Noname manuscript No.
(will be inserted by the editor)

Design and Control of Soft Robots using Differentiable Simulation

Moritz Bächer · Espen Knoop · Christian Schumacher

Accepted: 18 March 2021 / Published online: 10 May 2021

Abstract *Purpose of Review* We discuss the use of differentiable simulation for computational problems in soft robotics. This includes characterizing the mechanical behavior of soft robots, optimally controlling embedded soft actuators or active materials, and estimating the robot's state from readings of embedded sensors. Moreover, we discuss how design optimization can help to optimally place soft actuators and sensors.
Recent Findings We expatiate on the adoption of simulation and optimization tools in the process of designing and controlling soft robots. We include a discussion of rigid-flexible systems and the use of differentiable simulation in combination with machine learning.
Summary We review the state of the art in the computational modeling of soft robots and provide a summary of the required mathematical tools. We also review several open questions where computation could help to move the field forward, and discuss the role of differentiable simulation in managing the ever-growing design complexity of next-generation soft robots.
Keywords Differentiable Simulation; Computational Design and Control; State Estimation; Sensor and Actuator Design

1 Introduction

Soft robots hold the promise of enabling the manipulation of fragile objects of vastly different shapes [1], the inherently safe interaction with humans in complex co-working environments, and the execution of intricate tasks in difficult-to-access areas or unstructured terrain [2]. We attribute these desirable properties to the capability of a *single* soft-bodied system to adopt to *many* drastically different environments.

To date, the field has primarily been driven by experimental research and prototype-based development. However, as the field matures, it will become increasingly important to leverage simulation and optimization tools in the design and control of soft-bodied robots.

In contrast to a recent review on design optimization of soft robots [3], we restrict our expedition to techniques that rely on *differentiable simulation* (refer to Fig. 1), and review computational challenges when designing soft systems made of active and passive materials, equipped with various types of soft actuators and sensors [4, 5].

Differentiable simulation provides a principled mathematical framework to (1) solve complex *characterization problems* to detect and close application-specific sim-to-real gaps, (2) optimally control embedded soft actuators for grasping or locomotion tasks (*actuation problem*), and (3) estimate the mechanical state of the soft system from a set of embedded sensors. While the former improves the prediction accuracy of simulations, the latter two enable optimal open- and closed-loop control of manually designed soft robots.

Moreover, building on top of this, we can also use optimization to *design* robots with optimally placed sensors and actuators. For example, with appropriate objective functions, we can solve for a desired grasping, locomotion, or deformation behavior under the constraint that we solve the actuation and state estimation problems to first-order optimality.

In several subfields, the soft robotics community has adopted simulation as a design tool. However, to scale soft robotic technologies along complexity axes spanned by shape, material, sensing, and actuation parameters, and ultimately build robots that perfect the imitation of biological systems or ex-

Disney Research
Stampfenbachstrasse 48
8032 Zurich
Switzerland
E-mail: moritz.baecher@disney.com

Moritz Bächer is the correspondence author.

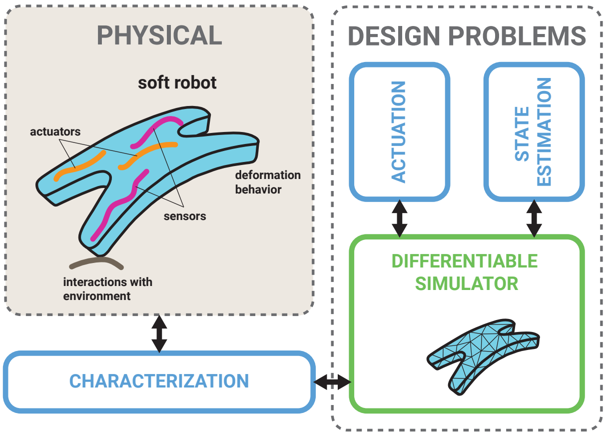

Fig. 1 In this survey we are concerned with the general setting of a soft-bodied robot (left), and a differentiable simulation representation of the same robot (right). We discuss problems of *characterization*, where we identify model parameters that minimize the discrepancy between the simulated and real robot (sim-to-real gap), *actuation*, where we optimally control a set of embedded actuators in order to achieve some desired deformation behavior, and also *state estimation*, where we are concerned with estimating the state of the robot given a set of sensory measurements. In *design* optimizations, we can then ask for optimal actuator and sensor placement for a specific task.

cel in a wide variety of highly complex tasks [6–8], simulation is an important building block but insufficient if used in isolation. To make soft system design scalable, several abstraction layers above a simulation are required, in particular due to the intimate and complex coupling between forces and continuous deformation. We believe that differentiable simulation will play a key role in achieving this ambitious vision, as the differentiability property enables the formulation of gradient-based optimizations, and therefore the introduction of clean abstraction layers. Moreover, in combination with learning, differentiable simulation enables the end-to-end training of loss functions which depend on the simulation state of soft robots.

In this survey, we will first review differentiable quasi-static and dynamic simulation and their use in optimization, providing the reader with specific examples. In a second part, we will review where we stand as a community in the adoption of computational tools summarized in Fig. 1 right. Important subproblems such as soft contact and flexible-rigid systems are discussed as well.

The use of differentiable simulation in soft robotics is somewhat scarce, as the topic is still relatively new. It is our hope that a high-level summary of mathematical tools required for approaching complex control and design tasks will help in this regard.

2 Mathematical Tools for Optimal Design and Control

The first step towards a digitization of the design of general, autonomous soft robots is the use of simulation. Because soft robots undergo large deformations under actuation and contact forces, hyperelastic models with nonlinearities in both the strain and stress measures are often necessary. While hyperelastic materials are well understood [9], soft sensor models, which map deformations to simulated readings, and actuator models, which map actuation parameters to deformations, are an active area of research.

Characterization Problem In the process of developing accurate simulation models, it is key to be able to quantify if a simulator provides us with the desired prediction accuracy, and to detect where it fails to model the underlying physical behavior. To this end, a natural approach is to compare the simulated behavior to captured data, under matching actuation and external forces (see Fig. 2 top). This characterization can either be performed on material specimens [10] and individual sensors and actuators [11], or on the full robot [12].

The important question is then how to find *optimal* values for the set of material parameters $\mathbf{p}_{\text{mat}}$ that best explain the captured behavior (e.g., by comparing the simulated to the observed behavior at a few marker locations on the surface of the robot). If the prediction accuracy is insufficient for the task at hand, the optimal characterization can help us with the identification of inaccuracies in the simulation model (*sim-to-real gap*).

