Soft robotics and the quest for modeling the physics of embodied intelligence

Gianmarco Mengaldo[1*], Federico Renda[2], Steven Brunton[3], Moritz Bächer[4], Marcello Calisti[5], Christian Duriez[6], Gregory S. Chirikjian[1] and Cecilia Laschi[1*]

[1]National University of Singapore, Singapore, SG.
[2]Khalifa University, Abu Dhabi, UAE.
[3]University of Washington, Seattle, USA.
[4]Disney Research, Zurich, Switzerland.
[5]University of Lincoln, Lincoln, UK.
[6]INRIA, Lille, France.

*Corresponding author(s): mpegim@nus.edu.sg; mpeclc@nus.edu.sg;
Contributing authors: federico.renda@ku.ac.ae; sbrunton@uw.edu; moritz.baecher@disneyresearch.com; mcalisti@lincoln.ac.uk; christian.duriez@inria.fr; mpegre@nus.edu.sg;

Abstract

Embodied intelligence, or intelligence that requires and leverages a physical body, is ubiquitous in biological systems, both in animals and plants. Through embodied intelligence, biological systems efficiently interact with and use their surrounding environment to let adaptive behaviour emerge. In soft robotics, this is a well-known paradigm, whose mathematical description and consequent computational modelling remain elusive. We argue that filling this gap will enable full uptake of embodied intelligence in soft robots. The resulting models can be used for design and control purposes. In this perspective, we provide a concise guide to the main mathematical modelling approaches, and consequent computational modeling strategies, that can be used to describe soft robots and their physical interactions with the surrounding environment, including fluid and solid media. The goal of this perspective is to convey the challenges and opportunities within the context of modeling the physical interactions underpinning embodied intelligence. We emphasize that interdisciplinary work is required, especially in the context of fully coupled robot-environment interaction modeling. Promoting this dialogue across disciplines is a necessary step to further advance the field of soft robotics.

Keywords: Soft robotics, Embodied intelligence, Mathematical modeling, Computational modeling, Computational physics

1 Introduction

Soft robotics is largely motivated by the functional role of soft tissues in living organisms [1]. Life has had millions of years to adapt to their surrounding environment and co-evolve nervous and muscle-skeletal systems to achieve task-efficient performance, synergistically. We observe that living beings are soft and compliant, and we argue that this is instrumental to their embodied intelligence [2]. According to this modern view of intelligence, the physical body play a much larger role in shaping intelligence, since a part of sensory–motor behaviour emerges from its interaction with the surrounding environment, with minimal or no involvement of the nervous system. Soft biological systems use their complex internal body structure to efficiently leverage physical interactions with the external environment and achieve the desired actions. Indeed, external interaction forces, instead of being treated as disturbances needing compensation, are used for the intended movements [3]. As an example, in locomotion, gravity is exploited for stepping forward, and adaptation to uneven terrains is provided by compliant elements within the leg joints, with limited need for active inputs from the central nervous system. Similarly, octopuses, an iconic model for soft robotics, adopt highly effective unfolding arm reaching movements by leveraging the buoyancy and interaction forces from the water surrounding them.

Embodied intelligence is a well-known paradigm in robotics and in soft robotics. In a typical robot sensory–motor behaviour scheme, we can consider a sensory system perceiving the environment, controllers processing the incoming information and planning a motor action, and then a mechanical system that executes motor actions in the physical environment. Embodied intelligence can be seen as the mechanical feedback from the environment, directly onto the mechanical system of the robot physical body, with no involvement of controllers or processing (see Fig. 1, from [2]). This view gives a clear perception of how powerful embodied intelligence can be in simplifying robot sensory–motor behaviour and increasing overall robot efficiency and effectiveness. All this works if we assume that a soft, compliant body receives, and does not reject, the mechanical feedback from the environment.

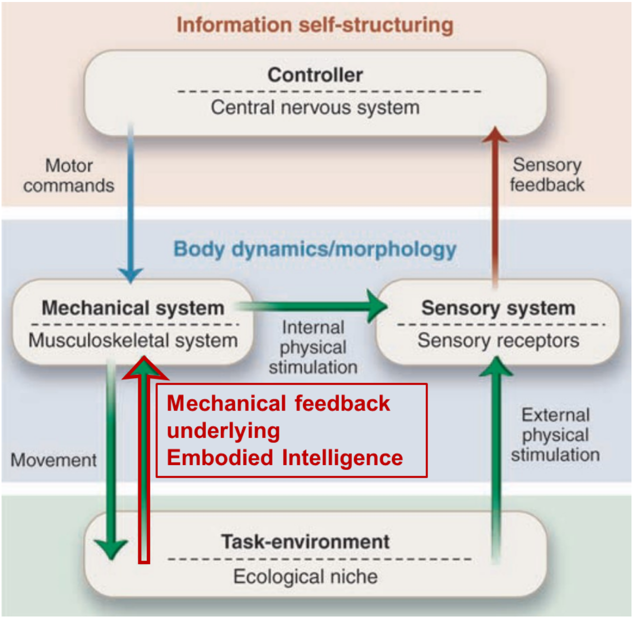


Fig. 1: In a typical robot sensory–motor scheme, embodied intelligence can be seen in the mechanical feedback received by the physical body from the environment. It allows closing a very short control loop, by-passing most of the computational processes. (Reprinted with permission from [2] and adapted).

How to systematically design embodied intelligence into soft robots is one of the current challenges in the field. We argue that this goal can be achieved by means of a mathematical description of the physical interactions characterizing embodied intelligence. Related computational modeling would enable simulations to be used for both design and control purposes.

The task of constructing this modeling framework is not an easy one. Take as an example a soft arm immersed in a fluid and interacting with solids (see, e.g., the octopus in Fig. 2). To describe and model the physics underlying embodied intelligence, it is necessary to consider:

- the continuous deformations of the arm deriving from muscle activations,
- the coupled interaction of the arm with the fluid, when e.g., the arm reaches out for target objects and moves the water, and the water contributes to its deformation,
- the interactions of the soft arm with solids (rigid or deformable), like the seafloor when walking, or external objects when grasping.

The description of these three points present a number of crucial challenges.

First, the problem is inherently multiphysics, due to the physically heterogeneous nature of interactions underlying embodied intelligence. In fact, one needs to take into account the physics of the soft body and the dynamics of muscle activation, the flow physics and its coupling with the soft body, and the physics describing contact and adhesion between two solids. This poses the critical challenge (and opportunity) of interdisciplinary work across multiple communities, ranging from robotics, fluid dynamics, structural mechanics, tribology, and contact mechanics. These fields typically have their own customs and terminology, and cross-fertilization can prove difficult, yet will be beneficial to bring advances in modeling embodied intelligence.

Second, the problem is multiscale, as the number of scales one needs to describe may range from millimeters to meters. For instance, in our example the arm reaching movement may undergo an overall displacement of several centimeters. Yet, the description of the deformation of the soft body, and its interaction with the flow and solid, may require a much finer description to accurately capture its behaviour. To this end, fast computational methods to solve multiscale problems are required. Integration with the scientific computing and applied mathematics communities may therefore prove important.

Multiphysics and multiscale problems are notoriously challenging but have successfully been conquered in some fields, including the aerospace industry, biomedical engineering, and material science, to cite a few. In the context of soft robotics, these problems have yet to be addressed, although promising advances have been made. Some of the factors that make modelling particularly difficult here are: i) how the actuation forces are designed and mathematically modelled to achieve a desired action, and how the surrounding environment influences them; ii) large deformations of nonlinear materials and their coupled interactions with the surrounding medium, which leads to high-dimensional models that are often prohibitively expensive to simulate and use inside control loops; and iii) partially known interfacial physics and unmodeled dynamics – e.g., nonlinear friction mechanisms for solid–solid interaction, and turbulent drag for fluid–structure interaction. In addition, the range of morphologies (e.g., arms, fingers, legs, fins), materials (e.g., hyperelastic, heterogeneous, functional), abilities (e.g., reaching, grasping, walking, morphing, growing, swimming, jumping, crawling, digging) and intended applications (e.g., healthcare, manufacturing, underwater sensing and manipulation, scientific exploration, entertainment) is extremely diverse, exacerbating the complexity of the modelling task. The latter aspect may lead to staggered and application-specific modelling strategies, that may not be beneficial to the soft robotics field.

In this paper, to support our argument of distilling embodied intelligence into physical interactions, we provide a concise guide to the latter and to their underlying mathematical models. In particular, we present and discuss the most prominent models that describe 1) the interactions of soft robot components (including actuators) that lead to efficient movements and deformation, and 2) the robot interactions with the surrounding environment. We refer to the former as *internal interactions*, and the latter as *external interactions*, where the environment can be constituted by fluid or a solid.[1] To introduce the models underlying the physical interactions found in embodied intelligence, we use the octopus depicted in Fig. 2 as a proxy for a general biological or soft robotics system.

[1]We note that the surrounding environment may be constituted of heterogeneous granular material. In this case, one may need to rely on different methods than the ones presented in this paper. Yet, the case of solid or fluid media is sufficiently general to provide an effective guide on modeling interactions between soft bodies and the environment.

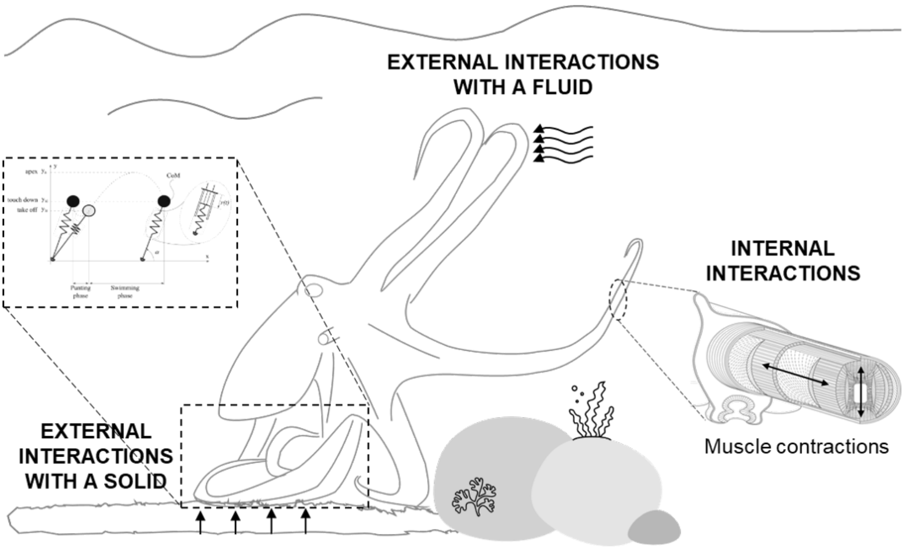


Fig. 2: An illustration of our scope in modeling the physics of embodied intelligence. We use an octopus as a proxy for a general soft body, in its environment, where we highlight the three key modeling areas that contribute to a substantial insight of embodied intelligence: *internal interactions*, and *external interactions* models, either with a fluid medium or a solid support. Internal interactions refer to the deformations of the soft body induced by actuators, or muscles. External interactions refer to the action of the external environment (fluid or solid) on the soft body. We also highlight how the accurate description of internal and external interactions can lead to low-dimensional models that capture the soft body behaviour with a reduced number of state variables.

