

Realizing Robotic Swimming with Unified Fluid-Robot Multiphysics

Jeong Hun Lee^{1*}, Junzhe Hu^{1*}, Sofia Kwok¹, Carmel Majidi¹ and Zachary Manchester^{1,2}

¹Carnegie Mellon University, ²Massachusetts Institute of Technology, *Equal Contribution

Project Page: <https://unified-fluid-robot-multiphysics.github.io/>

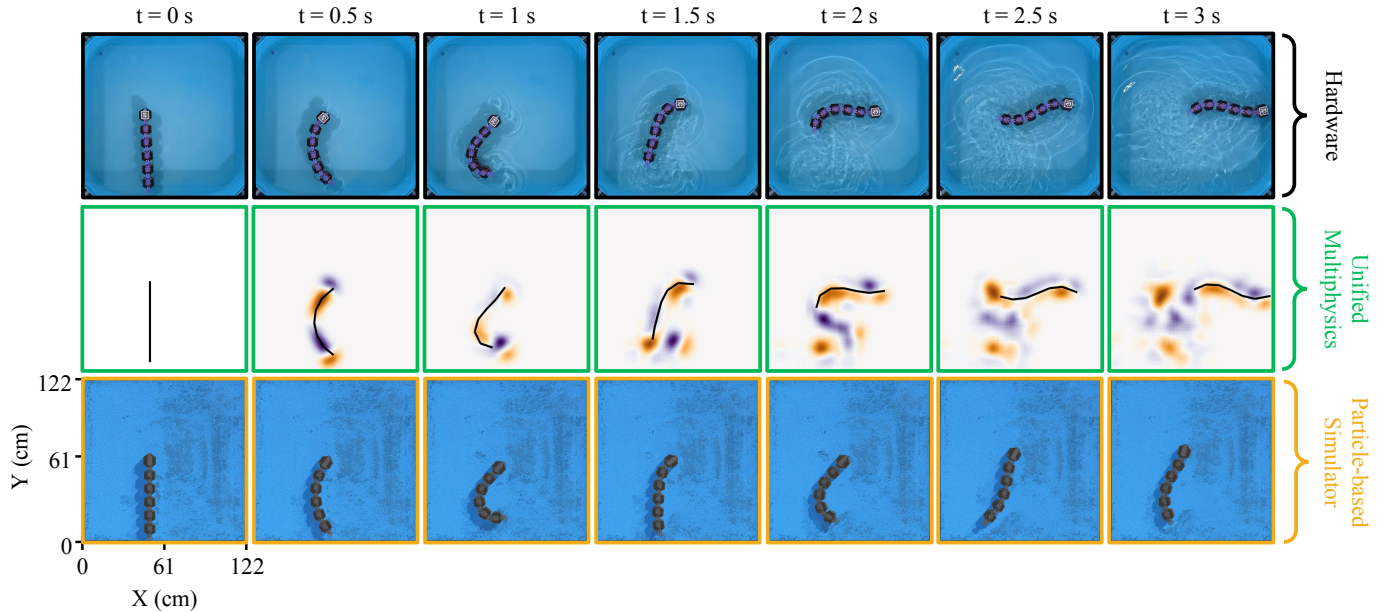


Fig. 1: A highly dynamic fish-inspired C-start escape maneuver executed by a multilink eel-robot. The top row (black) shows execution of the maneuver on the robot, while the middle row (green) is our simulation method and the bottom row (orange) is a baseline implemented in Genesis [17], an open-source robotics simulator. The C-start gait was optimized using the gradients from our differentiable simulator and was successfully executed on real-world hardware. Meanwhile, the smoothed-particle hydrodynamics baseline from Genesis fails to capture the correct dynamics—rolling out the same control trajectory achieves little lateral movement and does not complete the intended 90° turn.

Abstract—Matching the swimming efficiency and agility of fish has remained an elusive goal in underwater robotics. Such locomotion capabilities rely on complex vortex interactions between the robot’s body and the surrounding fluid. However, simulating these dynamics, which are governed by coupled ordinary and partial differential equations, is significantly more difficult than the multi-body dynamics of classical rigid robotic systems. We present a differentiable framework for simulating strongly coupled fluid-robot multiphysics as a unified optimization problem. The coupled manipulator and incompressible Navier-Stokes equations are derived together from a single Lagrangian using the principle of least action. We employ discrete variational mechanics to derive a stable, well-conditioned, and physically accurate scheme for jointly simulating articulated bodies and the surrounding fluid. We leverage the implicit function theorem to compute derivatives of the fully coupled dynamics. Using this simulator and its gradients, we realize undulating swimming gaits and optimize a highly dynamic C-start escape maneuver for a bioinspired eel robot. We validate both gaits on physical hardware, demonstrating successful sim-to-real transfer. Simulation code, hardware data, and schematics for the eel robot can be found here: <https://unified-fluid-robot-multiphysics.github.io/>

I. INTRODUCTION

In recent years, there has been considerable interest in designing robot control policies in multiphysics settings such as deformable-object manipulation [57, 68, 69, 72], fluid transport [26, 32], and locomotion [10, 49, 71]. However, real-world data collection and training in such complex environments is challenging. As a result, multiphysics simulation platforms [9, 17, 38] have seen growing interest and ongoing development, with many robotics applications still lacking viable physics engines.

Bio-inspired locomotion for underwater vehicles [15, 29, 36, 40, 43, 48] has garnered significant interest as a means to improve efficiency and maneuverability via intricate shedding of vortices [7, 61, 62]. Although various bio-inspired hardware designs have been introduced [2, 3, 8, 29], relatively little work has been done to make these vehicles operate autonomously with the same performance as their biological counterparts [3, 66]. We attribute this to the difficulty of

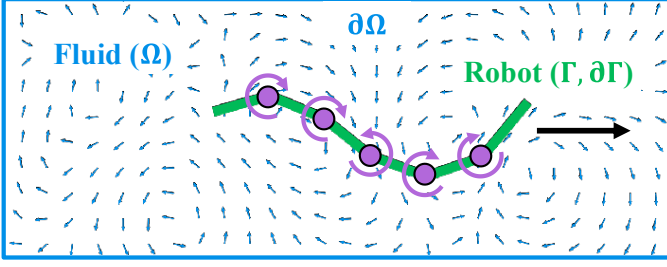


Fig. 2: An overview of the multiphysics problem we address, in which a controlled, rigid-body robotic system must interact with the surrounding incompressible Newtonian fluid (e.g., water) to swim. We denote the fluid domain as Ω and its boundary $\partial\Omega$. Γ and $\partial\Gamma$ correspond to the robot geometry and its boundary.

modeling and simulating such systems, where both whole-body robot dynamics and complex fluid-structure interaction (FSI) are critical (Fig. 2).

We propose a unified framework for jointly deriving and simulating differentiable fluid-robot multiphysics to realize robotic swimming. Specifically, we formulate the combined Lagrangian using the principle of least action to model the multiphysics as a single continuous-time optimization problem, from which we derive the coupled incompressible Navier-Stokes and manipulator equations (Fig. 3). We show that the multiphysics coupling stems from a constraint that enforces the no-slip boundary condition at the fluid-robot interface. We then employ discrete variational mechanics [41] to discretize the unified action directly, which results in an implicit time-integration scheme. We also extend the immersed boundary method from the computational fluid dynamics (CFD) literature [53, 59] to derive an integral “weak-form” version of the no-slip constraint that is amenable to multibody robotic systems. The implicit function theorem is then used to compute simulation derivatives for downstream learning and optimization tasks. Using our unified multiphysics, we realize bioinspired swimming on an eel robot, which we validate in real-world hardware experiments. We also benchmark against a smoothed-particle hydrodynamics (SPH) environment implemented in Genesis [17], a state-of-the-art robotics multiphysics simulator. We investigate both steady undulatory swimming and the highly dynamic C-start escape maneuver, which we optimize to achieve a 90-degree turn. In summary, our contributions are:

- A unified derivation of strongly coupled fluid-robot multiphysics via the principle of least action.
- A new weak formulation of the no-slip boundary constraint at the fluid-robot interface.
- A differentiable fluid-robot multiphysics simulator.
- Realization of accurate, dynamic robotic swimming on a bioinspired eel robot with validation against real-world hardware.

The remainder of this paper is organized as follows: In Section II, we provide a literature review on bioinspired swimming, fluid-robot interaction simulators, and variational

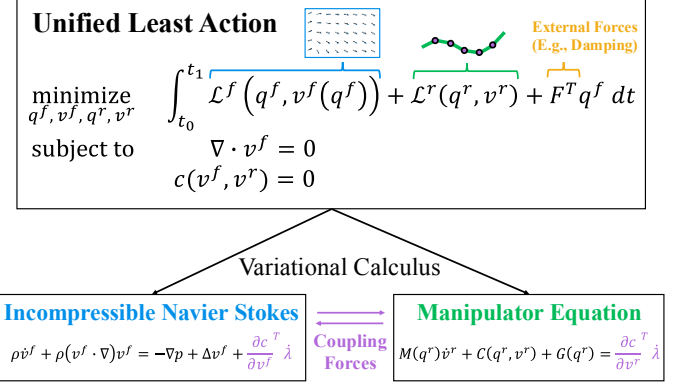


Fig. 3: An overview of our multiphysics model, in which the unified Lagrangian governing the coupled fluid-robot dynamics is formulated using the principle of least action. Rather than the individual differential equations, the combined action is discretized directly to achieve consistent strongly coupled simulation.

mechanics. In Section III, we review background material on the least-action principle, variational integrators, and the immersed-boundary method. Section IV then describes our approach for posing and simulating strongly coupled fluid-robot multiphysics. In Section V, we provide simulation and hardware swimming results for an eel-like swimming robot. Finally, Section VII summarizes our conclusions and discusses limitations and directions for future work.