Actuation Problem A wide range of methods for actuating soft robots have been explored, including active materials, artificial muscle actuators, tendons, and pneumatic actuation [5], and new methods are actively being explored by the soft robotics community. While some simulation models exist, modeling is at an embryonic stage for recently developed actuators. Moreover, the coupled behavior of actuators when embedded in passive materials often requires modeling refinements.

Assuming a sufficiently accurate simulation of the coupled behavior, an important question is how to solve for optimal actuation parameters $\mathbf{p}_{\text{act}}$ to achieve a desired deformation behavior (e.g., to optimally grasp a class of objects). The objective for this optimal control problem (Fig. 2 middle) is therefore the difference between the simulation state of the robot and a single or a set of target states, taking contact forces and gravity into account [11].

State Estimation Problem Many biological systems are capable of sensing their proprioceptive state [13]. This is made possible with receptors that detect and measure deformations of muscle fibers. While the hardware and manufacturing of entirely soft, autonomous robots is still in its infancy [8], control of soft systems requires a sufficiently accurate estimation of the state the robot is in, from readings of a set of embedded sensors.

Simulations typically take external contact forces, body forces such as gravity, and actuation parameters as input, and output a deformed configuration of the robot under these forces. In state estimation, external contact forces are assumed to be unknown, and one seeks to estimate these forces such that the simulated sensor readings, that depend on the resulting deformed configuration, explain the measured readings well [14]. Referring to these external forces as parameters $\mathbf{p}_{sta}$, we therefore seek to find optimal values such that the deformed configuration of the robot leads to as small differences between simulated and observed readings as possible (Fig. 2 bottom). Analogously to the detection of sim-to-real gaps when solving for optimal material parameters, optimal state estimation is useful to decide if a set of sensors is sufficient to estimate the state of the robot for a targeted deformation space and interactions with the environment.

Design Problem A soft robot made of passive materials, and actuated through external interactions, has a rest shape with an assigned material at every point, displaying a locally or globally homogenous or inhomogenous; isotropic or anisotropic behavior. Both the rest shape and the material assignment influence the deformation behavior of the robot. Representing these shape and material parameters with a parameter vector $\mathbf{p}_{des}$, we seek to solve for their optimal values to, for example, maximize the contact surface or forces between the soft robot and objects of varying shape [15].

A more challenging task is to optimize the placement of sensors and actuators, to minimize the target matching or state estimation error over a user-specified deformation space, represented, for example, with a discrete set of sample poses [11, 14]. Before discussing how we can approach this latter class of problems, where a simulation, a sensing or actuation, and a design problem are *nested*, we will review mathematical tools to solve optimization problems where the objectives depend on the quasi-static or dynamic state of the robot. Because the same mathematical tools apply to all of the above problems, we will work with a general parameter vector $\mathbf{p}$.

2.1 Differentiable Simulation

Quasi-Static Problems For slowly-moving robots, a quasi-static modeling, where inertial forces are ignored, is often sufficient. A general soft robot may consist of one-dimensional entities such as artificial muscle fibers or stretch sensors, two-dimensional entities such as layers of active materials, and three-dimensional entities such as passive materials surrounding other entities. To simulate a soft robot, we first discretize these entities using, for example, the finite element method (FEM). Representing the deformed state with a set of discrete degrees of freedom $\mathbf{x}$, we then solve the nonlinear force equations [9]

$$\mathbf{f}(\mathbf{p}, \mathbf{x}(\mathbf{p})) = \mathbf{0} \tag{1}$$

to equilibrium, at which the internal forces balance externally applied forces. Note that, whenever we vary the parameters $\mathbf{p}$, the deformed configuration at which the forces sum up to zero at every point within the soft body, changes. $\mathbf{x}$ is therefore implicitly dependent on $\mathbf{p}$.

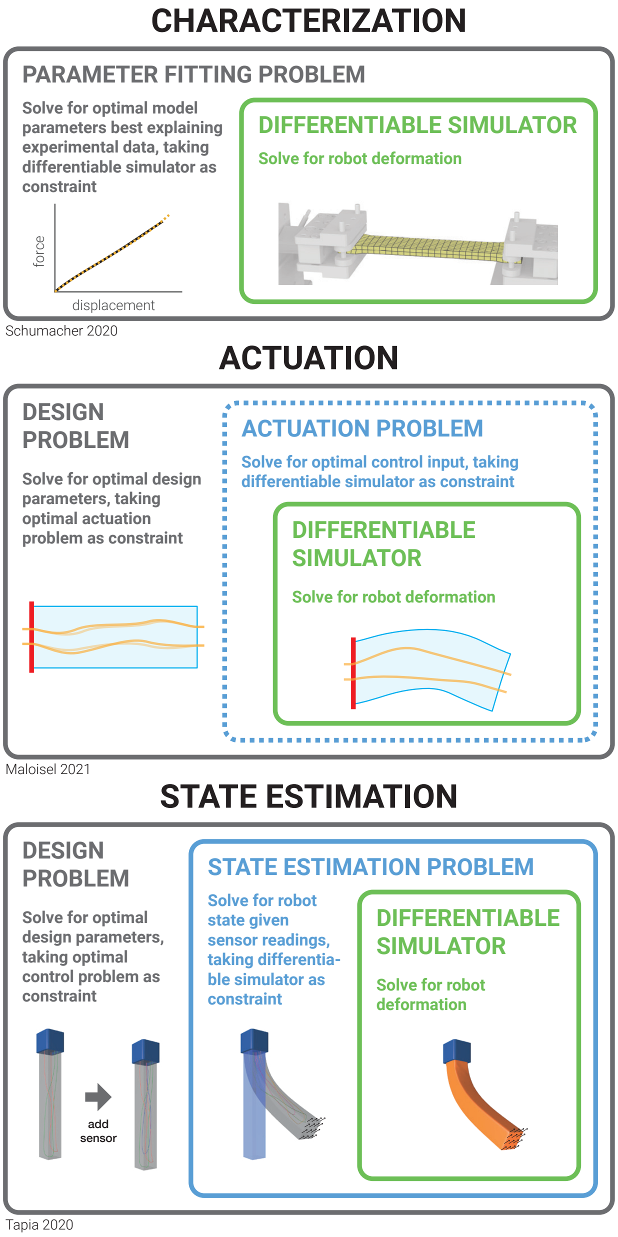

Fig. 2 Nested view of Characterization, Actuation and State Estimation problems. The illustrations for the three problems have been taken from [10], [11], and [14] — these papers exemplify the steps outlined here and we refer to the original papers for in-depth discussions.