1.1 A Mathematical Framework for Modeling of Soft Robots

The mathematical framework adopted belongs to the realm of mechanics that is at the foundations of physical interactions in both solids and fluids. To describe the mechanics of physical systems there are two views, Lagrangian and Eulerian. In the former, one tracks the trajectories of particles, while in the latter one observes the particle velocities at a fixed point in space. The

Lagrangian setting is commonly used in classical solid mechanics, while the Eulerian setting is commonly used in fluid mechanics. These two views allow us to introduce an abstract mathematical formalism for the soft body subject to *internal* and *external* interactions that consists of a set of differential equations:

$$\mathcal{D}\mathbf{q}_{\mathrm{sb}} = \mathcal{N}_{\mathrm{sb}} + \mathcal{C}_{\mathrm{int}} + \mathcal{C}_{\mathrm{ext}}, \quad \text{in } \Omega_{\mathrm{sb}} \tag{1}$$

Equation (1), along with suitable initial and boundary conditions, can describe both models arising in continuum mechanics and multibody dynamics in the soft body domain Ω_{sb}. The former typically yields a set of partial differential equations, while the latter a set of ordinary differential equations. Depending on the approach used, the variables $\mathbf{q}_{\mathrm{sb}}$, that describe the soft body (sb), can have a different meaning. For instance, in a Lagrangian view, $\mathbf{q}_{\mathrm{sb}}$ represents the position and momentum of material particles, while in an Eulerian view, $\mathbf{q}_{\mathrm{sb}}$ is the observed velocity at each given point. $\mathcal{D}$ is a differential operator that may be a partial (∂) or a total (d) derivative (including a material derivative D), of first or second order.

$\mathcal{N}_{\mathrm{sb}}$ is a nonlinear term that describes the soft body (sb) mechanics, which may depend on $\mathbf{q}_{\mathrm{sb}}$, its partial derivatives with respect to spatial coordinates $\mathbf{x}$, time t, and by a set of tunable constants such as viscosity, stiffness, and other actuation and material parameters.

$\mathcal{C}_{\mathrm{int}}$ is a coupling term that accounts for internal (int) interactions (e.g., actuation forces for tendons or pressure of pneumatic chambers). $\mathcal{C}_{\mathrm{ext}}$ is a coupling term that accounts for external (ext) interactions (e.g., contact with external solids or interactions with the surrounding medium).

While equation (1) may not be the standard way of introducing models in the realm of soft robotics, it allows a sufficiently general mathematical framework that can be used as a basis to construct a unified formalism for the multiphysics of internal and external interactions. We divided the right-hand side into a nonlinear term $\mathcal{N}_{\text{sb}}$, and two coupling terms $\mathcal{C}_{\text{int}}$ and $\mathcal{C}_{\text{ext}}$, to emphasize the mechanics of the soft body (through $\mathcal{N}_{\text{sb}}$) and its coupling not only with internal (through $\mathcal{C}_{\text{int}}$) but also with external systems (through $\mathcal{C}_{\text{ext}}$). In particular, $\mathcal{C}_{\text{int}}$, and $\mathcal{C}_{\text{ext}}$ can constitute forcing terms, that results from lumping the interactions into simplified terms, or can be terms coupling the equation describing the soft body (1) to additional equations describing the physics of internal and/or external interactions (e.g., actuators, and the surrounding environment). These interactions could be for instance described by equality and inequality constraints, especially when the soft body is in contact with a medium. In this case, the constraints can be embedded into the equations by means of, e.g., Lagrange multipliers, and encapsulate the physics of interactions. For example, frictional contact between two surfaces is best modeled with inequality constraints which become active as soon as a soft body is in contact with either itself or the environment.

Starting from equation (1), in the following we introduce the models and solution methods adopted for *internal* and *external* interactions. More specifically, the outline of this paper is as follows. In section 2, we detail the models for internal interactions, describing both the models for soft-body mechanics (i.e., $\mathcal{N}_{\text{sb}}$), and the internal interactions with actuators (i.e., $\mathcal{C}_{\text{int}}$). In section 3, we describe the models for external interactions (i.e., $\mathcal{C}_{\text{ext}}$). In section 4, we outline how these models can be used in practice for soft robotics. In section 5, we review the modeling challenges and opportunities described, and draw some conclusions and perspectives.

2 Internal interactions

2.1 Challenges

A soft-bodied system, such as the arm of the octopus depicted in Fig. 2, is composed of a set of sensors and actuators, distributed inside a soft tissue, that work together to achieve a given task. The various sensing and actuation mechanisms interact internally with one another, and they typically operate within a nonlinear and possibly heterogeneous material constituting the soft tissue. We focus here on the fundamental robotics task of modeling the relations between the actuator space and the soft-body deformations. A mathematical model describing internal interactions should therefore capture the robot deformation (i.e., $\mathcal{N}_{\mathrm{sb}}$) produced by the internal actuation forces, where the latter also needs to be accurately represented (i.e., $\mathcal{C}_{\mathrm{int}}$).

The degrees of freedom represented by the internal interactions are only a part of the degrees of freedom of the system. The first step is to describe the physics of actuation in a subspace of the deformable system. Once this space is defined, the key challenges specific to modeling internal interactions are:

1. to model the physics of the actuation in such a way that it can be identified on the real system, and
2. to couple the two physics: the one of the deformation of the structure, and the one of the actuation.

To this end, we take into account the following three main actuation strategies:

Tendons (1D). We consider here a punctual action of a cable or similar tendon-like actuator, which we consider as one-dimensional (1D). If the tension is exerted by a tendon which pulls, it is relatively easy to describe the force field created on a soft robot structure, based on the geometrical path. When a series of tendons are placed on a robot and their input lengths need to be integrated, the problem becomes more complex: since the tendons are coupled, it is necessary to verify that they are tense to exert a force. The tension of one tendon on a structure could slack on other ones. Also the tension inside the tendon is a signed force: we can pull with a tendon, but not push. The algebraic equations, translating the motion of the tendon, lead to constraints of inequality and complementarity. Finally, more advanced models that could be used to replicate the mechanics of the tendons (extension, bending, internal friction, etc.) are neglected to focus on this tensile force.

Flexible fluidic actuators (2D). In this case, we consider a chamber that deforms when pressurized with a fluid, so that a pressure acts on a surface, and we assume it to be constrained to two-dimensional (2D) deformations. In the same way, for these fluidic actuators, it is possible to directly take into account the pressure exerted in cavities. This pressure is then integrated on the surface of the cavity to calculate a force field applied to the soft robot structure. This approach is often used for pneumatic actuation, based on pressure regulators. On the other hand, in the case of liquid injection, it is often the variation of the cavity volume to constitute the input of the model. As with the tendon case, it is then necessary to add an algebraic equation in the model translating the volume variation. Moreover, the weight exerted by the liquid on the structure can create additional deformations. Finally, one can imagine, in the longer term, coupling the deformation models with dynamic models of the fluid.

Smart materials (3D). Smart materials are a large class of materials responding to external stimuli, and we consider this as a three-dimensional (3D) actuator. Some smart materials, such as electro-active or electro-ionic polymers, have their internal stress field dependent on an electric field. Other materials, such as shape memory materials, have their constitutive law dependent on temperature. Materials can also be integrated into soft robots that react to a magnetic field. In all these cases (and possibly in many others, the field of smart material being vast), the problem is the coupling of the deformation equations with other equations of physics, like electric or magnetic fields or temperature diffusion, often at scales that are much smaller than the scale of the robot.

The modeling of those actuators constitutes a challenge *per se*, and their integration with different soft-body models can be non-trivial. In particular, soft robots are highly under-actuated (the degrees of freedom associated with actuation are significantly fewer than the degrees of freedom of the soft body). Therefore, it is frequently required to add algebraic constraints, often associated with Lagrange multipliers $\boldsymbol{\Lambda}$ and $\boldsymbol{\Theta}$, to drive the motion of the soft body within the actuator space. In this case, the coupling term with the actuators becomes $\mathcal{C}_{\text{int}} = \boldsymbol{\Lambda}^T\mathcal{I}_e + \boldsymbol{\Theta}^T\mathcal{I}_i$, where $\mathcal{I}_e$ and $\mathcal{I}_i$ are equality and inequality constraints, respectively. The method of the Lagrange multipliers is a strategy to enforce equality constraints to a functional, in this case equation (1).

2.2 Models

Two modeling approaches of internal interactions exist in robotics. The first, called direct modeling approach, starts from the knowledge of the motion and/or forces in the actuator space and from other unactuated degrees of freedom to compute the robot shape. The second, known as inverse modeling approach, starts from a desired shape or position of the robot and calculate the actuator space in order to reach the desired shape/position.

In this section we will focus on the direct modeling approach. The actuator forces will modify the static equilibrium or the dynamics of the robot.

In the following, we introduce the main models that can be used for describing the soft body mechanics $\mathcal{N}_{\mathrm{sb}}$ and how they can be integrated with the actuation strategies reported above, through $\mathcal{C}_{\mathrm{int}}$. We introduce continuum solid mechanics models (fully 3D or rod/shells) and finite-dimensional parametrization models and we keep their description at a level general enough to provide the reader with tools adaptable to most cases. We also introduce the possible use of data-driven methods for the same modelling problems.

2.2.1 Continuum three-dimensional solid mechanics models

The most general model that captures the full complexity of the soft body is given by three-dimensional continuum solid mechanics (see Fig. 3(a)). If we consider equation (1), taking an Eulerian view:

$\mathcal{D}$ becomes $\partial/\partial t$, the partial derivative with respect to time t,

$\mathbf{q}_{\mathrm{sb}}$ becomes $\mathbf{v}_{\mathrm{sb}}$, representing the velocity field of the soft body,

$\mathcal{N}_{\mathrm{sb}}$ becomes $\nabla \cdot \boldsymbol{\sigma}_{\mathrm{sb}}$, the divergence of the soft body stress tensor $\boldsymbol{\sigma}_{\mathrm{sb}}$.

This equation describes the balance of linear momentum, that is usually complemented by the balance of angular momentum, through $\boldsymbol{\sigma} = \boldsymbol{\sigma}^T$, and balance of mass, through $\partial\rho/\partial t + \rho(\nabla \cdot \mathbf{v}_{\mathrm{sb}}) = 0$. One can also consider adding the balance of energy. The constitutive relations for describing the soft-body material usually rely on both elastic and hyperelastic models, or ad-hoc constitutive models. These ad-hoc constitutive relations can originate from e.g., phenomenological experiments of a specimen and provide an accurate simulation of the soft body deformation [4]. Note that hyperelastic models sometimes have many parameters that are difficult to identify in practice. Moreover, other phenomena such as viscosity, plasticity or anisotropy need also to be considered into the constitutive equations. Anisotropy has, for example, a direct influence on the kinematics of the soft robot [5]. Finally, these deformation models are very sensitive to the boundary conditions, thus to the modeling of the actuators but also to external interactions.