II. RELATED WORKS

A. Bioinspired Robotic Swimming

For decades, biological swimmers have captivated researchers with their remarkable swimming efficiency [14, 18, 35] and high maneuverability [11, 65]. While early work focused on biomechanics [18, 35, 65], later studies attributed this performance to the detection and exploitation of vortices [61, 62]. This has driven considerable interest in the robotics community toward developing bioinspired hardware for empirical studies, which include robotic fish [3, 29, 66], oscillating foils [2], and lateral-line pressure-sensor arrays [63, 70]. However, relatively little work has enabled these swimmers to operate with the robustness achieved by legged robots [30, 42, 73]—a gap we attribute to the lack of simulators capable of handling both complex fluid physics and rigid-body robot dynamics [4, 17, 37].

B. Fluid Simulation for Robotics

Multiphysics simulators for robotics [17, 38] have garnered recent interest for simulating diverse scenes beyond classical rigid-body dynamics, such as rigid-soft-body interactions for deformable-object manipulation [54] and liquid pouring [67]. These simulators generally employ methods originating from the graphics and particle-based CFD community [25, 39], leveraging their parallelizability and visual-rendering capabilities. However, due to their Lagrangian (i.e., particle-based) representations, these simulators are tailored towards fluid-transport tasks (e.g., liquid pouring) encountered in robotic

manipulation. Compared to their Eulerian (grid-based) counterparts, these fluid simulators are currently unable to simulate complex boundary conditions (e.g., free stream) and become intractable when the fluid must be modeled as a continuum. In addition, SPH-based methods can struggle with maintaining certain constraints such as incompressibility and conservation of volume, which are better posed in Eulerian-based methods [21, 45, 58].

C. Fluid-Structure Interaction (FSI)

Modeling fluid-structure interaction has long been of interest in the CFD community [33, 37, 46, 51], where methods are typically tailored to single-body systems. Recently, Angelidis et al. [4] proposed an SPH-based platform for bioinspired, multibody robotic swimmers. However, the authors note that smoothed-particle hydrodynamics introduces additional damping and parameter sensitivity, causing unstable simulation and sim-to-real gap. Meanwhile, Eulerian methods capable of handling articulated multibody systems commonly found in robotics have yet to be fully realized [60]. Previous works have simplified the fluid dynamics to potential [28] or Stokes flow [12, 19, 20] for low-Reynolds-number regimes. Nava et al. proposed a physics-informed, neural-network model of FSI for the optimization of soft robotic swimmers [46]; however, accuracy is generally poor when generalizing to new shapes and flow conditions. The industry-standard COMSOL multiphysics solver [9] provides strongly-coupled, Eulerian-based FSI via the Arbitrary Lagrangian-Eulerian method, but requires computationally expensive remeshing that struggles with large displacements [44]. As a result, Lee et al. [33] proposed Aquarium, a differentiable incompressible-Navier-Stokes-based simulator for robotics. However, Aquarium currently does not jointly solve the fluid and robot dynamics and relies on a separate dynamics solver to obtain robot states (i.e., "weakly" coupled), which can result in unstable simulation [23]. It additionally relies on the classical immersed-boundary method to represent robot geometry, which is tailored toward single-body systems and suffers from singularities when bodies move close together. Therefore, we develop a unified multiphysics formulation to achieve tightly coupled, generalizable simulation of multibody robots in higher-Reynolds-number fluid environments.

D. Variational Mechanics for Simulation

Over the past two decades, variational integrators have been developed for accurate simulations of various physical systems [24, 34, 41]. Specifically, variational integrators discretize the least-action principle from which the governing equations of motion are derived. The result is stable simulation with good constraint satisfaction and momentum and energy-conservation properties. Although originally for conservative systems, variational integrators have been extended to damped systems and continuum mechanics [34], with connections made to widely used implicit Runge-Kutta schemes [41]. Recent efforts have extended variational integrators to robotics

in the context of multibody simulation with rigid-body contact seen in legged locomotion and manipulation [24]. However, variational integrators remain an unconventional method for physics simulation, which we partly attribute to their configuration-only representation. Therefore, we extend the variational integrator to a fluid-robot multiphysics setting with full robot-state representations found in standard robotics simulators [60].

In the context of fluid dynamics, variational principles have long been studied [5, 56] to pose fluid physics as continuous-time optimization problems. This optimization-based treatment has inspired several practices in CFD, including the treatment of boundary conditions as constraints with associated dual variables [1, 52, 59]. We aim to extend these variational principles to the discrete setting with robot dynamics in mind, resulting in naturally coupled multiphysics simulation.

III. BACKGROUND

This section provides a brief overview of variational calculus, variational integrators, and the immersed-boundary method for fluid-structure interaction. We refer the reader to the comprehensive, existing literature for more details [5, 41, 56, 64].

A. Physics as Optimization

Various differential equations for modeling physics can be derived from the principle of least action via variational calculus. Specifically, the principle of least action represents the dynamics as a continuous-time optimization problem:

$$\begin{aligned} & \underset{q, v}{\text{minimize}} && \int_{t_0}^{t_f} T(v(t)) - U(q(t)) dt \\ & \text{subject to} && v = \dot{q}, \end{aligned} \quad (1)$$

where $q(t) \in \mathbb{R}^n$ is the configuration of the system (e.g., fluid-particle position and robot pose) as a function of time, t ; $v(t) \in \mathbb{R}^m$ is the velocity; $T(v) \in \mathbb{R}^+$ is the kinetic energy of the system; and $U(q) \in \mathbb{R}^+$ is the potential energy; and $v = \dot{q}$ represents the kinematic constraint. We denote $L(q, v) = T(v) - U(q)$ as the *physics* Lagrangian, which we can use to derive an expression for the action,

$$S(q, v) = \int_{t_0}^{t_f} L(q, v) + \lambda^T(\dot{q} - v) dt, \quad (2)$$

where λ is the dual variable. The first-order necessary (FON) conditions of the optimization problem (5) form the Euler-Lagrange equation that governs the system dynamics:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = 0, \quad (3)$$

which is commonly rewritten as the manipulator equation in robotics:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = 0. \quad (4)$$

We note that the least-action principle can be extended to non-conservative systems via the Lagrange-D'Alembert principle of virtual work [34]; to more complex kinematics $v = f(q, \dot{q})$ as encountered e.g. with different parametrizations

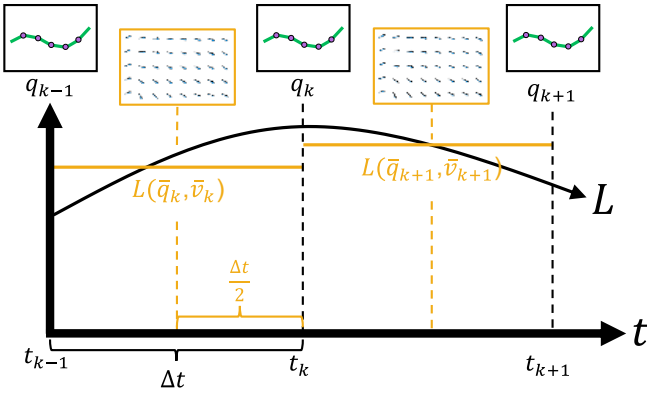


Fig. 4: A midpoint variational integrator, where a quadrature rule is traditionally used to discretize the time integral of Fig. 3 to simulate robot configurations, q_k . However, the fluid is represented as a velocity field, $\bar{v}_k$ that is approximated at the midpoint, causing a temporal mismatch. Therefore, a new variational integrator is formulated for the fluid-robot multiphysics.

of attitude [27]; and to include constraints such as joints in manipulator arms [6]:

$$\begin{aligned} & \underset{q, v}{\text{minimize}} && \int_{t_0}^{t_f} T(q, v) - U(q) + F(t)^T q(t) dt \\ & \text{subject to} && v = f(q, \dot{q}), \\ & && c(q, v) = 0, \end{aligned} \quad (5)$$

where $F(t) \in \mathbb{R}^n$ represents external forces such as damping and motor torques.

B. Variational Integrators

Robotics simulators commonly integrate (4) via a Runge-Kutta (RK) scheme [60]. However, most RK schemes are known to exhibit unstable behavior or introduce artificial damping. Meanwhile, variational integrators provide implicit time-integration schemes that are known to conserve energy and momentum, making them attractive for robotics simulators.