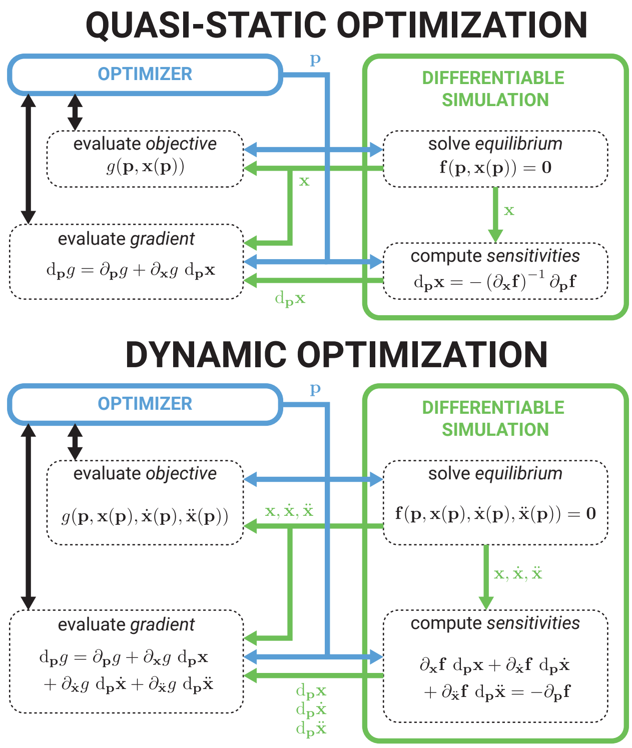

Fig. 3 To minimize objectives that depend on the simulation state of the robot, we first solve for the equilibrium state and, if needed, for the sensitivities of the state with respect to the parameters before we evaluate the objective or its gradient.

For characterization, sensing, actuation, or rest shape optimzation, one typically has a target state $\bar{\mathbf{x}}$ given, and seeks to find the optimal set of parameters such that the difference between the simulated and target state, namely $\frac{1}{2}\|\mathbf{x}(\mathbf{p})-\bar{\mathbf{x}}\|^2$ or similar, is minimal. Using g as a placeholder for an arbitrary objective that depends on the deformed state of the robot, we seek to solve the minimization problem

$$\min_{\mathbf{p}} g(\mathbf{p},\mathbf{x}(\mathbf{p})) \quad \text{s.t. } \mathbf{f}(\mathbf{p},\mathbf{x}(\mathbf{p}))=\mathbf{0}, \tag{2}$$

constraining the deformed configuration to be an equilibrium.

To solve this problem, one common approach is to apply the implicit function theorem or so-called sensitivity analysis: in an infinitesimal neighborhood of $\mathbf{p}$, the derivative of the equilibrium equations "remains" zero, enabling us to *compute* the analytical derivative of the deformed configuration with respect to the optimization parameters[1]

$$\partial_{\mathbf{p}}\mathbf{f}+\partial_{\mathbf{x}}\mathbf{f}\ \mathrm{d}_{\mathbf{p}}\mathbf{x}=\mathbf{0} \implies \mathrm{d}_{\mathbf{p}}\mathbf{x}=-\left(\partial_{\mathbf{x}}\mathbf{f}\right)^{-1}\partial_{\mathbf{p}}\mathbf{f}. \tag{3}$$

As we summarize in Fig. 3 top, standard unconstrained optimization can be used to solve this problem (e.g., a quasi-Newton algorithm [16]). While the derivative of the forces with respect to the deformed configuration, $\partial_{\mathbf{x}}\mathbf{f}$, is the standard stiffness matrix, the derivative of the forces with respect to parameters is usually *not* available in standard simulation packages. For efficiency, the so-called adjoint method is often used in implementations.

[1] We use $\mathrm{d}_{\mathbf{a}}\mathbf{b}$ and $\partial_{\mathbf{a}}\mathbf{b}$ to denote the total and partial derivatives of a vector-valued function $\mathbf{b}$ with respect to a parameter vector $\mathbf{a}$. We rely on the numerator layout in our derivations.

Dynamics Problems For very soft materials or fast motions, dynamic modeling is needed. For dynamic systems [9], the force equilibrium is time dependent and includes additional inertia and damping terms that depend on the accelerations $\ddot{\mathbf{x}}$ and velocities $\dot{\mathbf{x}}$ of the degrees of freedom $\mathbf{x}$

$$\mathbf{f}(\mathbf{p},\dot{\mathbf{x}}(\mathbf{p}),\dot{\mathbf{x}}(\mathbf{p}),\ddot{\mathbf{x}}(\mathbf{p}))=\mathbf{0}. \tag{4}$$

Analogously to the quasi-static problem, we can ask a soft robot's state $\mathbf{x}$ at time t to be as close as possible to a target state $\bar{\mathbf{x}}$, and integrate these differences to form an objective g. The generic form of the dynamics problem is therefore

$$\min_{\mathbf{p}} g(\mathbf{p},\mathbf{x}(\mathbf{p}),\dot{\mathbf{x}}(\mathbf{p}),\ddot{\mathbf{x}}(\mathbf{p})) \text{ s.t. } \mathbf{f}(\mathbf{p},\mathbf{x}(\mathbf{p}),\dot{\mathbf{x}}(\mathbf{p}),\ddot{\mathbf{x}}(\mathbf{p}))=\mathbf{0}. \tag{5}$$

A strategy to solve this problem emerges when we apply the implicit function theorem to the time-varying equilibrium equations [17]

$$\partial_{\mathbf{p}}\mathbf{f}+\partial_{\mathbf{x}}\mathbf{f}\ \mathrm{d}_{\mathbf{p}}\mathbf{x}+\partial_{\dot{\mathbf{x}}}\mathbf{f}\ \mathrm{d}_{\mathbf{p}}\dot{\mathbf{x}}+\partial_{\ddot{\mathbf{x}}}\mathbf{f}\ \mathrm{d}_{\mathbf{p}}\ddot{\mathbf{x}}=\mathbf{0}. \tag{6}$$

We proceed analogously to the quasi-static case (Fig. 3 bottom): Whenever we evaluate the objective g or its gradient after updating the set of parameters, we first solve for the state of the robot, represented by $\mathbf{x}$ and its time derivatives $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$, by integrating the dynamics system (Eq. 4) forward in time. For gradient evaluations, we solve, in addition, the derivative of the motion equations (Eq. 6) for the unknown sensitivities $\mathrm{d}_{\mathbf{p}}\mathbf{x}$, $\mathrm{d}_{\mathbf{p}}\dot{\mathbf{x}}$, and $\mathrm{d}_{\mathbf{p}}\ddot{\mathbf{x}}$. Note that both systems of equations are ordinary differential equations (ODEs), requiring initial values. Especially for problems with high parameter counts, the discrete or continuous adjoint method is commonly used [17, 18]. Otherwise, the problems quickly become intractable.