All the three actuation strategies, tendons, fluidic actuators, and smart materials, can be adopted in conjunction with continuum solid mechanics. Their coupling with the soft body is obtained through the term $\mathcal{C}_{\text{int}}$. This can be constituted by Lagrange multipliers $\mathcal{C}_{\text{int}} = \mathbf{\Lambda}^T \mathcal{I}_e + \mathbf{\Theta}^T \mathcal{I}_i$ arising from a set of constraints imposed by the actuators on the soft body. They can also be imposed as a set of boundary conditions for equation (1).

Solution strategies rely on numerical methods, e.g., finite and spectral element methods, whereby the partial differential equations are solved at a set of prescribed nodes (or mesh). This is typically extremely expensive computationally, with a corresponding number of degrees of freedom in the order of $\mathcal{O}(N^3)$, where N is the number of nodes of the underlying mesh.

Successful implementations in the realm of soft robotics applications exist. The software *SOFA* implements the finite element method to simulate continuum solid mechanics [6], and offers solutions, such as model reduction [7], to find a compromise between accuracy and computation time. *ChainQueen* implements a differentiable Lagrangian-Eulerian physical simulator based on the moving least square material point method for solid mechanics, along with actuation and contact with external objects [8]. *Evosoro* uses a solid mechanics engine, Voxelyze [9], that allows the simulation of soft multi-material robots [10]. These three options are open-source and currently under active development. Some additional commercial software to tackle the internal interaction modeling challenge exist, including *ABAQUS* [11], *ANSYS* [12], *COMSOL* [13], and Altair [14], among others. These provide platforms for solving full complexity solid mechanics problems, but they are not tailored to soft robotics.

This modeling approach, along with the computational strategies for solving it, allows the description of all topologies commonly required in soft robotics, e.g., rods, lattice of beams, shells, volumes. These topologies can be generated by means of Computer-Aided Design (CAD), and standard meshing practices. This however, may lead to an inexact geometrical representation of the soft body, that in turn can yield an inaccurate solution for the soft-body motion. This drawback could be addressed by using accurate mesh generation practices (e.g., high-order mesh generation [15, 16]) and by increasing the number of mesh nodes, or with isogeometric analysis [17].

2.2.2 Shell and rod models

A significant number of soft robots are characterized by an elongated structure with two dimensions much smaller compared to the third one. In this case, it is possible to significantly reduce the number of degrees of freedom required to describe the robot by adopting 1D rod models. Sometimes, only one dimension is negligible, and 2D shell models may be adopted. These fall under the realm of continuum mechanics although not in 3D.

A continuum modelling approach used in soft robotics is the Cosserat modeling approach, particularly useful to describe rods and shells. Here, we focus on this specific model, while acknowledging that there exist several others in practice, both for rods and shells. In the Cosserat model, the material point is replaced by a set of infinitesimal micro-solids stacked along the dominant dimension [18] that ensures high accuracy in simulating the geometric nonlinearities arising from the finite deformation of the robot [19]. The mathematical formalism underlying a Cosserat model relies on the Lie group of rigid body transformation SE(3), due to the assumptions made on the microstructures. Following [20], if we consider rods, a Cosserat rod is modeled by a continuous set of rigid cross sections stacked along a material line parameterized by a curvilinear coordinate $\mathbf{x} = X \in [0, 1]$ to which a cross-sectional frame $\mathcal{F}(X) = (O, t_1, t_2, t_3)(X)$ is attached, where $O(X)$ is on the midline, $t_1(X)$ is a unit normal vector perpendicular to the cross-section, and $t_2(X)$ and $t_3(X)$ are the unit vectors spanning the cross-sectional plane (see Fig. 3(b)).

The balance of momentum underlying this model can be recast within equation (1), where:

$\mathcal{D}$ becomes $\partial/\partial t$, the partial derivative with respect to time t,

$\mathbf{q}_{\text{sb}}$ becomes $[v, \quad \omega]^T$, the velocity twist field of the soft-body composed of the linear and angular velocity, respectively,

$\mathcal{N}$ becomes $[\mathcal{N}_{\text{sb},v}, \quad \mathcal{N}_{\text{sb},\omega}]$, with

$\mathcal{N}_{\text{sb},v} = 1/(\rho A)(\partial n/\partial x + \bar{n})$, and

$\mathcal{N}_{\text{sb},\omega} = (I^{-1}/\rho)(-\omega \times (\rho I \omega) + \partial c/\partial X + \partial r/\partial X \times n + \bar{c})$.

In the above equations, ρ is the density of the medium, I is the moment of inertia, A is the area of the cross section, r is the position vector of $O(X)$, while n and c are the linear and angular cross-sectional stresses along the beam.

Tendon actuators can be easily modeled in this framework as active internal wrenches $\mathcal{C}_{\text{int}}$ that act directly on the cross-sectional stress. Similar equations can be derived for Cosserat shells [21]. This modeling framework is not suitable for inflated chambers and some kind of smart material actuations, that inherently require a 3D description of the soft body.

Solution strategies for Cosserat-like models require the use of numerical methods, as they belong to the realm of continuum mechanics. To this end, methods introduced in the previous section are all suitable, i.e., finite and spectral element methods, and they are usually formulated in generalized coordinates. More recently, Cosserat models have been formulated in their strong form (as opposed to weak form, that is the basis for finite and spectral element methods) on tree-like structures, thanks to the use of reduced coordinates. An approach that has been traditionally applied in robotics.

In addition to the many research papers based on the Cosserat approach, few software and toolboxes have been proposed recently. Elastica [22] is a free and open-source software, which provides a discrete differential geometry approximation of the Cosserat rod model. SoRoSim [23] is a MATLAB toolbox that implements a strain-parametrization of tendon-driven soft robotic arms. A piecewise-constant strain approximation has also been included as a plug-in in SOFA [24].

Although Cosserat-like rod and shell models guarantee a good accuracy for soft robots with one or two main dimensions (e.g., slender bodies) undergoing finite deformation, they are not suitable for full 3D soft robot body deformations.

2.2.3 Finite-dimensional parametrization models

The previous models derive from continuum solid mechanics, therefore they lead to sets of partial differential equations, as described in sections 2.2.1 and 2.2.2, that are computationally expensive to solve. It is also possible to describe the behaviour of the soft body via finite dimensional models. These rely on a description of the soft body that leverages a suitably parametrized central axis or "backbone curve", that assumes a prescribed expression. This description leads to sets of ordinary differential equations in contrast to partial differential equations produced by the previous two modeling strategies.

The parametrization of the backbone curve can be defined as follows. Each frame $\mathcal{F}(X)$ described in the previous section can be viewed as an element of the group of special Euclidean transformations (i.e., rigid-body displacements). The origin of this frame traces out the backbone curve as the curvilinear coordinate X varies. The tangent to this curve is the normal to the cross-sectional plane, $t_1(X)$ (see Fig. 3(c)). For each fixed value of time, the combined rigid-body instantaneous translational and rotational rate of change with respect to arclength is defined by matrix $\Theta(X) = [\mathcal{F}(X)]^{-1}(\partial\mathcal{F}(X)/\partial X)$ where $\mathcal{F}(X)$ is the roto-translational matrix describing the orientation and translation of the frame (relative to the world frame at the base of the robot), with components $\mathbf{R}(X)$ and $\mathbf{p}(X)$, respectively [25–27]. In the case when the backbone curve is inextensible (not stretchable), then there is coupling between the rotation matrix and the position vector. For instance, if the tangent to the soft robot arm at its base is $\mathbf{e}_1 = [1, 0, 0]^T$, then the tangent at X will be $\mathbf{R}(X)\mathbf{e}_1$ and integrating the tangent along the curve generates the position as $\mathbf{p}(X) = \int_0^X \mathbf{R}(s)\mathbf{e}_1 ds$. In general, it is always possible to use $\Theta(X)$, that is a coordinate-free parametrization. However, when there is coupling between the rotation matrix and the position vector, one can use an alternative parametrization of the backbone curve by expanding rates of rotation parameters such as Euler angles. In the planar case, they reduce to the same parametrization.

The matrix $\Theta(X)$ has embedded in it information about instantaneous rotational and translational changes as a function of arclength, which can be extracted to form a vector $\mathbf{q}_{\mathrm{sb}} = [v \quad \omega]^T$ alluded to in the previous section. The vector $\mathbf{q}_{\mathrm{sb}} = [v \quad \omega]^T$ can then be expanded using a modal approach, that describes the spatio-temporal behavior of v and ω. This modal expansion is commonly defined as $\mathbf{q}_{\mathrm{sb}}(X,t) = \sum_{i=1}^{n} \Phi_i(X)a_i(t)$, where $\Phi_i(X)$ are modes describing the spatial behaviour of the system, and $a_i(t)$ are coefficients describing its evolution in time. The v-part of this vector describes how the backbone curve stretches and how adjacent planar sections shear relative to each other. The ω-part of this vector describes bending and twisting, and is related to the classical concepts of curvature and torsion of space curves. The modal expansion includes as a special case piecewise-constant curvature models by choosing some functions $\Phi_i(X)$ to be piecewise constant for some values of X and equal to zero for others, as described in [26].

The parametrization and modal expansion introduced lead to a set of ordinary differential equations, that can be written using the formalism introduced in equation (1), where

- $\mathcal{D}$ becomes d/dt, the total derivative with respect to time t,
- $\widehat{\mathbf{q}}_{\rm sb}$ becomes $[v \quad \omega]^T$, and
- $\mathcal{N}$ becomes $-1/\mathbf{M}\,[\mathbf{D}(\mathbf{s}_{\rm sb}, \mathbf{q}_{\rm sb}) + \mathbf{K}(\mathbf{s}_{\rm sb})]$, where $\mathbf{s}_{\rm sb}$ is the displacement of the soft body.

The nonlinear term just introduced is derived from the robot dynamic model, where $\mathbf{M}$ is the inertial matrix, $\mathbf{D}$ is a dissipative term that includes internal friction and other dissipative forces (such as Coriolis forces), and $\mathbf{K}$ is an elastic term that encapsulate the stiffness of the system.

The coupling with actuation is achieved via an appropriate choice of the modal expansion, and it is commonly represented via $\mathcal{C}_{\rm int} = \boldsymbol{\alpha}$, where $\boldsymbol{\alpha}$ is the vector of actuation forces. For example, when motors specify the angles between rigid links of a traditional robot, the modes would be Dirac delta functions in ω to describe revolute joints or in v to describe prismatic joints, and the weights would be the joint angles. In this sense, the modal approach can be used not only for soft/continuum robots but also for ones composed of rigid links, as well as hybrids.

The solution of the models in section 2.2.2 requires the use of numerical methods, e.g., finite and spectral element approximations. In contrast, solution methods for ordinary differential equations arising in this section are based on iterative schemes that update the weights $a_i(t)$ in time. These involve inverse Jacobian iterations as described in the articles listed earlier in this section.

The computational benefit of this approach is that instead of solving partial differential equations, it averages properties over each section and reduces the problem to a finite number of ordinary differential equations. This approach has been applied to hyper-redundant manipulators of all kinds (i.e., those consisting of a large number of rigid links, continuum filament, soft) over the past several decades. Successful implementation in the realm of soft robotics applications exist [28, 29].