First, the continuous-time action integral is discretized via numerical quadrature. As a particular example (Fig. 4), a midpoint quadrature rule results in the following optimization problem:

$$\begin{aligned} & \underset{q_k}{\text{minimize}} && \sum_{k=1}^2 L_d(q_k, q_{k+1}) + \frac{h}{2} F(t_k + \frac{h}{2})^T (q_k + q_{k+1}) \\ & \text{subject to} && c(q_k) = 0, \end{aligned} \quad (6)$$

where q_k is the configuration at time t_k ; $h \in \mathbb{R}$ is the time step; and $L_d(q_k, q_{k+1}) \in \mathbb{R}^+$ is the discrete Lagrangian, which is expressed as:

$$L_d(q_k, q_{k+1}) = hL\left(\underbrace{\frac{q_k + q_{k+1}}{2}}_{q_{k+1}}, \underbrace{\frac{q_{k+1} - q_k}{h}}_{\bar{v}_{k+1}}\right), \quad (7)$$

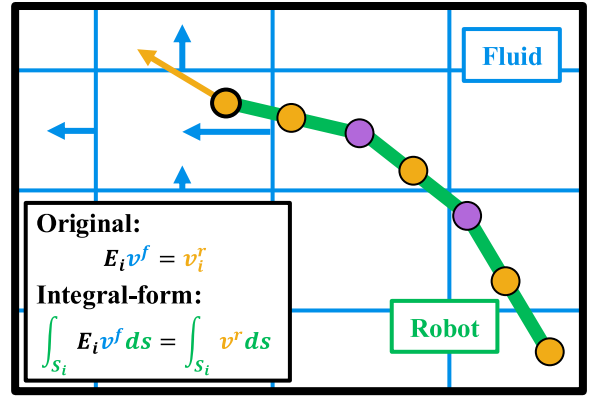


Fig. 5: The immersed boundary method, where the fluid domain and robot geometry are discretized separately into a grid and boundary-mesh. Originally, the fluid-robot coupling is realized by a convolution matrix E that maps fluid-cell velocities (blue) to those at the robot-boundary nodes (orange). This can lead to fluid-penetration and singularities at duplicate nodes (purple) that correspond to multiple bodies. Therefore, we introduce an extension that integrates the immersed boundary method across the boundary (green), improving both physical accuracy and compatibility with multibody representations, which are common in robotics.

where $\bar{q}_{k+1}$ and $\bar{v}_{k+1}$ are the configurations and velocities defined at the midpoint between t_k and t_{k+1} .

We then obtain the discrete Euler-Lagrange equation, which is the FON condition of (6) w.r.t. q_k :

$$D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1}) + h \bar{F}_k + h \underbrace{\lambda_k^T D_1 c(q_k)}_{\text{constraint force}} = 0, \quad (8)$$

$$c(q_{k+1}) = 0, \quad (9)$$

where $\lambda_k \in \mathbb{R}^z$ is the dual variable. Upon inspection, λ_k provides the impulse needed to enforce the constraint $c(q_{k+1}) = 0$. D_k denotes partial differentiation with respect to a function's k th argument, and $\bar{F}_k$ represents the averaged external force $F(t)$ at the midpoints:

$$\bar{F}_k = \frac{1}{2} (F(t_k - \frac{h}{2}) + F(t_k + \frac{h}{2})). \quad (10)$$

By initializing q_k and q_{k-1} , (8)-(9) can be solved to compute q_{k+1} . However, this integration method only integrates over configurations, with a half-time-step delay present between q_k and $\bar{v}_k$. This introduces a temporal inconsistency due to the fluid being modeled as a velocity field, $\bar{v}_k(q_k)$, while the robot's pose is defined by q_k . Marsden and West [41] previously connected this variational integrator to the implicit midpoint RK scheme over the full state, $x = [q; v]$, via the discrete Legendre transform. However, doing so with constraints can be difficult and unintuitive. Therefore, we aim to provide a simpler approach for performing variational integration over the full variational state, $x = [q; \bar{v}]$.

C. The Immersed Boundary Method

To account for the representation discrepancy between the Eulerian fluid-velocity grid and Lagrangian (free-floating) robot geometry, Peskin [53] formulated the immersed boundary method to model the no-slip boundary condition at the robot-fluid interface as shown in Fig. 5. Specifically, the fluid velocity is set equal to the robot velocity at its boundary, which is expressed as the following integration over the fluid-velocity field:

$$\int_{\Omega} v^f \delta(q^f - {}^b q^r) d\Omega = {}^b v^r({}^b q^r) \quad \forall {}^b q^r \in \partial\Gamma, \quad (11)$$

where Ω is the fluid domain; $\partial\Gamma$ is the robot geometry's boundary; v^f is the fluid velocity at position q^f ; q^r, v^r are the solid body (i.e., robot) configuration and velocity; and ${}^b v^r$ is the velocity at a point ${}^b q^r$ along $\partial\Gamma$. $\delta(\cdot)$ is the dirac delta function. In addition, the corresponding forcing term acting on the fluid is explicitly defined as an inverse mapping:

$$F^f = \int_{\Gamma} F^r \delta(q^f - {}^b q^r) d\Gamma \quad \forall q^f, \quad (12)$$

In discrete form, the fluid domain and robot geometry are discretized into a separate grid and mesh respectively, allowing for decoupled movement of the robot. Meanwhile, the dirac delta is approximated by a smoothed version. This results in the discretized immersed boundary method:

$$c(v^f, v^r) = E v_d^f - {}^b v_d^r(v^r) = 0, \quad (13)$$

$$F_d^f = E^T F_d^r, \quad (14)$$

where E is a convolution matrix that maps discrete fluid velocities on the grid to those of the robot-boundary nodes. However, upon inspection, this discretized constraint is only satisfied at finite nodes of the boundary mesh, allowing fluid to penetrate the boundary if the nodes are far apart. On the other hand, overlapping points can cause numerical ill-conditioning due to redundant no-slip constraints, which can occur in multibody robots as shown in Fig. 5. Therefore, we extend the immersed boundary method to derive an integral-form that will properly enforce no-slip in between nodes while being unaffected by overlapping ones.

IV. METHODOLOGY

A. Fluid-Robot Multiphysics as Optimization

As previously mentioned, existing simulators [17, 38] typically integrate over the individual differential equations before coupling the physics via a force term such as (12). We propose the least-action principle as an alternative, unified perspective of multiphysics from which the differential equations are derived.

We begin by proposing the least-action principle for the combined multiphysics problem between a rigid multibody

robot and an incompressible, Newtonian fluid (e.g., water):

$$\begin{aligned} & \underset{q_k^r, q^f, v^r, v^f}{\text{minimize}} && \int_{t_0}^{t_f} L^f(q^f, v^f) + L^r(q^r, v^r) + F(t)^T q^f dt \\ & \text{subject to} && c_1(v^f) = \dot{q}^f - v^f(q^f) = 0 \quad \forall q^f \in \Omega, \\ & && c_2(v^f) = \nabla \cdot v^f = 0 \quad \forall q^f \in \Omega, \\ & && c_3(v^f) = v^f - v_{bc} = 0 \quad \forall q^f \in \partial\Omega, \\ & && c_4(v^r) = \dot{q}^r - v^r = 0 \quad \forall q^r, \\ & && c_5(q^r) = 0 \quad \forall q^r, \\ & && c_6(v^f, v^r) = v^f - {}^b v^r = 0, \quad \forall q^f \in \Gamma, \end{aligned} \quad (15)$$

where Ω, Γ represent the fluid domain and robot geometry respectively with corresponding boundaries, $\partial\Omega, \partial\Gamma$; q^r, q^f are the configurations of the robot and fluid particle respectively and v^r, v^f are the corresponding velocities. $c_1(v^f)$ encodes the kinematics constraint for the fluid-velocity field (i.e., an array of velocity sensors). $c_2(v^f)$ encodes the conservation-of-mass constraint while $c_3(v^f)$ enforces boundary conditions along the fluid-domain boundary (e.g., walls, free-stream velocity, etc.). $c_4(q^r)$ encodes the kinematics constraint of the robot while c_5 enforces other robot-configuration constraints, such as joint constraints found in manipulator arms. $c_6(v^f, v^r)$ encodes the no-slip constraint, which couples the robot and fluid physics.