2.2 Nesting through First-Order Optimality Constraints

When solving sensor and actuator design problems, it is most natural to measure the performance of a candidate design by evaluating the state estimation and actuation problem for samples taken from a deformation space. If the sensing or target matching error is high, we rank the design lower than if the error is small. Nesting through first-order optimality constraints can help us to formalize and solve this class of problems in a principled manner.

For the sake of concreteness, we use the state estimation problem to develop the general formulation. Specifically, we consider the problem of routing a set of one-dimensional stretch sensors, parameterized with a vector $\mathbf{p}$, through a soft body such that the differences between simulated and measured sensor readings is minimized for a user-specified target deformation (a variant of the problems addressed in [11,14]).

In the sensing or state estimation problem, we solve for the unknown interaction parameters $\mathbf{q}$, measuring differences between the simulated and observed sensor readings with an objective g. In the design problem, we are then interested in how well the deformed configuration $\mathbf{x}$, that results from solving the sensing problem to first-order optimality, matches the user-specified target. Measuring this target matching error with an objective h, we can solve this problem by introducing a second abstraction layer or nesting

$$\min_{\mathbf{p}} \; h(\mathbf{p},\mathbf{q}(\mathbf{p}),\mathbf{x}(\mathbf{p},\mathbf{q}(\mathbf{p}))) \;\text{ s.t. }\; \mathrm{d}_{\mathbf{q}}g(\mathbf{p},\mathbf{q}(\mathbf{p}),\mathbf{x}(\mathbf{p},\mathbf{q}(\mathbf{p}))) = \mathbf{0}$$
$$\text{and }\; \mathbf{f}(\mathbf{p},\mathbf{q}(\mathbf{p}),\mathbf{x}(\mathbf{p},\mathbf{q}(\mathbf{p}))) = \mathbf{0} \tag{7}$$

where the constraint $\mathrm{d}_{\mathbf{q}}g = \mathbf{0}$ ensures that the state estimation problem is solved to first-order optimality. This nested view, here formulated for the quasi-static case, enables the computation of an analytical gradient for the design problem by applying the implicit function theorem to the first-order optimality constraint, proceeding analogously to the equilibrium-constrained case, or first nesting layer, otherwise (compare with Fig. 3). For detailed derivations, we point the interested reader to [14].

2.3 Differentiability of Simulation Representations

To be able to solve design and control problems with numerical optimization, the simulation representation needs to be differentiable with respect to the optimization parameters. This is in general a non-trivial task. For example, for rest shape optimization, the topology of the simulation mesh is changing over time (remeshing), constituting a discontinuous operation. Meshfree methods [19] can remedy this problem. Alternatively, the soft robot can be embedded in a simulation grid which remains constant throughout optimizations [20], enabling soft robot design optimizations directly on CAD representations. Strategies to decouple the simulation accuracy from the quality of the simulation mesh provide another alternative [21].

Embedded actuators or sensors can lead to another source of discontinuities. Due to the discontinuity of the deformation gradient at element boundaries for various standard finite elements (e.g., ones that reply on Lagrange shape functions), the strain field is not differentiable everywhere within the volume enclosed by the soft robot. Especially for the placement optimization of one- or two-dimensional entities (e.g., actuators or sensors), this constitutes a challenge. Maloisel et al. [11] propose to use moving-least-squares (MLS) shape functions to make the simulation differentiable with respect to spline parameters that define the routing of artificial muscle actuators. Similar strategies could help to make simulation representations of general soft robots differentiable.

A third source of discontinuities is frictional contact. Contact is a discrete event. However, especially for soft bodied systems, contact forces gradually increase, and smooth or soft contact models can help to mitigate or avoid these type of discontinuity [22,23], without introducing significant sim-to-real gaps.

2.4 Rigid-Flexible Systems

So far we have discussed the modeling of entirely soft robots. However, for many industrial applications, soft robotic components will augment traditional systems, but will not fully replace them. It is therefore important to be able to formulate design and control optimizations for robots consisting of rigid *and* flexible components, coupled to one another with constraints. Flexible multibody simulation is a well-studied problem. However, its differentiability has received less attention. We point the interested reader to design and control approaches (quasi-statics [24]; dynamics [12,25]) for examples on how to expand the mathematical tools for soft bodied models (Sec. 2.1) to handle systems with rigid bodies and coupling constraints between flexible and rigid components, or other constraint types.

2.5 Differentiable Simulation and Learning

We can think of a simulator as a highly complex function, taking forces at a discrete set of locations as input, returning a deformed configuration as output. There are two incentives to replace this function at least partially with a neural network: (1) to augment the simulation model where the sim-to-real gap is high and more accurate models are unknown (e.g., for frictional contact [26]), and (2) to reduce the time complexity of simulations to either speed up simulation-driven design and control [27,28], or to enable the interactive exploration of soft robot designs [29]. There is also work on completely replacing a simulator with a neural network [30], learned from either real or simulated data. While providing an exciting avenue for future research, a complete replacement of the simulator comes at the cost of longer training and requiring more data. A disadvantage of neural simulation is also that simulators become difficult-to-interpret blackboxes.

Another interesting use case of differentiable simulation is in the training of loss functions that depend on the state of the robot (see, e.g., Geilinger et al. [22]). Due to the differentiability property, we can consider the simulation as a node without weights, and directly use it in backpropagation.

2.6 Alternative Optimization Techniques

For rest shape or material distribution optimization, topology optimization has seen widespread use [31–36]. We can interpret traditional topology optimization as an application of differentiable simulation. However, because a linear elastic behavior is often assumed, it is, in its simplest form, not an ideal tool for soft robotics applications. To formulate shape optimizations, combinations of eXtended Finite Elements (XFE), level-set approaches that enable the continuous movement of material-material or material-void boundaries through elements, and analytical sensitivity analysis has seen use [15,20,37,38]. This is again a variant of differentiable simulation that enables the optimization of soft robots consisting of composite materials [39]. Shape derivative techniques are also common [40].