2.2.4 Data-driven and machine learning models

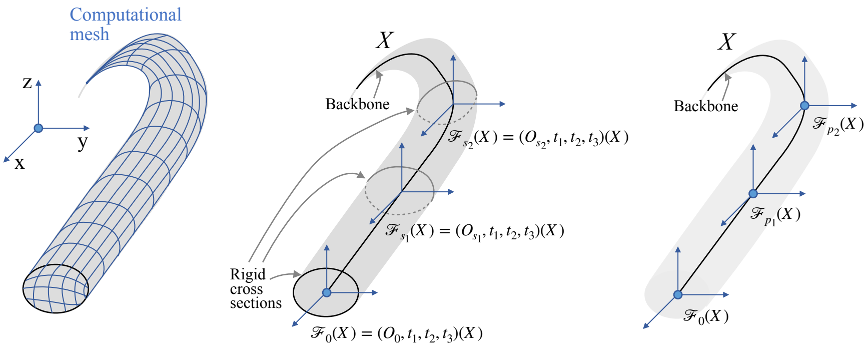
An alternative approach to the models above is the use of machine learning techniques. In robotics, learning has been widely used for building the kinematic and dynamic relations between a robot actuators and its position in space. Neural networks have been used to this aim, mainly for control purposes, implementing so-called neuro-controllers. In the same way, in soft robots, neural networks can be used to learn the soft body dynamics and solve the transformations from the actuator space to the space of the soft body deformation and position. Following the formalism in equation (1), one tries to learn the nonlinear and the coupling term for internal interactions, $\mathcal{N}$ and $\mathcal{C}_{\text{int}}$. This results in learning the right-hand side of equation (1) in one go, with e.g., a neural network (NN): $\mathcal{D}\mathbf{q}_{\text{sb}} = \text{NN}[\mathbf{q}_{\text{sb}}(k) \to \mathbf{q}_{\text{sb}}(k+1)]$, where k are different time instances.

In [30], an analysis is given of how learning-based blocks can replace some of the steps of the longer transformation chain involved in soft robot control. Diverse network topologies and learning paradigms can be used for this purpose. In [31], a quantitative survey is presented, showing how supervised learning is more widely used than unsupervised or reinforcement learning, with references to the different techniques used. The survey presented in

[32] describes the learning techniques used for obtaining the mapping from actuation space to task space in continuum robots.

An interesting comparison between a model-based and a learning-based approach to the control of a same soft robot arm is presented in [33]. While the model-based method is more accurate in controlling the end-effector position, as resulting in simulation, its error tends to increase together with the model inaccuracies, e.g. fabrication inaccuracies. The learning-based controller error tends to be insensitive to such modeling inaccuracies, especially if training is accomplished on the physical robot arm.

Looking ahead, we envisage a promising direction of progress in the integration of modeling techniques with machine learning, where models can feed neural networks and learning can replace specific sub-system mappings.



(a) 3D continuum (b) Shells and rods (c) Parametrization

Fig. 3: A conceptual illustration of the models for internal interactions described in section 2, applied to the octopus arm of Fig. 2. (a) A representation of the mesh description in 3D continuum mechanics models. (b) Cosserat's approach for rods. (c) Finite-dimensional parametrization models.

3 External interactions

3.1 Challenges

To benefit from embodied intelligence, we require medium–robot interaction modeling that can grasp the emergence of sensory–motor behaviour from the interplay with the surrounding environment. Modeling these external interactions, and coupling them with the internal interactions discussed in the previous section, closes the gap required to modeling the physics of embodied intelligence.

The task when modeling external interactions is to capture the complex, time-varying forces between the actuated soft-robot structures and the surrounding medium. The dynamics of fluids and deformable solids are generally nonlinear and unsteady, and interactions between the soft robot and the medium may create complex feedback loops and hysteresis effects. The key challenges to modeling external interactions are:

1. the multiphysics nature of the problem, that requires substantial interdisciplinary efforts, and
2. the partially known interfacial physics and unmodeled dynamics (e.g., nonlinear friction and turbulent drag).

In the following, we focus on medium-robot interaction, that we denote with the coupling term $\mathcal{C}_{\text{ext}}$ introduced in equation (1). Here, the surrounding medium is either a fluid or a solid. The robot (i.e. the soft body) is assumed to be described by the models presented in section 2, along with its actuation. In general, any of the soft-body models in section 2 can be used in conjunction with the models introduced next, unless otherwise specified.

3.2 Fluid models

When fluid forces acting on a robot have a tangible impact on movement, as with the octopus in Fig. 2, modeling fluid–robot interaction becomes essential. Here the task is to capture the complex, time-varying, bidirectional forces between the actuated soft robot structures and the surrounding fluid flow. To this end, the field of fluid–structure interaction [34] is a key cross-discipline that should be taken into account.

We describe three main modeling strategies, that can be adopted in the context of fluid–robot interaction. These include continuum fluid mechanics models, simplified lumped parameter models, and the relatively new field of machine learning for flow modeling. We remark that we keep the description at a general level, and we avoid entering into details that may hinder grasping the essence of these modeling strategies.

3.2.1 Continuum fluid mechanics models

In analogy to continuum solid mechanics, the equations governing flow physics in a continuum setting can be considered as the most general model of a fluid interacting with the soft body described in section 2 (see Fig. 4(a)). The equations describing continuum fluid mechanics are the Navier-Stokes equations, constituted, in the most general case, by conservation of mass, momentum and energy. From a soft robot perspective, the action of the fluid modeled by the Navier-Stokes equations emerges in the form of equality constraints at the interface between the soft body and the fluid, denoted by $\Gamma_{\mathrm{sb,f}}$.

In particular, we can write these interface constraints as follows:

$$\mathcal{I}_{\text{e,f}} = \begin{cases} \mathbf{q}_{\text{sb}} = \mathbf{q}_{\text{f}} & \text{on } \Gamma_{\text{sb,f}} \quad (2a) \\ \boldsymbol{\sigma}_{\text{sb}} \cdot \mathbf{n} = \boldsymbol{\sigma}_{\text{f}} \cdot \mathbf{n} & \text{on } \Gamma_{\text{sb,f}} \quad (2b) \\ \mathbf{x}_{\text{sb}} = \mathbf{x}_{\text{f}} & \text{on } \Gamma_{\text{sb,f}}. \quad (2c) \end{cases}$$

In equation (2), $\mathbf{q}_{\text{sb}} = \mathbf{v}_{\text{sb}}$ corresponds to the velocity of the soft body, and $\mathbf{q}_{\text{f}} = \mathbf{v}_{\text{f}}$ is the velocity of the fluid at the interface $\Gamma_{\text{sb,f}}$, $\mathbf{n}$ is the normal at any soft body location, $\boldsymbol{\sigma}_{\text{f}}$ is the fluid Cauchy stress tensor composed of the sum of the deviatoric stress tensor $\boldsymbol{\tau}^{(\text{f})}$ (that accounts for the viscosity) and a pressure term $-p\mathbf{I}$, and $\mathbf{x}_{\text{sb}}$ and $\mathbf{x}_{\text{f}}$ are the positions of the soft body and fluid interfaces, respectively. The solution for the solid velocity is achieved through the modeling strategies introduced in section 2. The solution for the fluid velocity is instead obtained by solving the Navier-Stokes equations. For the purpose of this article, we report the momentum equation, that can be written as follows $\partial(\rho_{\text{f}}\mathbf{q}_{\text{f}})/\partial \text{t} + \nabla \cdot (\rho_{\text{f}}\mathbf{q}_{\text{f}} \otimes \mathbf{q}_{\text{f}}) = \nabla \cdot \boldsymbol{\sigma}_{\text{f}} + \boldsymbol{\beta}_{\text{f}}$, where ρ_{f} is the fluid density, and $\boldsymbol{\beta}_{\text{f}}$ is a forcing term acting on the fluid (e.g., gravity). Depending on the nature of the problem, we can have several approximations, including incompressible Navier-Stokes equations, where the density is a constant, compressible Navier-Stokes equations, where the density is not constant and the flow might develop shock waves, thin-layer Navier-Stokes equations, that describes flow in thin layers, and potential flow equations that describe irrotational flows, to cite a few.

The compatibility conditions in equation (2) can enter into the equations describing the soft body (1) as Lagrange multipliers $\mathcal{C}_{\text{ext}} = \boldsymbol{\Lambda}^T \mathcal{I}_{\text{e,f}}$, or being imposed as external equality constraints $\mathcal{I}_{\text{e,f}}$ or as boundary conditions. Given the nature of the coupling, the majority of the soft body models presented in section 2 may be used, with the due approximations required for coupling. For instance, in the case of continuum solid mechanics (section 2.2.1), the compatibility conditions are applied to the interface nodes between the soft body mesh and the fluid mesh (or a proxy representing the interface, if the soft body / fluid nodes are not matching). In this case, there exist two mainstream approaches for coupling: partitioned and monolithic. The former uses a structural and a fluid solver iteratively until convergence, while the latter solves a fully coupled system of equations in one go [35]. When using continuum fluid mechanics in conjunction with simplified methods, such as finite-dimensional parametrization models (section 2.2.3), one instead needs to lump the effect of the fluid solution into a set of degrees of freedom compatible with the finite expansions used to describe the deformation of the body.

In terms of solution strategies, there exist well-established approaches in the realm of fluid–structure interaction methods. The two main approaches are conforming and non-conforming methods [35]. The former enforces equations (2b) and (2c), while the latter enforces equation (2a). Conforming methods can be used in conjunction with partitioned methods, where a popular algorithm is the arbitrary Lagrangian Eulerian approach [36]. Non-conforming methods are typically used in conjunction with monolithic methods, where a popular algorithm is the immersed boundary method [37–39].

While the successful implementation of continuum fluid mechanics and its interaction with soft bodies have been largely explored in biology [40, 41], in soft robotics it is still nascent. One of the key challenges is the computational costs of the simulations and an accurate coupling with the soft body dynamics and actuation system. To solve the equations arising in this modeling strategy, there exist both commercial and open-source multiphysics packages. These include ANSYS [12] and COMSOL [13], Altair [14], Flow3D [42], Adina [43], and OpenFOAM [44], to cite a few. However, these are not readily tuned for soft robotics applications. For example, implementing actuation strategies in a nonlinear elastic material interacting with a fluid described by the Navier-Stokes equations remain challenging.

Continuum fluid mechanics can fully capture the effect that a fluid has on a soft robot. However, their inherent computational costs make these models prohibitive to use for practical design and control purposes, and they have found limited use in soft robotics, as of today. As computational resources become more available and underlying algorithms more efficient, the scope of these models will expand and they may be eventually used for design and control purposes. Yet, today they can already be adopted for high-fidelity simulations that can grasp the complex behaviour of biological systems interacting with a fluid, and extract the key physical principles underlying their embodied intelligence. These principles can in turn improve simplified and less expensive models used for design and control.