We first define the fluid and (single) rigid-body Lagrangians:

$$L^f(q^f, v^f) = \int_{\Omega} \frac{1}{2} \rho (v^f)^T v^f - \rho g^T q^f dV, \quad (16)$$

$$L^r(q^r, v^r) = \frac{1}{2} (v^r)^T M^r v^r - g^T M^r q^r, \quad (17)$$

where ρ is the fluid density; M^r is the mass matrix of the robot body; dV denotes the volume differential; and g is gravity. In this problem, damping is introduced by the fluid viscosity:

$$F(t) = \int_{\Omega} \mu \Delta v^f dV, \quad (18)$$

where μ is the dynamic viscosity of the fluid. In a manner similar to (2), we write down the action for the combine problem:

$$S(q^r, q^f, v^f, v^r) = \int_{t_0}^{t_f} L^f + L^r + F^T q^f + \sum_{i=1}^6 \lambda_i c_i dt. \quad (19)$$

The FON conditions of (38) can be expressed by the variations w.r.t. q^r, v^r and q^f, v^f , which results in the coupled incompressible Navier-Stokes and manipulator equations, respectively:

$$\begin{aligned} \rho(\dot{v}^f + (v^f \cdot \nabla)v^f) &= -\nabla p + \mu \nabla^2 v^f - \rho g \\ &\quad - \frac{\partial c_3}{\partial v^f} \dot{\lambda}_3 - \frac{\partial c_6}{\partial v^f} \dot{\lambda}_6, \end{aligned} \quad (20)$$

$$\nabla \cdot v^f = 0, \quad (21)$$

$$\overbrace{M^r \dot{v}^r + M^r g}^{(4)} = -\frac{\partial c_5}{\partial q^r} \dot{\lambda}_5 - \frac{\partial c_6}{\partial v^r} \dot{\lambda}_6. \quad (22)$$

$$\dot{q}^r = v^r, \quad (23)$$

where $p = \dot{\lambda}_2 \in R^m$ is the fluidic pressure, which Lagrange [31] originally realized as the dual variable that enforces the

conservation of mass. (20) also provides the same constraint-based treatment of fluid-boundary conditions of Allmaras [1], in which coupling forces arise from the no-slip constraint at the fluid-robot interface.

B. Coupled Discretization of the Fluid-Robot Multiphysics

Using the unified least-action principle of (38), we develop a variational integrator to simulate fluid-robot multiphysics. However, we again note that q^r and v^f have a half-time-step lag in traditional variational integrators. Therefore, we aim to derive a new variational integrator that can integrate q_k and v_k while handling this temporal inconsistency.

First, we discretize the fluid domain spatially using a second-order finite-volume method [13], which provides a discrete counterpart to the following continuous operators:

$$\begin{aligned} \int_{\Omega} \rho v^f dV &\Rightarrow M^f v^f, & \int_{\Omega} \nabla p dV &\Rightarrow Gp, \\ \int_{\Omega} (v^f \cdot \nabla) v^f dV &\Rightarrow N(v^f), & \int_{\Omega} \Delta v^f dV &\Rightarrow Lv^f. \end{aligned}$$

Through a slight abuse of notation, we will use q^f , v^f , and p to refer to their discrete versions throughout the rest of this section. We may now express the spatially discrete fluid Lagrangian as:

$$L^f(q^f, v^f) = \frac{1}{2}(v^f)^T M^f v^f - M^f g^T q^f dV, \quad (24)$$

To handle temporal lag, we will define the constraints corresponding to q_k and $\bar{v}_k$ at their respective time steps, resulting in the following discretized action:

$$\begin{aligned} \text{minimize}_{q_k^r, q_k^f} & \sum_{k=1}^2 L_d^f(q_k^f, q_{k+1}^f) + L_d^r(q_k^r, q_{k+1}^r) + hF(t_k + \frac{h}{2})^T \bar{q}_{k+1}^f \\ \text{subject to} & \quad c_2(\bar{v}_k^f) = G^T \bar{v}_k^f = 0, \\ & \quad c_3(\bar{v}_k^f) = B \bar{v}_k^f - v_{bc} = 0, \\ & \quad c_5(q_k^r) = 0, \\ & \quad c_6(\bar{v}_k^f, \bar{v}_k^r) = E \bar{v}_k^f - {}^b \bar{v}^r(\bar{v}_k^r) = 0, \end{aligned} \quad (25)$$

where $B \bar{v}_k^f$ extracts fluid velocities located at the fluid-domain boundary, and $c_6(\bar{v}_k^f, \bar{v}_k^r)$ is expressed using the discretized immersed boundary method posed in (13). The external force is additionally modified to account for the convective term, $N(v^f)$:

$$F(t_k + \frac{h}{2}) = L \bar{v}_{k+1}^f + N(\bar{v}_{k+1}^f). \quad (26)$$

Upon inspection, we note that the derivatives of (8) can be expressed in terms of $\bar{q}^f$ and $\bar{v}^r$ using the chain rule:

$$D_2 L_d(q_{k-1}, q_k) = h \left(\underbrace{\frac{\partial L}{\partial \bar{q}_k} \frac{\partial \bar{q}_k}{\partial q_k}}_{\frac{1}{2}} + \underbrace{\frac{\partial L}{\partial \bar{v}_k} \frac{\partial \bar{v}_k}{\partial q_k}}_{\frac{1}{h}} \right), \quad (27)$$

$$D_1 L_d(q_k, q_{k+1}) = h \left(\underbrace{\frac{\partial L}{\partial \bar{q}_{k+1}} \frac{\partial \bar{q}_{k+1}}{\partial q_k}}_{\frac{1}{2}} + \underbrace{\frac{\partial L}{\partial \bar{v}_{k+1}} \frac{\partial \bar{v}_{k+1}}{\partial q_k}}_{-\frac{1}{h}} \right). \quad (28)$$

We can now substitute (27)-(28) into (8). This results in the following FON conditions for (25) corresponding to the robot

dynamics:

$$q_{k+1}^r = q_k^r + h \bar{v}_{k+1}^r, \quad (29)$$

$$\begin{aligned} \frac{1}{2} M^r \bar{v}_{k+1}^r - h \left(\frac{\partial c_5}{\partial q_k^r} \right)^T \lambda_{5,k} \\ - h \left(\frac{\partial c_6}{\partial \bar{v}_k^r} \right)^T \lambda_{6,k} = \frac{1}{2} M^r \bar{v}_k^r - h M^r g, \end{aligned} \quad (30)$$

as well as that corresponding to the fluid dynamics:

$$\begin{aligned} \frac{1}{2} (M^f - \mu h L) \bar{v}_{k+1}^f - \frac{h}{2} N(\bar{v}_{k+1}^f) \\ + G p_k + B \lambda_{3,k} - h \left(\frac{\partial c_6}{\partial \bar{v}_k^f} \right)^T \lambda_{6,k} \\ = \frac{1}{2} (M^f + \mu h L) \bar{v}_k^f + \frac{h}{2} N(\bar{v}_k^f) - h M^f g. \end{aligned} \quad (31)$$

Finally, the constraints of (25) form the rest of the FON conditions:

$$G^T \bar{v}_{k+1}^f = 0, \quad (32)$$

$$B \bar{v}_{k+1}^f - v_{bc} = 0, \quad (33)$$

$$E \bar{v}_{k+1}^f - {}^b \bar{v}^r(\bar{v}_{k+1}^r) = 0, \quad (34)$$

$$c_5(q_{k+1}^r) = 0. \quad (35)$$

The system of equations defined by (29)-(35) can be solved using Newton's method. Specifically, $q_{k+1}^r, \bar{v}_{k+1}^r, \bar{v}_{k+1}^f$ and the corresponding dual variables can be solved for given $q_k^r, \bar{v}_k^r, \bar{v}_k^f$. Derivatives can then be computed using the implicit function theorem on the entire multiphysics problem [33]. Upon further inspection, we also note that this integration scheme results in a 2nd-order implicit leap-frog method [22].

C. An Integral-Form Immersed Boundary Method

As previously mentioned, the immersed boundary method can be both ill-posed and ill-conditioned due to enforcing the no-slip constraint at a set of (possibly changing) discrete node locations. Therefore, we propose a new formulation that integrates the original immersed boundary method along the solid-body boundary as shown in Fig. 5:

$$\oint_{\partial\Gamma} \int_{\Omega} v^f \delta(q^f - {}^b q^r(q^r)) dV dS = \oint_{\partial\Gamma} {}^b v^r(q^r, v^r) dS, \quad (36)$$

where $\oint dS$ denotes a surface integral. We discretize (36) with piecewise-linear interpolation along the boundary. The discrete delta function is then integrated along each linear interpolant, resulting in another convolution matrix for the following discretized immersed-boundary method:

$$\bar{E} v^f - {}^b v^r(v^r) = 0, \quad (37)$$

where $\bar{E}$ is the integrated convolution matrix. Rather than explicitly formulating a forcing term such as (12), we simply differentiate through (37) as seen in (30)-(31).