Despite its utility, differentiable simulation has also disadvantages. It can take considerable time and effort to make existing simulators differentiable, or to implement them from scratch. Moreover, for very complex problems, the differentiability property may be near-impossible to achieve, and one has to resort to derivative-free optimization. While evolutionary algorithms and their siblings require more iterations, they can help to navigate complex design spaces [41] with many local minima and discontinuities [42,43].

3 State of the Art

This section summarizes existing work that has been carried out within the domain of simulation-driven control and design. We refer to Tab. 1 for a concise summary and categorization of the related works. The first column (*problem*) categorizes work based on whether they address the *characterization*, *actuation*, or *state estimation* problem (see also Fig. 2). Next, we list whether the problem *setting* is *quasi-static* or takes into account *dynamics*, and if *contact* is modeled. We also list the *optimization variables* considered: *actuation* (or control) variables, *shape*, *material/meta-material* design, or *actuation or sensor layout* parameters. In addition, we list which type of actuation method is used. For a number of papers, the deformation behavior is optimized by assuming some external force is applied, without explicitly modeling the actuators producing the force — for these papers we have categorized the actuation method as *external*.

Characterization In [10], an optimization-driven approach for fitting hyperelastic material parameters to multiaxial testing data was presented. By accurately modeling the full geometry of the test specimen, their approach achieves a smaller sim-to-real gap with fewer tests, compared to conventional approaches. Tackling the dynamic problem, recent work by Hahn et al. [12] fits viscoelastic model parameters to dynamic test data of soft robots.

Actuation and Control Differentiable simulation has seen use in the design and control of tendon-driven soft robots [44], and also in locomotion and manipulation tasks where contact with the environment must be considered [45,46].

Rod models have seen widespread use in soft body modeling. In [47], a system for the quasi-static design of flexible rod meshes was proposed, co-optimizing shape and external actuation forces. Differentiable quasi-statics has also seen use in the design and control of flexible tendon-actuated wire robots. A quasi-static-based optimization for the function-preserving conversion of traditional into compliant mechanisms was described by Megaro et al. [48], with applications in soft robots. Studying the dynamic problem, [25] addressed the vibration-minimizing retargeting of motions onto lightweight robots, consisting of rod-like components and traditional actuators.

The design and actuation of pneumatic chambers has been a topic of study in several related works. Early work by Skouras et al. [49] optimized the shape of balloons such that a desired shape was obtained under inflation — there are clear parallels to soft robotic applications. More recent work has looked at the shape optimization of tendon-driven soft robots [38,41] and also the shape optimization of pneumatic actuators [32,35,36,50].

To achieve a desired deformation behavior, the co-optimization of the tendon routing together with the material distribution was tackled in [51]. Bern et al. [52] developed a system for automatically routing tendons through plush robots in order to match target deformations. The related problem of designing a cable network to actuate hierarchical assemblies of rigid components, jointed together with compliant hinges, was addressed by Megaro et al. [53]. The design of pneumatic chambers, created with multi-material 3d printing, has also been studied [54]. Recently, Maloisel et al. [11] presented an automated system for routing McKibben artificial muscle actuators through a soft-bodied robot.

Tackling the dynamics case, Geilinger et al. [22] presented a differentiable dynamics simulator with contact, allowing for the solution of inverse dynamics problems for bouncing soft bodies.

State Estimation State estimation for soft robots is an inherently challenging problem, due to the high (infinite) dimensionality of the state. By leveraging a FEM-driven model in the state estimation, one can solve for the soft-body state that best explains a set of observed measurements while minimizing overall energy. This has been recently demonstrated

Table 1 Table categorizing discussed work, with regards to the type of problem solved, the optimization variables, and the class of sensors/actuators. See the main text for details. Entries have been ordered chronologically.

		Problem			**Setting**			**Optimization Variables**						
First Author, Year	**Differentiable Simulation**	*Characterization*	*Actuation*	*State Estimation*	*Quasi-Static*	*Dynamic*	*Contact*	*Actuation*	*Material Model*	*Metamaterial/ Multi-Material*	*Actuator/ Sensor Design*	*Shape*	**Sensing/Actuation Method**	**Ref.**
Skouras, 2012	•		x		x			x				x	Pneumatic	[49]
Skouras, 2013	•		x		x			x		x	x		Tendons; external	[51]
Liu, 2014	•		x		x							x	External	[15]
Panetta, 2015	•		x		x					x			External	[40]
Pérez, 2015	•		x		x					x			External	[47]
Schumacher, 2015	•		x		x					x			External	[33]
Lum, 2016			x		x			x			x		Magnetic (external)	[57]
Megaro, 2017	•		x		x			x			x		Tendons	[53]
Bern, 2017	•		x		x			x			x		Tendons	[52]
Megaro, 2017	•		x		x							x	Servomotors, external	[48]
Bern, 2017	•		x		x			x					Tendons	[44]
Zhang, 2017	•		x		x							x	Pneumatic	[36]
Ma, 2017	•		x		x					x	x		Pneumatic	[54]
Zehnder, 2017	•		x		x					x			External	[39]
Dämmer, 2018			x		x							x	Pneumatic	[50]
Liu, 2018	•		x		x							x	External	[34]
Chen, 2018	•		x		x							x	Tendons	[38]
Zhang, 2018	•		x		x							x	Pneumatic	[35]
Xu, 2018	•		x		x			x		x		x	Tendons	[24]
Goury, 2018			x		x	x	x	x					Tendons; pneumatic	[74]
Coevoet, 2019	•		x			x	x	x					Tendons; pneumatic; jamming	[46]
Morzadec, 2019			x		x			x				x	Tendons	[41]
Hu, 2019	•	x	x			x	x	x	x				Servomotors; abstracted (sim)	[29]
Bern, 2019	•		x			x	x	x					Tendons	[45]
Hoshyari, 2019	•		x			x		x					Servomotors	[25]
Chen, 2019	•		x		x						x		DEAs	[37]
Hafner, 2019	•		x		x							x	External	[20]
Hahn, 2019	•	x							x				Tendons; external	[12]
Chen, 2019	•		x		x							x	Pneumatic	[32]
Spielberg, 2019	•	x	x			x	x	x	x				Servomotors; abstracted (sim)	[28]
Bern, 2020	•		x		x			x					Tendons	[30]
Schumacher, 2020	•	x			x				x				External	[10]
Tapia, 2020	•			x	x						x		Liquid-metal strain sensors	[14]
Tian, 2020	•		x		x							x	Magnetic (external)	[56]
Navarro, 2020				x	x								Capacitive; pressure	[55]
Chen, 2020	•		x		x						x		DEAs	[31]
Geilinger, 2020	•		x			x	x	x					Servomotors; ballistic	[22]
Maloisel, 2021	•		x		x			x			x		Pneumatic (McKibben)	[11]
Du, 2021	•	x	x			x	x	x	x		x		Abstracted (sim)	[27]

by Navarro et al. [55], where a soft body state is reconstructed from a combination of capacitive and pressure sensors, explaining the measurements best.