3.2.2 Lumped parameter models

In some circumstances, it is acceptable and convenient to simplify the description of the fluid interacting with the soft body. In these cases, one no longer solves the equations governing continuum fluid mechanics, but aggregates the overall effect of the fluid into a set of lumped contributions. The main lumped contributions consist of added mass, drag/lift forces, and buoyancy (see Fig. 4(b)). These can be included as forces into the coupling term $\mathcal{C}_{\text{ext}}$ as follows: $\mathcal{C}_{\text{ext}} = \mathbf{f}_{\text{added mass}} + \mathbf{f}_{\text{drag}} + \mathbf{f}_{\text{lift}} + \mathbf{f}_{\text{buoyancy}}$, where the subscript of each term is self-explanatory. The added mass, $\mathbf{f}_{\text{added mass}}$, is the virtual mass or inertia added to the soft robot due its need to move the fluid surrounding it. For instance, if the soft body was a simple sphere immersed in an incompressible fluid, the added mass would be equal to $\mathbf{f}_{\text{added mass}} = \rho_{\text{f}} V_{\text{sb}}[\text{D}\mathbf{q}_{\text{f}}/\text{Dt} - \text{d}\mathbf{q}_{\text{sb}}/\text{dt}]$, where $\mathrm{D}\mathbf{q}_\mathrm{f}/\mathrm{Dt}$ is the material derivative of the fluid velocity, $\mathrm{d}\mathbf{q}_\mathrm{sb}/\mathrm{dt}$ is the total derivative of the spherical soft body velocity, ρ_f is the fluid density and V_sb is the volume of the spherical soft robot. Drag and lift forces are typically proportional to the velocity of the fluid flowing around the soft body, and they can be calculated as $\mathbf{f}_\mathrm{drag} = (1/2)\rho_\mathrm{f}\,\mathbf{q}_\mathrm{f}\,A_\mathrm{sb}\,C_\mathrm{drag}$, and $\mathbf{f}_\mathrm{lift} = (1/2)\rho_\mathrm{f}\,\mathbf{q}_\mathrm{f}\,A_\mathrm{sb}\,C_\mathrm{lift}$, where A_sb is the area of the soft body exposed to the fluid, and C_drag and C_lift are the drag and lift coefficients that, for simplified geometries, are commonly tabulated. Finally, the buoyancy term is typically accounted for as a net vertical force composed by buoyancy plus gravity.

The term $\mathcal{C}_\mathrm{ext}$ just defined enters into the right-hand side of the soft-body equation (1) as a forcing term, and can be used in conjunction with any of the models presented in section 2. Obviously, the lumped contributions need to correctly interface with the degrees of freedom of the soft body. In the case of continuum solid mechanics models, (i.e., the ones introduced in sections 2.2.1 and 2.2.2), the lumped forces need to be distributed onto the nodes underlying the discretization of the soft body. Similarly, for finite-dimensional parametrization models (i.e., the ones introduced in section 2.2.3), $\mathcal{C}_\mathrm{ext}$ needs to be applied to the degrees of freedom of the functional parametrization adopted.

The coupling term $\mathcal{C}_\mathrm{ext}$, and its constitutive components, (i.e., added mass, lift, drag and buoyancy) depend on the shape of the soft body, and therefore they can change over time as effect of actuations or interactions with the environment. Employing such changes to increase efficiency, to direct behaviours, or to improve performances is key to establish quantitative advantages of soft robotics.

In practice, the forces expressed in $\mathcal{C}_\mathrm{ext}$ are often coupled with continuum solid models of soft robots (i.e., the ones described in section 2.2.1 and 2.2.2), such that a complete coupling between continuum fluid mechanics models and continuum solid mechanical models is generally not performed due to its computational costs. Successful implementation of these strategies in soft robotics has been performed, for example in [45].

Eventually, even if external forces are included without modelling the continuum of the fluid, the dependence of such forces on shape-varying coefficients allows reaching a trade-off between accuracy and relevance of simulation, with the computational power available for modelling. Depending on the goal of the simulation, this is similar to fluid–structure interaction in the aerospace industry: while complete coupled simulations are used for design, flight simulators employ a similar coupling to match fidelity of simulation with real-time computation.

3.2.3 Data-driven and machine learning models

Recent data-driven and machine learning methods have become widely used for modeling complex fluids [46–48]. Several recent approaches have leveraged machine learning to direct speed up high-fidelity simulation of the Navier-Stokes equations, especially the ones involving turbulence [49–52]. Indeed, for complex flows, it is often impractical to resolve all scales of the flow, and instead researchers employ turbulence models, such as the Reynolds averaged Navier-Stokes (RANS) equations or large eddy simulation (LES). These approximate the smaller scales, and allow for less computationally expensive simulations of the Navier-Stokes equations. Machine learning is rapidly advancing these computational fields [46], providing enhanced data-driven turbulence closures [53–57]. For even further reduction in computation, it is often possible to develop reduced-order models (ROMs) that are tailored to a specific flow and provide an optimal balance between accuracy and efficiency. Reduced-order models are typically at least partially data-driven, as they are based on modal decompositions [58], and they have close connections to machine learning. Several recent approaches have provided more accurate and generalizable ROMs, for example based on sparse regression [59–63] and the use of deep neural networks to learn effective coordinate systems [64]. Other promising recent techniques for physics computation based on machine learning include physics-informed neural networks [65] and deep operator networks [66].

In addition to using machine learning techniques to develop surrogate models for the complex fluid environment, there is also a need to model discrepancies that arise in complex multiphysics applications. The equations of a simple fluid are relatively well understood, but interfacial dynamics, non-Newtonian fluids, and multiphase flows all pose significant modeling challenges even to represent the fundamental physical effects in a full-fidelity model, let alone in a reduced-order model. There are considerable efforts to use machine learning approaches to model these discrepancies between observed data and idealized physical models.

3.3 Solid models

A model of solid–robot interaction should accurately capture how normal forces, tangential forces or friction, and adhesion/cohesion forces affect and can therefore be leveraged by the soft body. To this end, contact mechanics [67] and tribology [68] represent the key broad cross-disciplines for modeling solid–robot interaction.

In the following, we outline the main modeling strategies that can be adopted in the context of solid–robot interaction. These include the use of continuum solid mechanics to model a deformable solid, simplified lumped parameter models, and more recent machine learning strategies. A key for these models to be accurate is related to the frictional interaction, and consequent constraints at the interface between the solid and the soft robot.

3.3.1 Continuum three-dimensional solid mechanics models

The models underlying continuum solid mechanics have been introduced in section 2.2.1, albeit for the soft robot. These models can be used also to describe a surrounding deformable solid medium (see Fig. 4(c)). Its interaction with the soft robot is described by the coupling term $\mathcal{C}_{\text{ext}}$ introduced in equation (1). This is typically the result of a series of constraints that need to be satisfied at the interface between the soft robot and the solid medium. Such constraints are similar to the one introduced in section 3.2.1, except that they need to account for complementarities, that is, when the system transition from no contact to contact, or from static friction (i.e., no tangential relative movement between two elastic surfaces) to dynamic fiction (i.e., relative tangential movement).

For instance, the normal contact between two elastic surfaces can be modeled using the Signorini's framework, that uses a distance function S_n to measure the distance between two surfaces along the normal direction. Signorini's formulation can be written as $S_n(\mathbf{q}) \perp \boldsymbol{\sigma}_n$, where $\boldsymbol{\sigma}_n$ is the stress along the normal direction $\mathbf{n}$ at the contact interface, and the symbol $\perp$ denotes complementarity: if $S_n(\mathbf{q}) > 0$ then $\boldsymbol{\sigma}_n = 0$ (no contact); if $S_n(\mathbf{q}) = 0$ then $\boldsymbol{\sigma}_n > 0$ (contact). Tangential or frictional contact can instead be modeled using the Coulomb's law framework, where the complementarity is given by the relation $\boldsymbol{\sigma}_t = \mu_s \boldsymbol{\sigma}_n$. Here, complementarity arises from static vs. dynamic friction. For $\boldsymbol{\sigma}_t \leq \boldsymbol{\sigma}_{\max}$ there is static friction (no relative sliding of the two surfaces), while for $\boldsymbol{\sigma}_t > \boldsymbol{\sigma}_{\max}$ there is dynamic friction (relative sliding of the two surfaces), where $\boldsymbol{\sigma}_{\max}$ is the maximum value of stress that allow static friction.

These interface conditions need to be encapsulated into the continuum solid mechanics models describing the soft robot and the elastic solid medium interacting with the robot, in a similar manner as described in section 3.2.1, for the constraints in equation (2). They can for instance be embedded into the equations through the term $\mathcal{C}_{\text{ext}} = \mathbf{\Lambda}\mathcal{I}_{\text{e,s}}$, using Lagrange multipliers for the constraints. However, in contrast to section 3.2.1, one needs to account for the complementarities in the contact and friction laws. These complementarities create jumps in velocities. When the coupled system transitions from one regime to another (e.g., from no contact to contact, or from static to dynamic friction), the inertial terms underlying the system of equations is not defined.

In order to solve this contact problem, it is necessary to rely on numerical methods that account for these complementarities, and the singularities they produce. To this end, a strategy is first to calculate $S_n(\mathbf{q})$ using a measurement of proximity distance, interpenetration distance, interpenetration volumes, or a precise measurement of the moment and the contact configuration between two simulation steps. Following this detection, a set of constraints are defined, often on contact points (but one can extend to volumes or to non-planar surfaces). Finally, the solution can be obtained with different numerical strategies such as Lagrange multipliers, penalty methods and augmented Lagrangian methods, in conjunction with optimization solvers dedicated to complementarity problems. The time-integration of the resulting equations is achieved via event-driven methods that account for the singularity in the underlying equations – see e.g., [69], [70].

An example of successful implementation of this strategy can be found in [71]. Here, the contact problem is formulated to solve an inverse model for the control of a soft robot, which leads to writing a quadratic problem with complementarity constraints (also referred to as QPCC).

Most of the major multiphysics computational software previously cited, including ABAQUS [11], ANSYS [12] and COMSOL [13], can be used to solve contact between two elastic solids, yet they are not necessarily tailored to soft robotics applications. In the article [72], a literature review of physical engines for simulation in robotic applications is listed, most of these engines support contact modeling. On the other hand, not all engines are adapted to deformable robots, in particular to simulate volume deformations. We can note the SOFA software [73], originally dedicated to medical simulation, offers plug-ins dedicated to soft robotics and proposes implementations of FEM and Cosserat rods that are compatible with contact modeling.

Similar considerations as for continuum fluid mechanics can be made. In particular, these models are computationally expensive, and they may lead to impractical solutions for soft robotics applications today, especially in the context of real-time control.

3.3.2 Lumped parameter models

Similarly to what we have introduced for fluid models, we can consider a simplified approach, where we can aggregate the effect of contact between two surfaces into lumped contributions. For instance, when the external solid can be considered rigid, one can introduce some simplifications to the model outlined in section 3.3.1. In particular, the state $\mathbf{q}_\mathrm{s}$ of a rigid body can be represented by the position of its center of mass, its orientation, and its linear and angular velocity (see Fig. 4(d)). Once these are known, it is possible to formulate the rigid-robot interaction problem. This implies identifying the point of contact between the soft body and the rigid solid, and applying an equivalent frictional force $\boldsymbol{f}$ and torque $\boldsymbol{\tau}$, induced by the interaction between the soft and the rigid body. The interaction surface between the rigid solid and the soft robot is typically modeled as a planar surface. This leads to models that contain only the three degrees of freedom associated to frictional forces at the planar surface, $\boldsymbol{f}$.