D. Optimization With Fluid-Robot-Multiphysics Gradients

By leveraging its differentiability, we integrate the unified fluid-robot multiphysics into a gradient-based optimization process. We begin with the general problem:

$$\begin{aligned} \text{minimize}_{\beta} & \quad J(x_{1:N}) \\ \text{subject to} & \quad f(x_{k+1}, x_k; \beta) = 0, \end{aligned} \quad (38)$$

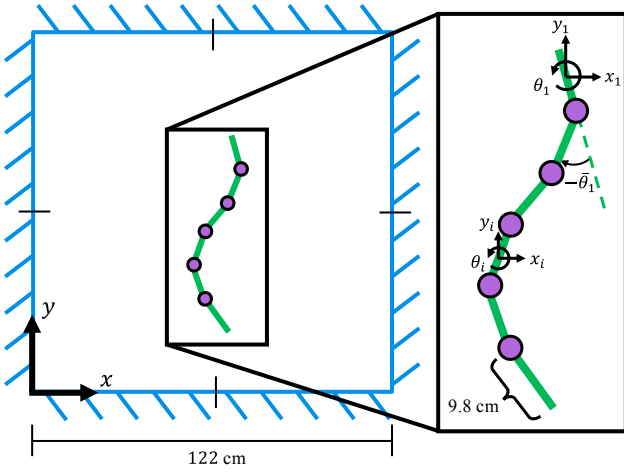


Fig. 6: Our experimental setup, where a bioinspired eel robot is navigating through a walled cavity of initially still water measuring 122×122 cm. In this work, we parameterize the robot configuration in maximum coordinates where all link configurations, $[x_i, y_i, \theta_i]$, are specified. Joint (i.e., motor) angles between links i and $i + 1$ are denoted as $\bar{\theta}_i$.

where β are the optimization parameters; $x_k = [q^r, \bar{v}^r, \bar{v}^f]_k$ is the full fluid-robot state at timestep k ; $J(x_{1:N})$ is the objective; and $f(x_{k+1}, x_k; \beta)$ are the implicit dynamics constraints of our unified fluid-robot multiphysics described by (29)-(35). Gradient-based optimizers such as L-BFGS [47] solve (38) using the objective gradient, $\nabla_{\beta} J$, which can be computed via the chain rule:

$$\nabla_{\beta} J = \left(\frac{\partial J}{\partial x_N} \frac{\partial x_N}{\partial \beta} + \frac{\partial J}{\partial x_N} \frac{\partial x_N}{\partial x_{N-1}} \frac{\partial x_{N-1}}{\partial \beta} + \dots \right)^T, \quad (39)$$

where $\frac{\partial x_k}{\partial \beta}$ and $\frac{\partial x_{k+1}}{\partial x_k}$ are computed by applying the implicit function theorem [33] to $f(x_{k+1}, x_k; \beta)$. By differentiating through the unified multiphysics, the gradients also take into account the sensitivity between the robot and fluid states.

V. EXPERIMENTAL RESULTS

This section presents the results of several validation studies to evaluate our fluid-robot multiphysics approach. Specifically, we look at the locomotion problem of an eel robot swimming through an initially-still fluid environment. We first realize the simulation of a steady forward-swimming gait before optimizing a C-start escape maneuver to achieve robotic turning in a desired heading direction. We compare our simulated trajectories with that of an SPH-based baseline implemented in Genesis [17] and real-world hardware.

A. Experimental Setup

As shown in Fig. 6, we simulate a swimming pool of 122×122 cm by defining a cavity of initially still water with walled boundary conditions. We utilize a 1 cm fluid cell with an equivalently-sized particle for the SPH baseline to maintain consistent fidelity. The eel robot consists of 6 segments, each 9.8 cm in length, approximated as rigid, thin plates connected at their ends via revolute joints. Servo motors drive the joints,

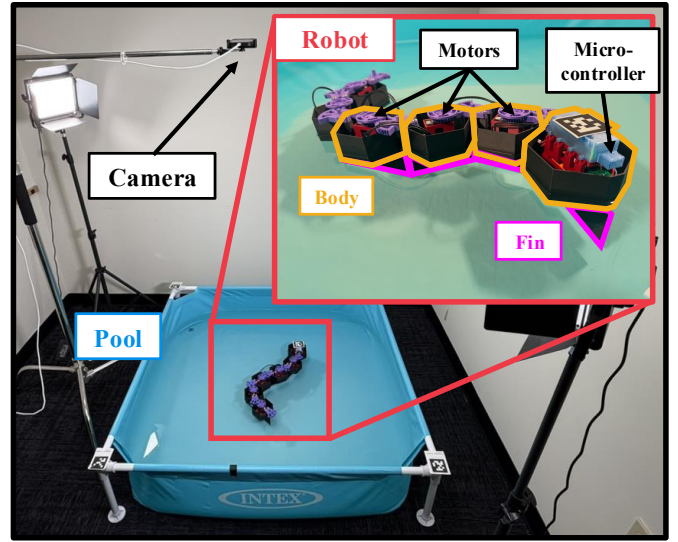


Fig. 7: Experiment setup for sim-to-real validation of bioinspired swimming. The walled cavity is recreated in a pool of equal dimensions, while the eel robot is fully realized on real-world hardware. The robot hardware is fully untethered with an onboard battery, a Robotis OpenRB-150 microcontroller, and Dynamixel XC-330-M288-T servo motors to actuate the joints. 3D-printed hollow bodies keeps the eel robot afloat while thin fins are attached beneath to maintain a thin-plate profile. The head-link configuration is tracked by a top-down-facing camera via AprilTags [50].

whose PD controllers we simulate to apply joint torques. We parameterize the robot configuration in maximal coordinates, which includes all link configurations, $\{[x_i, y_i, \theta_i]\}_{i=1}^6$. We refer to joint angles as $\bar{\theta}_i, i = 1, 2, \dots, 5$.

We recreate the simulated environment on real-world hardware as shown in Fig. 7. The eel-robot hardware is designed to be fully untethered with an onboard 900-mAh battery and a Robotis OpenRB-150 microcontroller on the head (i.e., first) link. Subsequent links consist of Dynamixel XC330-M288-T servo motors to drive the joint angles. The electronics hardware are housed in buoyant, 3D-printed bodies to avoid contact with the fluid. Thin, rigid fins are attached underneath the bodies to replicate the thin-plate profile realized in simulation. A camera is used to track the configuration of the head link using AprilTags [50].

B. Steady Undulatory Swimming

We first perform steady undulation to realize forward swimming of the eel robot. The joint angles are driven by a sinusoidal profile with the same amplitude, each with the appropriate phases to create a traveling wave that undulates at 1 hz with a wavelength equal to the length of the robot body. In this study, we evaluate swimming performance across multiple amplitudes of $10^\circ, 20^\circ, 30^\circ, 40^\circ$. Specifically, we compute the root-mean-square error (RMSE) across a 4-s trajectory of the head link configuration. This was performed over 20 trials for each amplitude to compute the mean RMSE and standard deviations reported in Fig. 8.

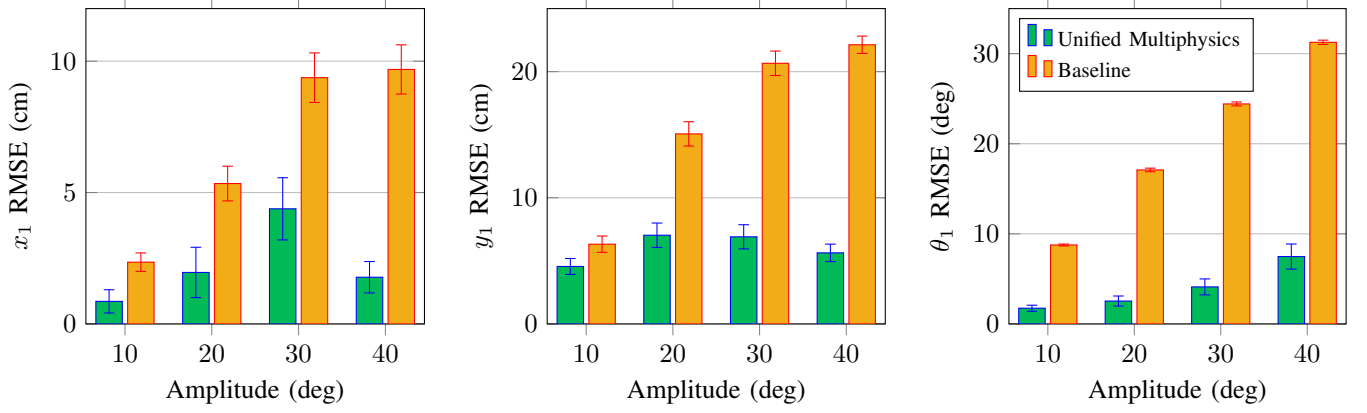


Fig. 8: Average root-mean-square error (RMSE) and associated standard deviations for simulated versus hardware-validated trajectories of an eel robot’s head-link configuration during steady undulatory swimming. This is performed for joint-angle amplitudes of 10° , 20° , 30° , and 40° across 20 trials per amplitude. Our unified fluid-robot multiphysics (green) consistently outperforms the smoothed-particle-hydrodynamics baseline (orange) across all undulation amplitudes, with the improvement becoming more pronounced at higher amplitudes. At 40° amplitude, we achieve error reductions as high as 75% compared to the baseline, demonstrating that strongly coupled fluid-robot simulation significantly improves fidelity for robotic swimming.