Work by Tapia et al. [14] addressed this state estimation problem, but also the problem of designing an optimal sensor network in order to best reconstruct a set of deformations. They considered soft liquid-metal strain sensors routed through a soft body, and developed an optimization pipeline that selects a subset of sensors from a large initial set.

Novel Actuators Most of the problems discussed thus far used established actuation methods such as tendons or pneumatic actuators. However, as new actuation technologies for soft robots mature, new and interesting design problems emerge. One example of such a problem that has been addressed is ferromagnetic actuation, where the magnetization profile of a robot can be optimized such that it deforms in a specified way when a magnetic field is applied [56,57]. Another example is the design of electrode patterns for Dielectric Elastomer Actuators (DEA), which are a class of entirely-soft electroactive polymer actuators, in order to achieve a desired deformation behavior [31,37].

4 Conclusions

This survey has reviewed recent work in the domain of differentiable simulation for design and control of soft robotics, and described the basic ingredients of the required mathematical formulation in a versatile framework. We have shown that differentiable simulation has been applied to a number of different problems in soft robotics, including characterization, actuation, and state estimation, in quasi-static and dynamic settings, and using a number of soft robot materials and actuation methods. With the rapid advancement of the soft robotics field, and the inclusion of more advanced optimization techniques and machine learning, it is our expectation that we will see significant new contributions as the field continues to grow over the coming years.

4.1 Open Problems

It is to be expected that novel approaches for creating soft robots will first be demonstrated *in robio*, before the modeling efforts are undertaken to solve a corresponding design problem. Here, we list some promising recent works that highlight exciting opportunities for future research.

There is much work on electroactive polymers [58], and although we have seen some work in simulation-driven design [37], there is significant unexplored potential. In particular, novel actuation approaches such as, e.g., zipping ribbon actuators [59] and also novel applications such as crawling robots [60] and even autonomous untethered robots [61] hold interesting potential.

Simulation-driven design could also hold the key to a more direct integration of artificial muscle actuators into soft-bodied robots. This has been demonstrated with McKibben pneumatic artificial muscles [11], but there are exciting opportunities for other fiber-based actuators such as shape memory alloys [6] or twisted-polymer muscles [62,63].

Simulation-driven state estimation for soft robots has been demonstrated for liquid-metal strain sensors [14] and also used to combine capacitive and pressure sensing [55]. Similar approaches could be taken for other sensory data such as force/torque, pressure [64], IMU, or optical data [65].

Soft robots are starting to make their way out of the lab: walking [66], swimming [67], and flying [68]. Recent years have also seen entirely new breeds of soft robots emerge: growing vine robots [69] and energy-storing jumping robots [70], opening up new modeling and design problems; and leveraged origami and kirigami techniques for producing complex behaviors with simple and entirely soft structures [71–73]. This presents a number of computational design challenges, such as differentiable multi-domain simulation for swimming and flying robots, or how to appropriately simulate robot interactions with unstructured and potentially unknown environments.

Fully-automated design and fabrication pipelines for soft robots have been demonstrated [8], but are still in their infancy. Such automated pipelines are perfectly suited for integration with automated simulation-driven design workflows, for systems that autonomously design and create functional robots given a requirement specification.

With regards to computational tools, execution speed remains an open challenge, in particular for high-complexity problems such as design optimizations of dynamic systems. Model reduction techniques [74] hold the promise of substantial speedup, and the combination of simulations with deep learning methods [26,29] would also appear to hold vast untapped potential.

Acknowledgements

The authors are employed by The Walt Disney Company. The Walt Disney Company owns patents that are broadly relevant to the work.

Compliance with Ethical Standards

Conflict of Interest The authors declare that they have no conflict of interest.

Human and Animal Rights and Informed Consent This article does not contain any studies with human or animal subjects performed by any of the authors.

References

1. 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. IEEE Computer Society; 1991. p. 1622–1623.
2. Shepherd RF, Ilievski F, Choi W, Morin SA, Stokes AA, Mazzeo AD, et al. Multigait soft robot. Proceedings of the national academy of sciences. 2011;108(51):20400–20403.
3. Chen F, Wang MY. Design Optimization of Soft Robots: A Review of the State of the Art. IEEE Robotics & Automation Magazine. 2020;27(4):27–43.
4. Rus D, Tolley MT. Design, fabrication and control of soft robots. Nature. 2015;521(7553):467–475.
5. Yang Y, Wu Y, Li C, Yang X, Chen W. Flexible actuators for soft robotics. Advanced Intelligent Systems. 2020;2(1):1900077.
6. Laschi C, Cianchetti M, Mazzolai B, Margheri L, Follador M, Dario P. Soft robot arm inspired by the octopus. Advanced robotics. 2012;26(7):709–727.
7. Mazzolai B, Laschi C. A vision for future bioinspired and biohybrid robots. Science Robotics. 2020;5(38).
8. Wehner M, Truby RL, Fitzgerald DJ, Mosadegh B, Whitesides GM, Lewis JA, et al. An integrated design and fabrication strategy for entirely soft, autonomous robots. Nature. 2016;536(7617):451–455.