In practice, a soft robot interacting with a rigid body leads to non-planar contact surfaces, with multiple points of contact. This introduces three additional degrees of freedom, and provide a six-dimensional model for the normal and frictional wrenches, as both force $\boldsymbol{f}$ and torque $\boldsymbol{\tau}$ are three-dimensional. Following the formalism adopted in [74], one can define the normal force and torque as $\boldsymbol{f}_n = -\int_S p \cdot \boldsymbol{n} \mathrm{d}S$, and $\boldsymbol{f}_n = -\int_S p \cdot [(\boldsymbol{r} \times \boldsymbol{n}] \mathrm{d}S$, respectively, and the frictional force and torque as $\boldsymbol{f}_t = -\mu \int_S p \cdot \boldsymbol{v}_r \mathrm{d}S$, and $\boldsymbol{\tau}_t = -\mu \int_S p \cdot [\boldsymbol{r} \times \boldsymbol{v}_r] \, \mathrm{d}S$, respectively, where S is the contact surface, $\boldsymbol{v}_r$ is the relative velocity between the rigid solid and the soft robot, $\boldsymbol{n}$ is the normal to the contact surface, $\boldsymbol{r}$ is the torque arm, p is the contact pressure distribution, and μ is the friction coefficient.

The coupling is achieved by encapsulating these forces and torques into the coupling term $\mathcal{C}_{\text{ext}}$. This acts as a forcing term on the right-hand side of the soft body equation, that is $\mathcal{C}_{\text{ext}} = \boldsymbol{f}_n + \boldsymbol{\tau}_n + \boldsymbol{f}_t + \boldsymbol{\tau}_t$.

The computational costs associated with these models are relatively small compared to the interaction between two deformable solids.

Applications of this modeling strategies in soft robotics exists, both in terms of analytical approaches [75, 76] and computational models [6, 77], that focused on simulating contact behaviours of soft robot grasping and manipulation as well as crawling [78, 79]. Inverse model of soft robot in situation of frictionless contact [71] and adhesive contact situations has been investigated for control of manipulation and locomotion [80].

These models can be broadly applied for describing several tasks involving the interaction of the soft robot with e.g., rigid solids, and similar considerations as in section 3.2.2 can be made.

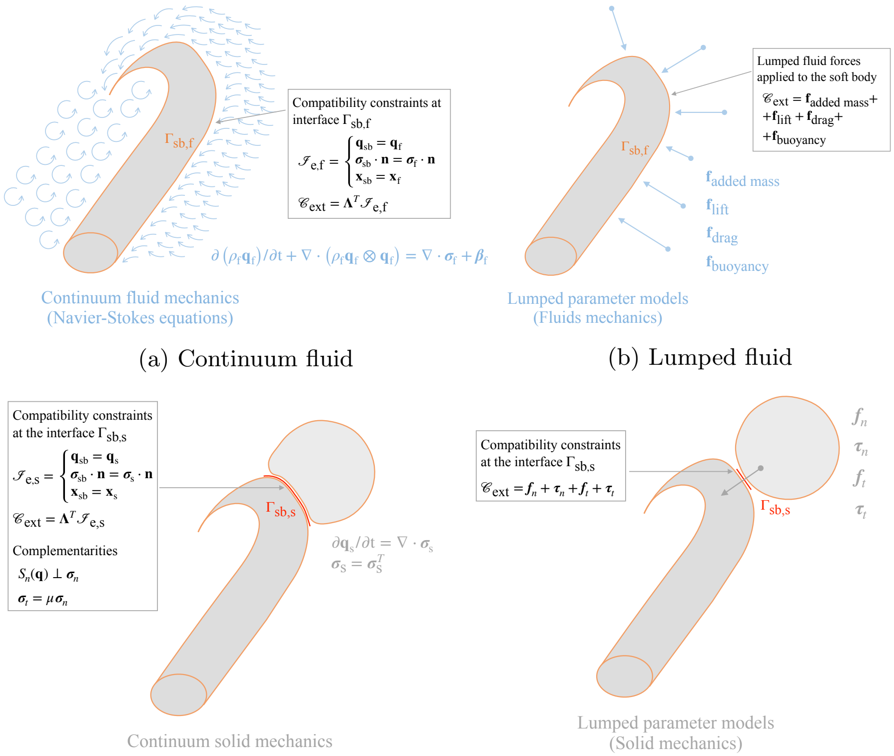
3.3.3 Data-driven and machine learning models

Frictional and normal contact modeling is notoriously difficult, and a high prediction accuracy is onerous to achieve, especially for soft-bodied robots that interact with environments or objects with vastly different surface properties.

Recent efforts explore the augmentation of simulation representations with neural networks to close sim-to-real gaps (see, e.g., [81–84]), reducing the uncertainty that contact with unknown objects or environments introduces.

A naive approach is to learn the mapping of the previous to the next state directly. This would mean that the nonlinear equations $\mathcal{N}$ in equation (1), and also the actuation and constraint force terms, are replaced with a neural network.

However, this is all but practical for complex robots, and it it is best to acquire data for interactions, such as frictional contact, that lead to prediction inaccuracies in our state-of-the-art models, and to only learn corrections. More specifically, we could learn corrections for the frictional forces that act on the soft-body, improving a first-order estimate that we get from a known simulation representation. For augmentation tasks, it is desirable to work with a differentiable simulation representation [85].



(a) Continuum fluid (b) Lumped fluid

(c) Continuum solid (d) Lumped solid

Fig. 4: A conceptual illustration of the models for external interactions described in section 3, applied to the octopus arm of Fig. 2. (a) Continuum fluid mechanics models. (b) Lumped fluid models. (c) Continuum solid mechanics models. (d) Lumped solid models.

4 From models to practice

As emerging from the previous sections, presenting the models involved in describing soft robots and embodied intelligence inside a unified framework is challenging, yet helpful. Therefore, it is not surprising that a structured modeling framework for the physics of interactions in soft robotics remains elusive.

The models presented range from high-dimensional (sections 2.2.1, 2.2.2 for internal interactions, and sections 3.3.1, 3.2.1 for external interactions) to low-dimensional (section 2.2.3 for internal interactions, and sections 3.3.2, 3.2.2 for external interactions). The former have found relatively little use in soft robotics, due to their computational costs, hence limited applicability for control and design purposes. The latter have been instead used with different degrees of success due to their simplifications, that may be unable to capture the rich physics of internal and external interactions.

High-dimensional models can provide a framework for simulation of internal and external interactions. These simulations, in turn, can be used to understand the principles behind an observed and/or desired behaviour and may eventually be adopted for automatic design [78, 79, 86] in the future, if computational power and algorithms will allow for a sufficiently fast model-driven workflow. Yet, having these models permits constructing improved approximated and low-dimensional representations of soft robots, that capture their overall behaviour at a cheaper computational cost, feasible for design and control purposes today. Indeed, the high-dimensional models can capture the key components underpinning embodied intelligence physical principles, thereby allowing for a more accurate low-dimensional description of how soft robots can achieve the embodied intelligence of biological systems. As an example, Fig. 2 depicts a low-dimensional model for octopus' underwater locomotion [87]. A low-dimensional description is typically constituted by a representation of the behaviour of the system in a reduced space, composed by much fewer degrees of freedom than the original high-dimensional counterpart [7, 88]. Therefore, one must identify a suitable low-dimensional representation that retains a sufficient accuracy to describe the whole system.

The discovery of low-dimensional representation (often referred to as fundamental or parsimonious models) is currently done empirically by observing the system. Whether a high-dimensional model is available, the synthesis to the lower-level counterpart (if any) can be obtained more systematically. Accurately informed fundamental models can capture the overall behaviour of the soft robot and expose a few key aggregate physical parameters to describe a specific task (e.g., locomotion, arm movements, etc.). These fundamental models provide a generic base upon which specific morphology/control could be developed, following the template/anchor approach proposed in [89]. The analysis of the fundamental models might increase the comprehension of the system and expose quantitative advantages of compliant robots over rigid ones.

This concept was successfully employed in underwater legged robots, where shape-dependent forces, elastic leg elements, and pushing-based actuation interweave to shape the basin of attraction of the hopping limit cycle. Shape morphing proved how deformable bodies can be exploited to overcome actuation limits [90]. Another remarkable example is presented in [91], where the elastic components of a soft robotic squid have been exploited to increase swimming efficiency. A second-order forced oscillation model was developed to catch the relationship between internal and external forces, mediated by actuation frequency, in an approximate and simplified way. In both cases, they are not a low-level counterpart of the models presented in section 2 and 3, but they are fundamental models which capture the specific behavior of interest, and can provide insights into the design of actual soft robots.

Finally, please note that, for all the three main scopes of internal interactions and external interactions with fluid and solid media, we discussed how machine learning techniques can be used to solve the same modeling problems.

5 Conclusions

We walked through the quest for modelling soft robots with a focus on how to describe the physics involved in their embodied intelligence. We argue that modelling the deformations of a soft robot body under internal actuation forces and external interaction forces can capture the practical essence of embodied intelligence. We described the mathematical models used for describing such internal and external interactions in a soft robot body, including related software tools, and we discussed how to use them in practice. For the first time, we show a unified view of the multiphysics of interactions arising from embodied intelligence, and we link them to the design of soft robots.

Tables 1, and 2 summarize this analysis and provides a practical guidance to the modelling methods that can be beneficially used in soft robotics (not including machine learning techniques), enabling soft robots to fully leverage on embodied intelligence, acquire unprecedented abilities and respond to unmet needs, ultimately contributing to further soft robotics progress.

We argue that, in contrast to the current trial (physically build the robot) and error (robot testing) approach, soft-robot design can transition to a model-informed workflow. Indeed, prototype-driven design (also referred to as trial-and-error) was likely the fastest and most efficient way to proceed, especially in the soft robotics exploratory phase. Today, we are at a stage where computational modelling within a model-driven umbrella can enable (i) scaling up soft-robot design in response to application needs, (ii) holistic model-based control embedding external interactions, and (iii) high-fidelity simulations, opening the path towards soft-robot digital twins.

This transition is within grasp, in an interdisciplinary dialogue of roboticists with communities like computational physics, applied mathematics, scientific computing and machine learning. This unified approach will allow soft robotics to thrive in the next few decades and establish itself as a model-driven scientific discipline that can tangibly impact human activities.

Table 1: Summary of models for soft body mechanics and internal interactions, with reference to equation (1). As discussed in Section 2.2.4, machine learning techniques can be also proposed to solve the same modeling problems.