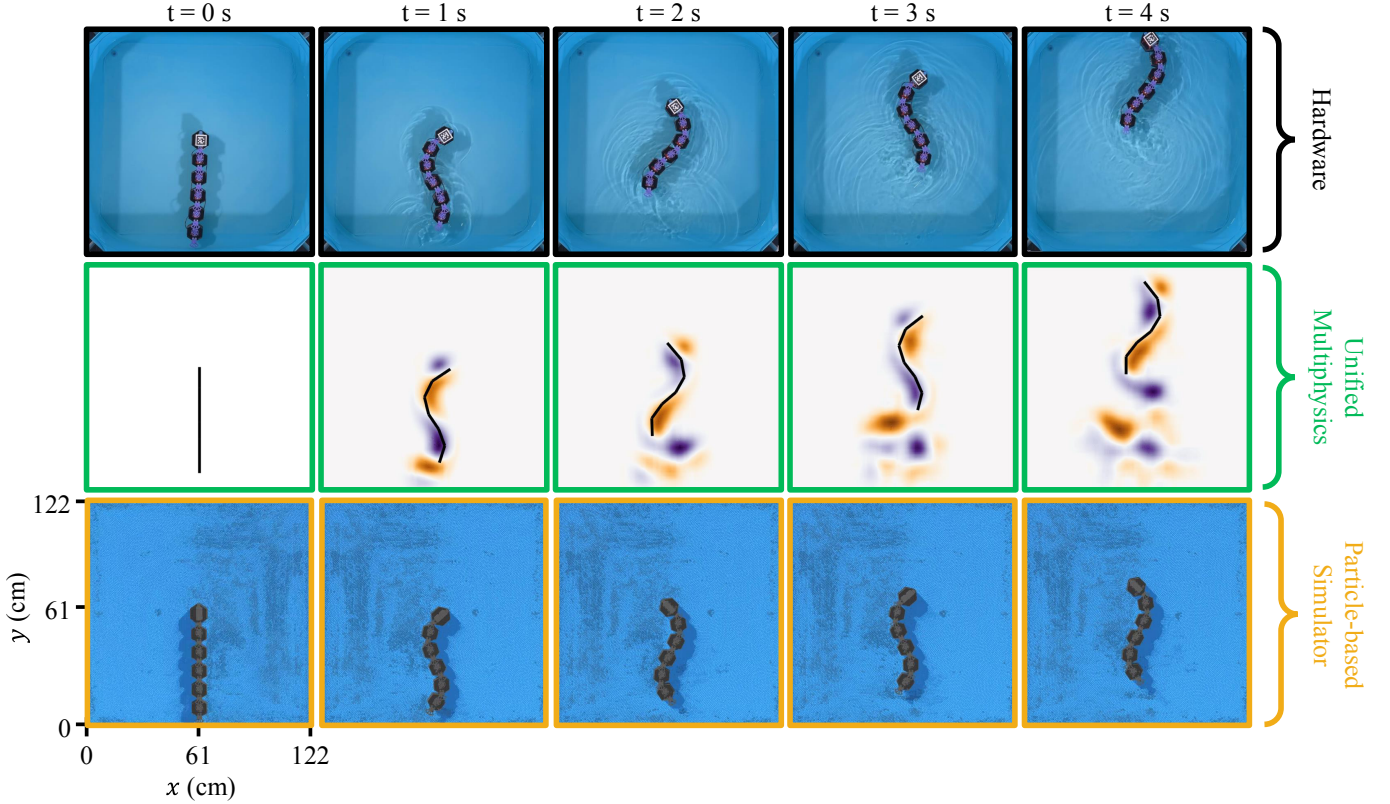


Fig. 9: Time-lapse sequences showing a multilink eel-robot executing an undulating gait with a 40° joint-angle amplitude. The top row (black) demonstrates the real-world robot, while the middle row (green) is our method and the bottom row (orange) is the smoothed-particle-hydrodynamics-based (SPH-based) baseline. While our cell-based method is able to closely follow the forward-swimming seen in the real world, the SPH-based baseline achieves comparatively little forward progress.

As shown in Fig. 8, our unified-multiphysics approach is able to qualitatively realize the forward-swimming motion seen on hardware for a 40° amplitude. Meanwhile the SPH baseline achieves comparatively little forward progress. This appears to be the case for all amplitudes, with our method con-

sistently achieving much lower RMSE error. In the forward-swimming direction (y), the baseline’s RMSE error appears to increase with amplitude while our method maintains a more consistent, slightly decreasing, trend. This results in a nearly 75% improvement for the 40° case and suggests that the

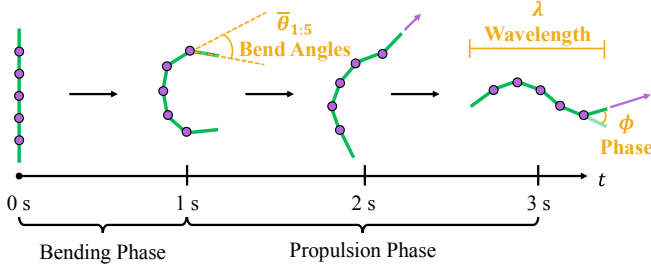


Fig. 10: Kinematic sequence of the eel robot performing a C-start escape maneuver to achieve rapid turning. During the bending phase (0–1 second), the robot curls its body into a characteristic C-shape parameterized by the bending angles at the joints, $\bar{\theta}_{1:5}$. In the propulsion phase (1–3 seconds), the body rapidly straightens and propagates a traveling wave toward the tail, generating a strong fluid interaction to accelerate in a new heading direction (purple arrows). This is parameterized by the traveling wave’s phase, ϕ and wavelength, λ . In this work, we aim to optimize this gait (orange) to perform a 90-degree turn from initial rest.

SPH baseline is unsuitable for more dynamic swimming. We attribute this to the particle-based fluid representation’s poor satisfaction of physical constraints such as incompressibility and volume conservation [45, 58].

C. Optimizing a Highly Dynamic C-Start Maneuver

We leverage our method’s differentiability to optimize a highly dynamic C-start escape maneuver inspired by fish [11, 16, 55]. As shown in Fig. 10, the maneuver involves the robot curling into a C-shape from an initial vertical orientation during a bending phase (0-1 seconds), followed by a powerful transition into undulation during the propulsion phase (1-3 seconds) to achieve fast reorientation and acceleration. We optimize the gait to perform a 90° turn, which is expressed by the following quadratic objective:

$$J(x_N) = \frac{1}{2}(q_N^{\text{com}} - q^{\text{goal}})^T Q(q_N^{\text{com}} - q^{\text{goal}}), \quad (40)$$

where q_N^{com} is the robot’s final-time center-of-mass configuration; q^{goal} is the goal configuration representing a successful 90° turn; and Q is the cost matrix. We optimize over the joint bending angles, $\bar{\theta}_{1:5}$ during the bend phase, the initial phase of $\bar{\theta}_1$, ϕ , and the wavelength of the undulating wave, λ . We fix the amplitude per joint to 25° to prevent the robot from hitting the wall. The optimization is performed using the gradient-based box-constrained L-BFGS algorithm [47] with bounds of $10^\circ \leq \bar{\theta}_{1:5} \leq 36^\circ$, $-\frac{\pi}{2} \leq \phi \leq 0$ rad, and $0.5L^B \leq \lambda \leq L^B$, where L^B is the body length of the robot. We begin the optimization with an initial guess of $\bar{\theta}_{1:5}^{\text{ig}} = 10^\circ$, $\phi^{\text{ig}} = -\frac{\pi}{3}$ rad, and $\lambda^{\text{ig}} = 0.6L^B$. The optimized values are $\bar{\theta}_{1:5}^* = [17^\circ, 18^\circ, 20^\circ, 24^\circ, 36^\circ]$, $\phi^* = -1.16$ rad, and $\lambda^* = 0.57L^B$. The resulting gaits are rolled out in an open-loop manner, and the corresponding time-lapse sequences are visualized in Fig. 1.

As shown in Fig. 1, our optimization process is able to calculate gait parameters that successfully realize the desired 90°

turn. We do note that our method consistently undershoots the travel distance observed in the real world, which is likely due to unmodeled free-surface dynamics, servo compliance, thin-plate geometry approximations, and uncertainty in physical parameters such as robot mass properties and fluid viscosity. Despite this sim-to-real gap, the 90° turn is still fully realized on hardware. Meanwhile, the SPH baseline fails to realize both the appropriate turning and propulsion, further suggesting its unsuitability for highly dynamic robot-swimming tasks.

D. Computational Cost

Regarding computational cost, a 3-second C-start simulation at a 100 Hz sample rate in our unified multiphysics framework requires approximately 261 seconds (0.87 seconds per timestep) on the CPU of an M3-Ultra Mac Studio (28 cores). The same rollout in Genesis [17] requires approximately 2727 seconds (9.09 seconds per timestep) on the M3-Ultra CPU and 300 seconds (1 seconds per timestep) on an NVIDIA RTX 3070-Ti GPU.

VI. LIMITATIONS

Our unified framework for modeling strongly coupled multiphysics has several limitations: First, the computational complexity of performing implicit time integration is high, especially when extended to 3D. Future work is needed to develop efficient sparsity-exploiting solvers to scale our method to large-scale problems. Additionally, while the Eulerian fluid representation is well suited for dynamic tasks like swimming, it is less convenient for visual rendering than particle-based methods.

VII. CONCLUSIONS

We have presented a unified framework for deriving and simulating fluid-robot multiphysics as a single optimization problem. Specifically, we model the unified least-action principle from which the coupled differential equations are derived. We build upon previous works in variational mechanics to discretize the action directly to simulate the fluid-robot interaction in a stable, tightly coupled manner. We additionally modify the original immersed-boundary method to handle articulated multibody systems found in robotics. We validate our approach on a bioinspired eel robot, where we realize both undulatory swimming and fast C-start escape maneuvers via gradient-based optimization. Compared to a weakly coupled, particle-based baseline, the resulting swimming behaviors align much more closely with real-world experiments, showcasing the importance of representation and coupling for improved sim-to-real transfer in fluid-robot multiphysics settings.