9. Belytschko T, Liu W, Moran B. Nonlinear Finite Elements for Continua and Structures. 2nd ed. John Wiley & Sons, Ltd; 2013.
10. Schumacher C, Knoop E, Bächer M. Simulation-Ready Characterization of Soft Robotic Materials. IEEE Robotics and Automation Letters. 2020;5(3):3775–3782.
11. •• Maloisel G, Knoop E, Schumacher C, Bächer M. Automated Routing of Muscle Fibers for Soft Robots. IEEE Transactions on Robotics (T-RO). 2021. **Automated design optimization of artificial muscle fiber routings, using MLS to permit fibers to move smoothly through the underlying simulation mesh.**
12. • Hahn D, Banzet P, Bern JM, Coros S. Real2sim: Viscoelastic parameter estimation from dynamic motion. ACM Transactions on Graphics (TOG). 2019;38(6):1–13. **Optimization-driven characterization of viscoelastic behavior of soft robots, from motion-capture data.**
13. Wang H, Totaro M, Beccai L. Toward Perceptive Soft Robots: Progress and Challenges. Advanced Science. 2018;5(9):1800541.
14. •• Tapia J, Knoop E, Mutný M, Otaduy MA, Bächer M. MakeSense: Automated Sensor Design for Proprioceptive Soft Robots. Soft Robotics. 2020;7(3). **Automated design of networks of embedded soft strain sensors, formulated using a three-level nesting: differentiable simulation within state estimation within sensor selection.**
15. Liu Y, Wang MY. Topology design of a conforming gripper with distributed compliance via a level set method. In: 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014). IEEE; 2014. p. 2191–2196.
16. Nocedal J, Wright SJ. Numerical Optimization. 2nd ed. New York, NY, USA: Springer; 2006.
17. Bradley AM. PDE-constrained optimization and the adjoint method. Stanford University; 2010.
18. Cao Y, Li S, Petzold L. Adjoint sensitivity analysis for differential-algebraic equations: algorithms and software. Journal of Computational and Applied Mathematics. 2002;149(1):171–191.
19. Babuška I, Banerjee U, Osborn JE. Meshless and Generalized Finite Element Methods: A Survey of Some Major Results. In: Griebel M, Schweitzer MA, editors. Meshfree Methods for Partial Differential Equations. Berlin, Heidelberg: Springer Berlin Heidelberg; 2003. p. 1–20.
20. Hafner C, Schumacher C, Knoop E, Auzinger T, Bickel B, Bächer M. X-CAD: optimizing CAD models with extended finite elements. ACM Transactions on Graphics (TOG). 2019;38(6):1–15.
21. Schneider T, Hu Y, Dumas J, Gao X, Panozzo D, Zorin D. Decoupling Simulation Accuracy from Mesh Quality. ACM Trans Graph. 2018 Dec;37(6).
22. Geilinger M, Hahn D, Zehnder J, Bächer M, Thomaszewski B, Coros S. ADD: analytically differentiable dynamics for multibody systems with frictional contact. ACM Transactions on Graphics (TOG). 2020;39(6):1–15.
23. Todorov E. Convex and analytically-invertible dynamics with contacts and constraints: Theory and implementation in MuJoCo. In: 2014 IEEE International Conference on Robotics and Automation (ICRA); 2014. p. 6054–6061.
24. Xu H, Knoop E, Coros S, Bächer M. Bend-it: design and fabrication of kinetic wire characters. ACM Transactions on Graphics (TOG). 2018;37(6):1–15.
25. Hoshyari S, Xu H, Knoop E, Coros S, Bächer M. Vibration-minimizing motion retargeting for robotic characters. ACM Transactions on Graphics (TOG). 2019;38(4):1–14.
26. Heiden E, Millard D, Coumans E, Sheng Y, Sukhatme GS. NeuralSim: Augmenting Differentiable Simulators with Neural Networks. arXiv preprint arXiv:201104217. 2020;.
27. Du T, Wu K, Ma P, Wah S, Spielberg A, Rus D, et al. DiffPD: Differentiable Projective Dynamics with Contact. arXiv preprint arXiv:210105917. 2021;.
28. • Spielberg A, Zhao A, Hu Y, Du T, Matusik W, Rus D. Learning-in-the-loop optimization: End-to-end control and co-design of soft robots through learned deep latent representations. Advances in Neural Information Processing Systems. 2019;32:8284–8294. **Using deep learning methods for the design of soft robots, building on top of a neural simulator.**
29. • Hu Y, Liu J, Spielberg A, Tenenbaum JB, Freeman WT, Wu J, et al. Chainqueen: A real-time differentiable physical simulator for soft robotics. In: 2019 International Conference on Robotics and Automation (ICRA). IEEE; 2019. p. 6265–6271. **Neural simulator for soft-bodied robots, used to solve control and design problems for dynamic robots with contact.**
30. Bern JM, Schnider Y, Banzet P, Kumar N, Coros S. Soft robot control with a learned differentiable model. In: 2020 3rd IEEE International Conference on Soft Robotics (RoboSoft). IEEE; 2020. p. 417–423.
31. Chen B, Wang N, Zhang X, Chen W. Design of dielectric elastomer actuators using topology optimization on electrodes. Smart Materials and Structures. 2020;29(7):075029.
32. Chen Y, Xia Z, Zhao Q. Optimal design of soft pneumatic bending actuators subjected to design-dependent pressure loads. IEEE/ASME Transactions on Mechatronics. 2019;24(6):2873–2884.
33. Schumacher C, Bickel B, Rys J, Marschner S, Daraio C, Gross M. Microstructures to control elasticity in 3D printing. ACM Transactions on Graphics (TOG). 2015;34(4):1–13.
34. Liu CH, Chen TL, Chiu CH, Hsu MC, Chen Y, Pai TY, et al. Optimal design of a soft robotic gripper for grasping unknown objects. Soft robotics. 2018;5(4):452–465.
35. Zhang H, Kumar AS, Chen F, Fuh JY, Wang MY. Topology optimized multimaterial soft fingers for applications on grippers, rehabilitation, and artificial hands. IEEE/ASME Transactions on Mechatronics. 2018;24(1):120–131.
36. Zhang H, Wang MY, Chen F, Wang Y, Kumar AS, Fuh JY. Design and development of a soft gripper with topology optimization. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE; 2017. p. 6239–6244.
37. Chen F, Liu K, Wang Y, Zou J, Gu G, Zhu X. Automatic design of soft dielectric elastomer actuators with optimal spatial electric fields. IEEE Transactions on Robotics. 2019;35(5):1150–1165.
38. Chen F, Xu W, Zhang H, Wang Y, Cao J, Wang MY, et al. Topology optimized design, fabrication, and characterization of a soft cable-driven gripper. IEEE Robotics and Automation Letters. 2018;3(3):2463–2470.
39. Zehnder J, Knoop E, Bächer M, Thomaszewski B. Metasilicone: design and fabrication of composite silicone with desired mechanical properties. ACM Transactions on Graphics (TOG). 2017;36(6):1–13.
40. Panetta J, Zhou Q, Malomo L, Pietroni N, Cignoni P, Zorin D. Elastic Textures for Additive Fabrication. ACM Transactions on Graphics (TOG). 2015;34(4):1–12.
41. Morzadec T, Marcha D, Duriez C. Toward shape optimization of soft robots. In: 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft). IEEE; 2019. p. 521–526.
42. Cheney N, Bongard J, Lipson H. Evolving soft robots in tight spaces. In: Proceedings of the 2015 annual conference on Genetic and Evolutionary Computation; 2015. p. 935–942.
43. Corucci F, Cheney N, Giorgio-Serchi F, Bongard J, Laschi C. Evolving soft locomotion in aquatic and terrestrial environments: effects of material properties and environmental transitions. Soft robotics. 2018;5(4):475–495.
44. Bern JM, Kumagai G, Coros S. Fabrication, modeling, and control of plush robots. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE; 2017. p. 3739–3746.
45. • Bern JM, Banzet P, Poranne R, Coros S. Trajectory Optimization for Cable-Driven Soft Robot Locomotion. In: Robotics: Science and Systems; 2019. **Trajectory optimization for locomoting tendon-driven soft robots, using a dynamic formulation with contact.**