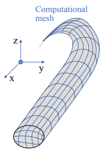
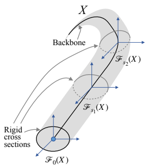
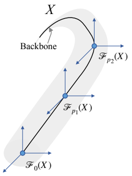
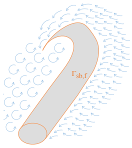
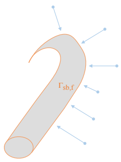
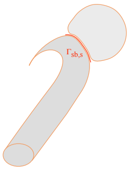
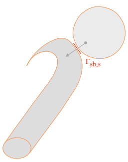
Model	$\mathcal{D}$	$\mathbf{q_{sb}}$	$\mathcal{N}_{\mathbf{sb}}$	$\mathcal{C}_{\mathbf{int}}$	**Software tools**
Continuum 3D solid mechanics  section 2.2.1	$\partial/\partial t$	$\mathbf{v}_{\text{sb}}$	$\nabla \cdot \boldsymbol{\sigma}_{\text{sb}}$	Actuation imposed as equality and inequality constraints: $\mathcal{I}_e, \mathcal{I}_i$, through $\boldsymbol{\Lambda}^T \mathcal{I}_e + \boldsymbol{\Theta}^T \mathcal{I}_i$ (e.g., Lagrange multipliers) [*used with 1D, 2D, 3D actuations*]	*SOFA* [6] *ChainQueen* [8] *Evosoro* [10] *Voxelyze* [9] *ABAQUS* [11] *ANSYS* [12] *COMSOL* [13] *Altair* [14]
Rods and shells, e.g., Cosserat  section 2.2.2	$\partial/\partial t$	$[v, \quad \omega]^T$	$[\mathcal{N}_{\text{sb},v}, \quad \mathcal{N}_{\text{sb},\omega}]^T$ $\mathcal{N}_{\text{sb},v} = 1/(\rho A)(\partial n/\partial x + \bar{n})$ $\mathcal{N}_{\text{sb},\omega} = I^{-1}/\rho[-\omega \times (\rho I \omega) + +\partial c/\partial X + +\partial r/\partial X \times n + \bar{c}]$	Actuation imposed as active internal wrenches $\boldsymbol{\alpha}, \boldsymbol{\tau}$ [*used with 1D, 2D actuation*]	*SOFA* [6] *Elastica* [22] *SoRoSim* [23]
Finite-dimensional parametrization  section 2.2.3	d/dt	$[v \quad \omega]^T$	$-1/\mathbf{M}\,[\mathbf{D}(\mathbf{s}_{\text{sb}}, \mathbf{q}_{\text{sb}}) + +\mathbf{K}(\mathbf{s}_{\text{sb}})]$ $\mathbf{s}_{\text{sb}}$ is the displacement of the soft body	Actuator forces grouped in a vector $\boldsymbol{\alpha}$ that acts as a forcing term on the right-hand side [*used with 1D actuation*]	Bespoke tools written in various languages

Table 2: Summary of models for external interactions, with reference to equation (1). As discussed in sections 3.2.3 and 3.3.3, machine learning techniques can be also proposed to solve the same modeling problems.

Fluid models	$\mathcal{C}_{\text{ext}}$	**Software tools**
Continuum fluid mechanics  section 3.2.1	Compatibility constraints at interface $\Gamma_{\text{sb,f}}$: $\mathcal{I}_{\text{e,f}} = \begin{cases} \mathbf{q}_{\text{sb}} = \mathbf{q}_{\text{f}} \\ \boldsymbol{\sigma}_{\text{sb}} \cdot \mathbf{n} = \boldsymbol{\sigma}_{\text{f}} \cdot \mathbf{n} \\ \mathbf{x}_{\text{sb}} = \mathbf{x}_{\text{f}} \end{cases}$ $\mathcal{C}_{\text{ext}} = \boldsymbol{\Lambda}^T \mathcal{I}_{\text{e,f}}$ (e.g., Lagrange multipliers) Segregated, monolithic approaches	*ANSYS* [12] *COMSOL* [13] *Altair* [14] *Flow3D* [42] *Adina* [43] *OpenFOAM* [44]
Lumped parameters  section 3.2.2	Fluid forces applied on the right-hand side of equation (1) as aggregated contributions $\mathbf{f}_{\text{added mass}} + \mathbf{f}_{\text{drag}} + \mathbf{f}_{\text{lift}} + \mathbf{f}_{\text{buoyancy}}$	Bespoke tools written in various languages
Solid models	$\mathcal{C}_{\text{ext}}$	**Software tools**
Continuum solid mechanics  section 3.3.1	Compatibility constraints at interface $\Gamma_{\text{sb,s}}$: $\mathcal{I}_{\text{e,s}} = \begin{cases} \mathbf{q}_{\text{sb}} = \mathbf{q}_{\text{s}} \\ \boldsymbol{\sigma}_{\text{sb}} \cdot \mathbf{n} = \boldsymbol{\sigma}_{\text{s}} \cdot \mathbf{n} \\ \mathbf{x}_{\text{sb}} = \mathbf{x}_{\text{s}} \end{cases}$ $\mathcal{C}_{\text{ext}} = \boldsymbol{\Lambda}^T \mathcal{I}_{\text{e,s}}$ (e.g., Lagrange multipliers) Complementarities $S_n(\mathbf{q}) \perp \boldsymbol{\sigma}_n$ $\boldsymbol{\sigma}_t = \mu \boldsymbol{\sigma}_n$	*SOFA* [6] *ABAQUS* [11] *ANSYS* [12] *COMSOL* [13]
Lumped parameters  section 3.3.2	Normal and tangential forces and torques applied on the right-hand side of equation (1) as aggregated contributions $\boldsymbol{f}_n, + \boldsymbol{\tau}_n + \boldsymbol{f}_t + \boldsymbol{\tau}_t$	*SOFA* [6] and other bespoke tools written in various languages

Acknowledgments Gianmarco Mengaldo acknowledges NUS support through his start-up grant (R-265-000-A36-133). Gregory Chirikjian acknowledges MOE Tier I grant (R-265-000-655-114) and the NRF Medium Sized Centre CARTIN allocation (R-261-521-002-592). Cecilia Laschi acknowledges NUS support through her start-up grant (R-265-000-A31-133 and R-265-000-A31-731) and the support of the National Research Foundation (NRF), Singapore, under its Medium Sized Centre CARTIN - Centre for Advanced Robotics Technology Innovation (R-261-521-001-592).

References

[1] Kim, S., Laschi, C., Trimmer, B.: Soft robotics: a bioinspired evolution in robotics. Trends in biotechnology **31**(5), 287–294 (2013)

[2] Pfeifer, R., Lungarella, M., Iida, F.: Self-organization, embodiment, and biologically inspired robotics. science **318**(5853), 1088–1093 (2007)

[3] Blickhan, R., Seyfarth, A., Geyer, H., Grimmer, S., Wagner, H., Günther, M.: Intelligence by mechanics. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **365**(1850), 199–220 (2007)

[4] Xavier, M.S., Fleming, A.J., Yong, Y.K.: Finite element modeling of soft fluidic actuators: Overview and recent developments. Advanced Intelligent Systems **3**(2), 2000187 (2021)

[5] Vanneste, F., Goury, O., Duriez, C.: Enabling the control of a new degree of freedom by using anisotropic material on a 6-dof parallel soft robot. In: Robosoft 2021 (2021)

[6] Duriez, C., Bieze, T.: Soft robot modeling, simulation and control in real-time. In: Soft Robotics: Trends, Applications and Challenges, pp. 103–109 (2017)

[7] Goury, O., Duriez, C.: Fast, generic, and reliable control and simulation of soft robots using model order reduction. IEEE Transactions on Robotics **34**(6), 1565–1576 (2018)

[8] Hu, Y., Liu, J., Spielberg, A., Tenenbaum, J.B., Freeman, W.T., Wu, J., Rus, D., Matusik, W.: Chainqueen: A real-time differentiable physical simulator for soft robotics. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 6265–6271 (2019)

[9] Hiller, J., Lipson, H.: Dynamic simulation of soft multimaterial 3d-printed objects. Soft robotics **1**(1), 88–101 (2014)

[10] Cheney, N., MacCurdy, R., Clune, J., Lipson, H.: Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding. ACM SIGEVOlution **7**(1), 11–23 (2014)

[11] ABAQUS software. https://www.3ds.com/products-services/simulia/products/abaqus/. Accessed: 2021-10-27

[12] ANSYS software. https://www.ansys.com. Accessed: 2021-10-27

[13] COMSOL software. https://www.comsol.com. Accessed: 2021-10-27

[14] Altair software. https://www.altair.com. Accessed: 2021-10-27

[15] Turner, M., Peiró, J., Moxey, D.: Curvilinear mesh generation using a variational framework. Computer-Aided Design **103**, 73–91 (2018)

[16] Mengaldo, G., Moxey, D., Turner, M., Moura, R.C., Jassim, A., Taylor, M., Peiro, J., Sherwin, S.J.: Industry-relevant implicit large-eddy simulation of a high-performance road car via spectral/hp element methods. arXiv preprint arXiv:2009.10178 (2020)

[17] Cottrell, J.A., Hughes, T.J.R., Bazilevs, Y.: Isogeometric Analysis: Toward Integration of CAD and FEA

[18] Eringen, A.C.: Microcontinuum Field Theories: I. Foundations and Solids, (2012)

[19] Antman, S.: Nonlinear Problems of Elasticity. Applied Mathematical Sciences. Springer, ??? (2006)

[20] Boyer, F., Lebastard, V., Candelier, F., Renda, F.: Dynamics of continuum and soft robots: A strain parameterization based approach. IEEE Transactions on Robotics **37**(3), 847–863 (2020)

[21] Boyer, F., Renda, F.: Poincaré's equations for cosserat media: Application to shells. Journal of Nonlinear Science **27**(1), 1–44 (2017)

[22] Gazzola, M., Dudte, L., McCormick, A., Mahadevan, L.: Forward and inverse problems in the mechanics of soft filaments. Royal Society open science **5**(6), 171628 (2018). https://doi.org/10.1098/rsos.171628

[23] Mathew, A.T., Hmida, I.B., Armanini, C., Boyer, F., Renda, F.: Sorosim: a matlab toolbox for soft robotics based on the geometric variable-strain approach. arXiv preprint arXiv:2107.05494 (2021)

[24] Adagolodjo, Y., Renda, F., Duriez, C.: Coupling numerical deformable models in global and reduced coordinates for the simulation of the direct

and the inverse kinematics of soft robots. IEEE Robotics and Automation Letters **6**(2), 3910–3917 (2021). https://doi.org/10.1109/LRA.2021.3061977

[25] Fu, Q., Gart, S.W., Mitchel, T.W., Kim, J.S., Chirikjian, G.S., Li, C.: Lateral oscillation and body compliance help snakes and snake robots stably traverse large, smooth obstacles. Integrative and comparative biology **60**(1), 171–179 (2020)

[26] Chirikjian, G.S., Burdick, J.W.: A modal approach to hyper-redundant manipulator kinematics. IEEE Transactions on Robotics and Automation **10**(3), 343–354 (1994)

[27] Kim, B., Ha, J., Park, F.C., Dupont, P.E.: Optimizing curvature sensor placement for fast, accurate shape sensing of continuum robots. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 5374–5379 (2014). IEEE

[28] Suzumori, K., Iikura, S., Tanaka, H.: Development of flexible microactuator and its applications to robotic mechanisms. In: Proceedings. 1991 IEEE International Conference on Robotics and Automation, pp. 1622–1623 (1991). IEEE Computer Society

[29] Webster III, R.J., Jones, B.A.: Design and kinematic modeling of constant curvature continuum robots: A review. The International Journal of Robotics Research **29**(13), 1661–1683 (2010)

[30] George Thuruthel, T., Ansari, Y., Falotico, E., Laschi, C.: Control strategies for soft robotic manipulators: A survey. Soft robotics **5**(2), 149–163 (2018)