In future work, we aim to address the computational scalability of our variational method for downstream design and trajectory-optimization tasks. We also wish to explore the connections to other CFD methods like implicit large-eddy simulations. Finally, we hope to extend our framework to other multiphysics problems such as rigid-soft robots that experience contact.

REFERENCES

- [1] Steven Allmaras. Lagrange multiplier implementation of Dirichlet boundary conditions in compressible Navier-Stokes finite element methods. In *17th AIAA Computational Fluid Dynamics Conference*, page 4714, 2005.
- [2] J. M. Anderson, K. Streitlien, D. S. Barrett, and M. S. Triantafyllou. Oscillating foils of high propulsive efficiency. *Journal of Fluid Mechanics*, 360:41–72, 1998. doi: 10.1017/S0022112097008392.
- [3] Jamie M. Anderson and Narendra K. Chhabra. Maneuvering and stability performance of a robotic tuna. *Integrative and comparative biology*, 42(1):118–126, February 2002. ISSN 1540-7063. doi: 10.1093/icb/42.1.118.
- [4] Emmanouil Angelidis, Jonathan Arreguit, Jan Bender, Patrick Berggold, Ziyuan Liu, Alois Knoll, Alessandro Crespi, and Auke J. Ijspeert. A smoothed particle hydrodynamics framework for fluid simulation in robotics. *Robotics and Autonomous Systems*, 185:104885, 2025. ISSN 0921-8890. doi: 10.1016/j.robot.2024.104885.
- [5] Vladimir Igorevich Arnold and Boris A Khesin. *Topological Methods in Hydrodynamics*, volume 19. Springer, 2009.
- [6] David Baraff. Linear-time dynamics using Lagrange multipliers. Technical Report CMU-RI-TR-95-44, Carnegie Mellon University, 1996.
- [7] D. N. Beal, F. S. Hover, M. S. Triantafyllou, J. C. Liao, and G. V. Lauder. Passive propulsion in vortex wakes. *Journal of Fluid Mechanics*, 549:385–402, 2006. doi: 10.1017/S0022112005007925.
- [8] Caleb Christianson, Yi Cui, Michael Ishida, Xiaobo Bi, Qiang Zhu, Geno Pawlak, and Michael Tolley. Cephalopod-inspired robot capable of cyclic jet propulsion through shape change. *Bioinspiration & biomimetics*, 16, September 2020. doi: 10.1088/1748-3190/abbc72.
- [9] COMSOL AB. COMSOL multiphysics® v. 6.4, 2025.
- [10] Daniele Costa, Giacomo Palmieri, Matteo-Claudio Pappacelli, David Scaradozzi, and Massimo Callegari. Design of a Carangiform Swimming Robot through a Multiphysics Simulation Environment. *Biomimetics (Basel, Switzerland)*, 5(4), September 2020. ISSN 2313-7673. doi: 10.3390/biomimetics5040046.
- [11] Paolo Domenici and Melina E. Hale. Escape responses of fish: A review of the diversity in motor control, kinematics and behaviour. *Journal of Experimental Biology*, 222(18):jeb166009, September 2019. ISSN 0022-0949. doi: 10.1242/jeb.166009.
- [12] Tao Du, Kui Wu, Andrew Spielberg, Wojciech Matusik, Bo Zhu, and Eftychios Sifakis. Functional optimization of fluidic devices with differentiable stokes flow. *ACM Transactions on Graphics (TOG)*, 39(6):1–15, 2020.
- [13] Joel H Ferziger, Milovan Perić, and Robert L Street. *Computational Methods for Fluid Dynamics*. springer, 2019.
- [14] F.E. Fish and G.V. Lauder. Passive and Active Flow Control by Swimming Fishes and Mammals. *Annual Review of Fluid Mechanics*, 38(Volume 38, 2006):193–224, 2006. ISSN 1545-4479. doi: 10.1146/annurev.fluid.38.050304.092201.
- [15] Amy Gao and Michael S. Triantafyllou. Independent caudal fin actuation enables high energy extraction and control in two-dimensional fish-like group swimming. *Journal of Fluid Mechanics*, 850:304–335, 2018. doi: 10.1017/jfm.2018.456.
- [16] M. Gazzola, W. M. Van Rees, and P. Koumoutsakos. C-start: Optimal start of larval fish. *Journal of Fluid Mechanics*, 698:5–18, 2012. doi: 10.1017/jfm.2011.558.
- [17] Genesis Authors. Genesis: A Universal and Generative Physics Engine for Robotics and Beyond, December 2024.
- [18] J. Gray. Studies in Animal Locomotion: VI. The Propulsive Powers of the Dolphin. *Journal of Experimental Biology*, 13(2):192–199, April 1936. ISSN 0022-0949. doi: 10.1242/jeb.13.2.192.
- [19] Jaskaran Grover and Daniel Vedova. Motion planning, design optimization and fabrication of ferromagnetic swimmers. In *Robotics Science and Systems*, 2019.
- [20] Jaskaran Grover, Jake Zimmer, Tony Dear, Matthew Travers, Howie Choset, and Scott David Kelly. Geometric Motion Planning for a Three-Link Swimmer in a Three-Dimensional low Reynolds-Number Regime. In *2018 Annual American Control Conference (ACC)*, pages 6067–6074, 2018. doi: 10.23919/ACC.2018.8431828.
- [21] J. L. Guermond, P. Mineev, and Jie Shen. An overview of projection methods for incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 195(44):6011–6045, 2006. ISSN 0045-7825. doi: 10.1016/j.cma.2005.10.010.
- [22] Ernst Hairer and Gerhard Wanner. *Solving Ordinary Differential Equations II*, volume 14 of *Springer Series in Computational Mathematics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1996. ISBN 978-3-642-05220-0 978-3-642-05221-7. doi: 10.1007/978-3-642-05221-7.
- [23] Matthias Heil, Andrew L. Hazel, and Jonathan Boyle. Solvers for large-displacement fluid–structure interaction problems: Segregated versus monolithic approaches. *Computational Mechanics*, 43(1):91–101, December 2008. ISSN 1432-0924. doi: 10.1007/s00466-008-0270-6.
- [24] Taylor A Howell, Simon Le Cleac’, J Zico Kolter, Mac Schwager, and Zachary Manchester. Dojo: A Differentiable Simulator for Robotics. 2022.
- [25] Yuanming Hu, Yu Fang, Ziheng Ge, Ziyin Qu, Yixin Zhu, Andre Pradhana, and Chenfanfu Jiang. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Transactions on Graphics (TOG)*, 37(4):1–14, 2018.
- [26] Jeffrey Ichnowski, Yahav Avigal, Yi Liu, and Ken Goldberg. Gomp-fit: Grasp-optimized motion planning for fast inertial transport. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 5255–5261.