46. 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). IEEE; 2019. p. 739–745.
47. Pérez J, Thomaszewski B, Coros S, Bickel B, Canabal JA, Sumner R, et al. Design and fabrication of flexible rod meshes. ACM Transactions on Graphics (TOG). 2015;34(4):1–12.
48. Megaro V, Zehnder J, Bächer M, Coros S, Gross M, Thomaszewski B. A Computational Design Tool for Compliant Mechanisms. ACM Trans Graph. 2017 Jul;36(4).
49. Skouras M, Thomaszewski B, Bickel B, Gross M. Computational design of rubber balloons. In: Computer Graphics Forum. vol. 31. Wiley Online Library; 2012. p. 835–844.
50. Dämmer G, Gablenz S, Hildebrandt A, Major Z. Design and shape optimization of PolyJet bellows actuators. In: 2018 IEEE International Conference on Soft Robotics (RoboSoft). IEEE; 2018. p. 282–287.
51. Skouras M, Thomaszewski B, Coros S, Bickel B, Gross M. Computational design of actuated deformable characters. ACM Transactions on Graphics (TOG). 2013;32(4):1–10.
52. Bern JM, Chang KH, Coros S. Interactive design of animated plushies. ACM Transactions on Graphics (TOG). 2017;36(4):1–11.
53. Megaro V, Knoop E, Spielberg A, Levin DI, Matusik W, Gross M, et al. Designing cable-driven actuation networks for kinematic chains and trees. In: Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation; 2017. p. 1–10.
54. Ma LK, Zhang Y, Liu Y, Zhou K, Tong X. Computational design and fabrication of soft pneumatic objects with desired deformations. ACM Transactions on Graphics (TOG). 2017;36(6):1–12.
55. Navarro SE, Nagels S, Alagi H, Faller LM, Goury O, Morales-Bieze T, et al. A model-based sensor fusion approach for force and shape estimation in soft robotics. IEEE Robotics and Automation Letters. 2020;5(4):5621–5628.
56. Tian J, Zhao X, Gu XD, Chen S. Designing ferromagnetic soft robots (FerroSoRo) with level-set-based multiphysics topology optimization. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2020. p. 10067–10074.
57. Lum GZ, Ye Z, Dong X, Marvi H, Erin O, Hu W, et al. Shape-programmable magnetic soft matter. Proceedings of the National Academy of Sciences. 2016;113(41):E6007–E6015.
58. Pelrine R, Kornbluh R, Pei Q, Joseph J. High-speed electrically actuated elastomers with strain greater than 100%. Science. 2000;287(5454):836–839.
59. Taghavi M, Helps T, Rossiter J. Electro-ribbon actuators and electro-origami robots. Science Robotics. 2018;3(25).
60. Duduta M, Clarke DR, Wood RJ. A high speed soft robot based on dielectric elastomer actuators. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2017. p. 4346–4351.
61. Ji X, Liu X, Cacucciolo V, Imboden M, Civet Y, El Haitami A, et al. An autonomous untethered fast soft robotic insect driven by low-voltage dielectric elastomer actuators. Science Robotics. 2019;4(37).
62. Haines CS, Lima MD, Li N, Spinks GM, Foroughi J, Madden JD, et al. Artificial muscles from fishing line and sewing thread. science. 2014;343(6173):868–872.
63. Kim K, Cho KH, Jung HS, Yang SY, Kim Y, Park JH, et al. Double helix twisted and coiled soft actuator from spandex and nylon. Advanced Engineering Materials. 2018;20(11):1800536.
64. Li G, Liu S, Wang L, Zhu R. Skin-inspired quadruple tactile sensors integrated on a robot hand enable object recognition. Science Robotics. 2020;5(49).
65. Ward-Cherrier B, Pestell N, Cramphorn L, Winstone B, Giannaccini ME, Rossiter J, et al. The tactip family: Soft optical tactile sensors with 3d-printed biomimetic morphologies. Soft robotics. 2018;5(2):216–227.
66. Tolley MT, Shepherd RF, Mosadegh B, Galloway KC, Wehner M, Karpelson M, et al. A resilient, untethered soft robot. Soft robotics. 2014;1(3):213–223.
67. Katzschmann RK, DelPreto J, MacCurdy R, Rus D. Exploration of underwater life with an acoustically controlled soft robotic fish. Science Robotics. 2018;3(16).
68. Ramezani A, Shi X, Chung SJ, Hutchinson S. Bat Bot (B2), a biologically inspired flying machine. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2016. p. 3219–3226.
69. Hawkes EW, Blumenschein LH, Greer JD, Okamura AM. A soft robot that navigates its environment through growth. Science Robotics. 2017;2(8).
70. Gorissen B, Melancon D, Vasios N, Torbati M, Bertoldi K. Inflatable soft jumper inspired by shell snapping. Science Robotics. 2020;5(42).
71. Rus D, Tolley MT. Design, fabrication and control of origami robots. Nature Reviews Materials. 2018;3(6):101–112.
72. Kim W, Byun J, Kim JK, Choi WY, Jakobsen K, Jakobsen J, et al. Bioinspired dual-morphing stretchable origami. Science Robotics. 2019;4(36).
73. Baek SM, Yim S, Chae SH, Lee DY, Cho KJ. Ladybird beetle–inspired compliant origami. Science Robotics. 2020;5(41).
74. Goury O, Duriez C. Fast, generic, and reliable control and simulation of soft robots using model order reduction. IEEE Transactions on Robotics. 2018;34(6):1565–1576.