[31] Kim, D., Kim, S.-H., Kim, T., Kang, B.B., Lee, M., Park, W., Ku, S., Kim, D., Kwon, J., Lee, H., *et al.*: Review of machine learning methods in soft robotics. Plos one **16**(2), 0246102 (2021)

[32] Wang, X., Li, Y., Kwok, K.-W.: A survey for machine learning-based control of continuum robots. Frontiers in Robotics and AI, 280 (2021)

[33] Giorelli, M., Renda, F., Calisti, M., Arienti, A., Ferri, G., Laschi, C.: Neural network and jacobian method for solving the inverse statics of a cable-driven soft arm with nonconstant curvature. IEEE Transactions on Robotics **31**(4), 823–834 (2015)

[34] Dowell, E.H., Hall, K.C.: Modeling of fluid-structure interaction. Annual review of fluid mechanics **33**(1), 445–490 (2001)

[35] Hou, G., Wang, J., Layton, A.: Numerical methods for fluid-structure interaction—a review. Communications in Computational Physics **12**(2), 337–377 (2012)

[36] Souli, M., Benson, D.J.: Arbitrary Lagrangian Eulerian and Fluid-structure Interaction: Numerical Simulation. John Wiley & Sons, ??? (2013)

[37] Mittal, R., Iaccarino, G.: Immersed boundary methods. Annu. Rev. Fluid Mech. **37**, 239–261 (2005)

[38] Taira, K., Colonius, T.: The immersed boundary method: a projection approach. Journal of Computational Physics **225**(2), 2118–2137 (2007)

[39] Goza, A., Colonius, T.: A strongly-coupled immersed-boundary formulation for thin elastic structures. Journal of Computational Physics **336**,

401–411 (2017)

[40] Dickinson, M.H., Farley, C.T., Full, R.J., Koehl, M., Kram, R., Lehman, S.: How animals move: an integrative view. science **288**(5463), 100–106 (2000)

[41] Lauder, G.V.: Fish locomotion: recent advances and new directions. Annual review of marine science **7**, 521–545 (2015)

[42] FLOW3D software. https://www.flow3d.com. Accessed: 2021-10-27

[43] Adina software. http://www.adina.com. Accessed: 2021-10-27

[44] Weller, H.G., Tabor, G., Jasak, H., Fureby, C.: A tensorial approach to computational continuum mechanics using object-oriented techniques. Computers in physics **12**(6), 620–631 (1998)

[45] Renda, F., Giorgio-Serchi, F., Boyer, F., Laschi, C., Dias, J., Seneviratne, L.: A unified multi-soft-body dynamic model for underwater soft robots. The International Journal of Robotics Research **37**(6), 648–666 (2018). https://doi.org/10.1177/0278364918769992

[46] Duraisamy, K., Iaccarino, G., Xiao, H.: Turbulence modeling in the age of data. Annual Reviews of Fluid Mechanics **51**, 357–377 (2019)

[47] Brunton, S.L., Noack, B.R., Koumoutsakos, P.: Machine learning for fluid mechanics. Annual Review of Fluid Mechanics **52**, 477–508 (2020)

[48] Vinuesa, R., Brunton, S.L.: The potential of machine learning to enhance computational fluid dynamics. arXiv preprint arXiv:2110.02085 (2021)

[49] Bar-Sinai, Y., Hoyer, S., Hickey, J., Brenner, M.P.: Learning data-driven discretizations for partial differential equations. Proceedings of the National Academy of Sciences **116**(31), 15344–15349 (2019)

[50] Kochkov, D., Smith, J.A., Alieva, A., Wang, Q., Brenner, M.P., Hoyer, S.: Machine learning accelerated computational fluid dynamics. arXiv preprint arXiv:2102.01010 (2021)

[51] Wang, R., Walters, R., Yu, R.: Incorporating symmetry into deep dynamics models for improved generalization. arXiv preprint arXiv:2002.03061 (2020)

[52] Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., Anandkumar, A.: Fourier neural operator for parametric partial differential equations. arXiv preprint arXiv:2010.08895 (2020)

[53] Ling, J., Kurzawski, A., Templeton, J.: Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. Journal of Fluid Mechanics **807**, 155–166 (2016)

[54] Maulik, R., San, O., Rasheed, A., Vedula, P.: Subgrid modelling for two-dimensional turbulence using neural networks. Journal of Fluid Mechanics **858**, 122–144 (2019)

[55] Novati, G., de Laroussilhe, H.L., Koumoutsakos, P.: Automating turbulence modelling by multi-agent reinforcement learning. Nature Machine Intelligence **3**(1), 87–96 (2021)

[56] Beetham, S., Capecelatro, J.: Formulating turbulence closures using sparse regression with embedded form invariance. Physical Review Fluids **5**(8), 084611 (2020)

[57] Beetham, S., Fox, R.O., Capecelatro, J.: Sparse identification of multiphase turbulence closures for coupled fluid–particle flows. Journal of Fluid Mechanics **914** (2021)

[58] Taira, K., Brunton, S.L., Dawson, S., Rowley, C.W., Colonius, T., McKeon, B.J., Schmidt, O.T., Gordeyev, S., Theofilis, V., Ukeiley, L.S.: Modal analysis of fluid flows: An overview. AIAA Journal **55**(12), 4013–4041 (2017)

[59] Brunton, S.L., Proctor, J.L., Kutz, J.N.: Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proceedings of the National Academy of Sciences **113**(15), 3932–3937 (2016)

[60] Loiseau, J.-C., Brunton, S.L.: Constrained sparse Galerkin regression. Journal of Fluid Mechanics **838**, 42–67 (2018)

[61] Loiseau, J.-C., Noack, B.R., Brunton, S.L.: Sparse reduced-order modeling: sensor-based dynamics to full-state estimation. Journal of Fluid Mechanics **844**, 459–490 (2018)

[62] Deng, N., Noack, B.R., Morzynski, M., Pastur, L.R.: Low-order model for successive bifurcations of the fluidic pinball. Journal of fluid mechanics **884**(A37) (2020)

[63] Deng, N., Noack, B.R., Morzyński, M., Pastur, L.R.: Galerkin force model for transient and post-transient dynamics of the fluidic pinball. Journal of Fluid Mechanics **918** (2021)

[64] Lee, K., Carlberg, K.T.: Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. Journal of

Computational Physics **404**, 108973 (2020)

[65] Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics **378**, 686–707 (2019)

[66] Lu, L., Jin, P., Pang, G., Zhang, Z., Karniadakis, G.E.: Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. Nature Machine Intelligence **3**(3), 218–229 (2021)

[67] Johnson, K.L., Johnson, K.L.: Contact Mechanics. Cambridge university press, ??? (1987)

[68] Vakis, A.I., Yastrebov, V.A., Scheibert, J., Nicola, L., Dini, D., Minfray, C., Almqvist, A., Paggi, M., Lee, S., Limbert, G., *et al.*: Modeling and simulation in tribology across scales: An overview. Tribology International **125**, 169–199 (2018)

[69] Studer, C., Glocker, C.: Simulation of non-smooth mechanical systems with many unilateral constraints. Proceedings ENOC-2005 Eindhoven (2005)

[70] Acary, V.: Projected event-capturing time-stepping schemes for nonsmooth mechanical systems with unilateral contact and coulomb's friction. Computer Methods in Applied Mechanics and Engineering **256**, 224–250 (2013)

[71] Coevoet, E., Escande, A., Duriez, C.: Optimization-based inverse model of soft robots with contact handling. IEEE Robotics and Automation Letters **2**(3), 1413–1419 (2017)

[72] Collins, J., Chand, S., Vanderkop, A., Howard, D.: A review of physics simulators for robotic applications. IEEE Access (2021)

[73] sofa framework. https://www.sofa-framework.org. Accessed: 2021-11-22

[74] Xu, J., Aykut, T., Ma, D., Steinbach, E.: 6dls: Modeling nonplanar frictional surface contacts for grasping using 6-d limit surfaces. IEEE Transactions on Robotics (2021)

[75] Xydas, N., Kao, I.: Modeling of contact mechanics and friction limit surfaces for soft fingers in robotics, with experimental results. The International Journal of Robotics Research **18**(9), 941–950 (1999)

[76] Majidi, C., Shepherd, R.F., Kramer, R.K., Whitesides, G.M., Wood, R.J.: Influence of surface traction on soft robot undulation. The International Journal of Robotics Research **32**(13), 1577–1584 (2013)

[77] Deimel, R., Brock, O.: A novel type of compliant and underactuated robotic hand for dexterous grasping. The International Journal of Robotics Research **35**(1-3), 161–185 (2016)

[78] Lipson, H., Pollack, J.B.: Automatic design and manufacture of robotic lifeforms. Nature **406**(6799), 974–978 (2000)

[79] Hiller, J., Lipson, H.: Automatic design and manufacture of soft robots. IEEE Transactions on Robotics **28**(2), 457–466 (2011)

[80] Coevoet, E., Escande, A., Duriez, C.: Soft robots locomotion and manipulation control using fem simulation and quadratic programming. In: 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft), pp. 739–745 (2019)

[81] Hwangbo, J., Lee, J., Dosovitskiy, A., Bellicoso, D., Tsounis, V., Koltun, V., Hutter, M.: Learning agile and dynamic motor skills for legged robots. Science Robotics **4**(26) (2019)

[82] Golemo, F., Taiga, A.A., Courville, A., Oudeyer, P.-Y.: Sim-to-real transfer with neural-augmented robot simulation. In: Conference on Robot Learning, pp. 817–828 (2018). PMLR

[83] Battaglia, P., Pascanu, R., Lai, M., Jimenez Rezende, D., et al.: Interaction networks for learning about objects, relations and physics. Advances in neural information processing systems **29** (2016)

[84] Jiang, Y., Liu, K.: Data-augmented contact model for rigid body simulation (2018)

[85] Bächer, M., Knoop, E., Schumacher, C.: Design and control of soft robots using differentiable simulation. Current Robotics Reports **2**(2), 211–221 (2021). https://doi.org/10.1007/s43154-021-00052-7

[86] Lipson, H.: Challenges and opportunities for design, simulation, and fabrication of soft robots. Soft Robotics **1**(1), 21–27 (2014)

[87] Calisti, M., Laschi, C.: Morphological and control criteria for self-stable underwater hopping. Bioinspiration & biomimetics **13**(1), 016001 (2017)

[88] Chenevier, J., González, D., Aguado, J.V., Chinesta, F., Cueto, E.: Reduced-order modeling of soft robots. PloS one **13**(2), 0192052 (2018)

[89] Full, R.J., Koditschek, D.E.: Templates and anchors: neuromechanical hypotheses of legged locomotion on land. Journal of experimental biology **202**(23), 3325–3332 (1999)

[90] Picardi, G., Chellapurath, M., Iacoponi, S., Stefanni, S., Laschi, C., Calisti, M.: Bioinspired underwater legged robot for seabed exploration with low environmental disturbance. Science Robotics **5**(42) (2020)

[91] Bujard, T., Giorgio-Serchi, F., Weymouth, G.D.: A resonant squid-inspired robot unlocks biological propulsive efficiency. Science Robotics **6**(50), 2971 (2021). https://doi.org/10.1126/scirobotics.abd2971