- IEEE, 2022.
- [27] Brian E. Jackson, Kevin Tracy, and Zachary Manchester. Planning With Attitude. *IEEE Robotics and Automation Letters*, 6(3):5658–5664, 2021. doi: 10.1109/LRA.2021.3052431.
 - [28] Yusheng Jiao, Feng Ling, Sina Heydari, Nicolas Heess, Josh Merel, and Eva Kanso. Learning to swim in potential flow. *Physical Review Fluids*, 6(5), May 2021. doi: 10.1103/physrevfluids.6.050505.
 - [29] Robert K. Katzschmann, Joseph DelPreto, Robert MacCurdy, and Daniela Rus. Exploration of underwater life with an acoustically controlled soft robotic fish. *Science Robotics*, 3(16):ear3449, 2018. doi: 10.1126/scirobotics.aar3449.
 - [30] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. RMA: Rapid Motor Adaptation for Legged Robots, 2021.
 - [31] Joseph Louis Lagrange. *Mécanique Analytique*, volume 1. Mallet-Bachelier, 1853.
 - [32] Alex LaGrassa, Moonyoung Lee, and Oliver Kroemer. Task-Oriented Active Learning of Model Preconditions for Inaccurate Dynamics Models, 2024.
 - [33] Jeong Hun Lee, Mike Y. Michelis, Robert Katzschmann, and Zachary Manchester. Aquarium: A Fully Differentiable Fluid-Structure Interaction Solver for Robotics Applications. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11272–11279, 2023. doi: 10.1109/ICRA48891.2023.10161494.
 - [34] Adrian Lew, Jerrold E Marsden, Michael Ortiz, and Matthew West. An overview of variational integrators. *Finite element methods*, pages 98–115, 1970.
 - [35] Michael James Lighthill. Large-amplitude elongated-body theory of fish locomotion. *Proceedings of the Royal Society of London. B. Biological Sciences*, 179(1055):125–138, November 1971. ISSN 0080-4649. doi: 10.1098/rspb.1971.0085.
 - [36] Xiaozhu Lin, Xiaopei Liu, and Yang Wang. Learning Agile Swimming: An End-to-End Approach Without CPGs. *IEEE Robotics and Automation Letters*, 10(2): 1992–1999, 2025. doi: 10.1109/LRA.2025.3527757.
 - [37] Wenji Liu, Kai Bai, Xuming He, Shuran Song, Changxi Zheng, and Xiaopei Liu. FishGym: A High-Performance Physics-based Simulation Framework for Underwater Robot Learning. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6268–6275, 2022. doi: 10.1109/ICRA46639.2022.9812066.
 - [38] Miles Macklin. Warp: A High-performance Python Framework for GPU Simulation and Graphics, March 2022.
 - [39] Miles Macklin, Matthias Müller, Nuttapong Chentanez, and Tae-Yong Kim. Unified particle physics for real-time applications. *ACM Trans. Graph.*, 33(4), July 2014. ISSN 0730-0301. doi: 10.1145/2601097.2601152.
 - [40] Audrey P. Maertens, Amy Gao, and Michael S. Triantafyllou. Optimal undulatory swimming for a single fish-like body and for a pair of interacting swimmers. *Journal of Fluid Mechanics*, 813:301–345, 2017. doi: 10.1017/jfm.2016.845.
 - [41] JE Marsden and M West. Discrete mechanics and variational integrators. *Acta Numerica*, pages 357–514, 2001.
 - [42] Takahiro Miki, Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning robust perceptive locomotion for quadrupedal robots in the wild. *Science Robotics*, 7(62):eabk2822, 2022. doi: 10.1126/scirobotics.abk2822.
 - [43] Jason G Miles and Nicholas A Battista. Don’t be jelly: Exploring effective jellyfish locomotion. *arXiv preprint arXiv:1904.09340*, 2019.
 - [44] Rajat Mittal and Jung Hee Seo. Origin and evolution of immersed boundary methods in computational fluid dynamics. *Phys. Rev. Fluids*, 8(10):100501, October 2023. doi: 10.1103/PhysRevFluids.8.100501.
 - [45] Prapanch Nair and Gaurav Tomar. Volume conservation issues in incompressible smoothed particle hydrodynamics. *Journal of Computational Physics*, 297:689–699, 2015. ISSN 0021-9991. doi: 10.1016/j.jcp.2015.05.042.
 - [46] Elvis Nava, John Z Zhang, Mike Yan Michelis, Tao Du, Pingchuan Ma, Benjamin F Grewe, Wojciech Matusik, and Robert Kevin Katzschmann. Fast Aquatic Swimmer Optimization with Differentiable Projective Dynamics and Neural Network Hydrodynamic Models. In *International Conference on Machine Learning*, pages 16413–16427. PMLR, 2022.
 - [47] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, second edition, 2006.
 - [48] Guido Novati, Siddhartha Verma, Dmitry Alexeev, Diego Rossinelli, Wim M Van Rees, and Petros Koumoutsakos. Synchronisation through learning for two self-propelled swimmers. *Bioinspiration & biomimetics*, 12(3):036001, 2017.
 - [49] Michael O’Connell, Guanya Shi, Xichen Shi, Kamyar Azizzadenesheli, Anima Anandkumar, Yisong Yue, and Soon-Jo Chung. Neural-Fly enables rapid learning for agile flight in strong winds. *Science Robotics*, 7(66): eabm6597, 2022. doi: 10.1126/scirobotics.abm6597.
 - [50] Edwin Olson. AprilTag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation*, pages 3400–3407, 2011. doi: 10.1109/ICRA.2011.5979561.
 - [51] Francisco Palacios, Thomas D Economou, Aniket Aranake, Sean R Copeland, Amrita K Lonkar, Trent W Lukaczyk, David E Manosalvas, Kedar R Naik, Santiago Padron, Brendan Tracey, et al. Stanford university unstructured (SU2): Analysis and design technology for turbulent flows. In *52nd Aerospace Sciences Meeting*, page 0243, 2014.
 - [52] T. C. Papanastasiou, N. Malamataris, and K. Ellwood. A new outflow boundary condition. *International Journal for Numerical Methods in Fluids*, 14(5):587–608, 1992. doi: 10.1002/fld.1650140506.
 - [53] Charles S Peskin. The immersed boundary method. *Acta*

- numerica*, 11:479–517, 2002.
- [54] Yiling Qiao, Junbang Liang, Vladlen Koltun, and Ming Lin. Differentiable simulation of soft multi-body systems. *Advances in Neural Information Processing Systems*, 34:17123–17135, 2021.
 - [55] Stefan Schuster. The archerfish predictive C-start. *Journal of comparative physiology. A, Neuroethology, sensory, neural, and behavioral physiology*, 209(5):827–837, September 2023. ISSN 1432-1351 0340-7594. doi: 10.1007/s00359-023-01658-2.
 - [56] Robert L Seliger and Gerald Beresford Whitham. Variational principles in continuum mechanics. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 305(1480):1–25, 1968.
 - [57] Haochen Shi, Huazhe Xu, Samuel Clarke, Yunzhu Li, and Jiajun Wu. Robocook: Long-horizon elasto-plastic object manipulation with diverse tools. *arXiv preprint arXiv:2306.14447*, 2023.
 - [58] Pratik Suchde, Christian Leithäuser, Jörg Kuhnert, and Stéphane P. A. Bordas. Volume and Mass Conservation in Lagrangian Meshfree Methods. *International Journal for Numerical Methods in Engineering*, 126(2):e7657, 2025. doi: 10.1002/nme.7657.
 - [59] Kunihiro Taira and Tim Colonius. The immersed boundary method: A projection approach. *Journal of Computational Physics*, 225(2):2118–2137, 2007. ISSN 0021-9991. doi: 10.1016/j.jcp.2007.03.005.
 - [60] Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012. doi: 10.1109/IROS.2012.6386109.
 - [61] M. S. Triantafyllou, G. S. Triantafyllou, and D. K. P. Yue. Hydrodynamics of Fishlike Swimming. *Annual Review of Fluid Mechanics*, 32(1):33–53, 2000. doi: 10.1146/annurev.fluid.32.1.33.
 - [62] Michael Triantafyllou and George Triantafyllou. An Efficient Swimming Machine. *Scientific American - SCI AMER*, 272:64–70, March 1995. doi: 10.1038/scientificamerican0395-64.
 - [63] Roberto Venturelli, Otar Akanyeti, Francesco Visentin, Jaas Ježov, Lily D Chambers, Gert Toming, Jennifer Brown, Maarja Kruusmaa, William M Megill, and Paolo Fiorini. Hydrodynamic pressure sensing with an artificial lateral line in steady and unsteady flows. *Bioinspiration & Biomimetics*, 7(3):036004, April 2012. doi: 10.1088/1748-3182/7/3/036004.
 - [64] Roberto Verzicco. Immersed Boundary Methods: Historical Perspective and Future Outlook. *Annual Review of Fluid Mechanics*, 55(Volume 55, 2023):129–155, 2023. ISSN 1545-4479. doi: 10.1146/annurev-fluid-120720-022129.
 - [65] P. W. Webb. Fast-Start Performance and Body form In Seven Species of Teleost Fish. *Journal of Experimental Biology*, 74(1):211–226, June 1978. ISSN 0022-0949. doi: 10.1242/jeb.74.1.211.
 - [66] Carl H White, George V Lauder, and Hilary Bart-Smith. Tunabot Flex: A tuna-inspired robot with body flexibility improves high-performance swimming. *Bioinspiration & Biomimetics*, 16(2):026019, March 2021. doi: 10.1088/1748-3190/abb86d.
 - [67] Zhou Xian, Bo Zhu, Zhenjia Xu, Hsiao-Yu Tung, Antonio Torralba, Katerina Fragkiadaki, and Chuang Gan. FluidLab: A Differentiable Environment for Benchmarking Complex Fluid Manipulation, 2023.
 - [68] Eliot Xing, Vernon Luk, and Jean Oh. Stabilizing Reinforcement Learning in Differentiable Multiphysics Simulation, 2024.
 - [69] Zhenjia Xu, Cheng Chi, Benjamin Burchfiel, Eric Cousineau, Siyuan Feng, and Shuran Song. DextAIRity: Deformable Manipulation Can be a Breeze, 2022.
 - [70] Yingchen Yang, Nam Nguyen, Nannan Chen, Michael Lockwood, Craig Tucker, Huan Hu, Horst Bleckmann, Chang Liu, and Douglas L Jones. Artificial lateral line with biomimetic neuromasts to emulate fish sensing. *Bioinspiration & Biomimetics*, 5(1):016001, January 2010. doi: 10.1088/1748-3182/5/1/016001.
 - [71] Zhong Yang, Mohammad N. Saadatzi, Ruoshi Zhang, Andriy Sherehiy, Danming Wei, Cindy K. Harnett, and Dan O. Popal. Multiphysics Dynamic Model Validation Methodology for Laser-Driven Microrobots. In *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, pages 1555–1561, 2019. doi: 10.1109/COASE.2019.8843032.
 - [72] Uksang Yoo, Adam Hung, Jonathan Francis, Jean Oh, and Jeffrey Ichnowski. RoPotter: Toward Robotic Pottery and Deformable Object Manipulation with Structural Priors. In *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*, pages 843–850. IEEE, 2024.
 - [73] John Z. Zhang, Taylor A. Howell, Zeji Yi, Chaoyi Pan, Guanya Shi, Guannan Qu, Tom Erez, Yuval Tassa, and Zachary Manchester. Whole-Body Model-Predictive Control of Legged Robots with MuJoCo, 2025.