \end{aligned}$$

where the last identity follows from the same steps as the general case and the fact that  $\zeta^0$  is fixed.  $\square$