



UNIVERSITÀ
DEGLI STUDI
FIRENZE

FLORE

Repository istituzionale dell'Università degli Studi di Firenze

Optimization-driven design of monolithic soft-rigid grippers

Questa è la versione Preprint (Submitted version) della seguente pubblicazione:

Original Citation:

Optimization-driven design of monolithic soft-rigid grippers / Pierluigi Mansueto, Matteo Lapucci, Mihai Dragusanu, Gionata Salvietti, Monica Malvezzi, Anjum Saeed. - In: SOFT ROBOTICS. - ISSN 2169-5180. - ELETTRONICO. - (2025), pp. 0-0. [10.1177/21695172251359016]

Availability:

The webpage <https://hdl.handle.net/2158/1435699> of the repository was last updated on 2026-03-20T11:41:37Z

Published version:

DOI: 10.1177/21695172251359016

Terms of use:

Open Access

La pubblicazione è resa disponibile sotto le norme e i termini della licenza di deposito, secondo quanto stabilito dalla Policy per l'accesso aperto dell'Università degli Studi di Firenze (<https://www.sba.unifi.it/upload/policy-oa-2016-1.pdf>)

Publisher copyright claim:

La data sopra indicata si riferisce all'ultimo aggiornamento della scheda del Repository FloRe - The above-mentioned date refers to the last update of the record in the Institutional Repository FloRe

(Article begins on next page)

OPTIMIZATION-DRIVEN DESIGN OF MONOLITHIC SOFT-RIGID GRIPPERS

Pierluigi Mansueto

Department of Information Engineering
University of Florence
Firenze, 50139, Italy
pierluigi.mansueto@unifi.it

Mihai Dragusanu

Department of Information Engineering
and Mathematics, University of Siena
Siena, 53100, Italy
mihai.dragusanu2@unisi.it

Anjum Saeed

Department of Information Engineering
and Mathematics, University of Siena
Siena, 53100, Italy
a.saeed1@student.unisi.it

Monica Malvezzi*

Department of Information Engineering
and Mathematics, University of Siena
Siena, 53100, Italy
monica.malvezzi@unisi.it

Matteo Lapucci

Department of Information Engineering
University of Florence
Firenze, 50139, Italy
matteo.lapucci@unifi.it

Gionata Salvietti*

Department of Information Engineering
and Mathematics, University of Siena
Siena, 53100, Italy
gionata.salvietti@unisi.it

ABSTRACT

Sim-to-real transfer remains a significant challenge in soft robotics due to the unpredictability introduced by common manufacturing processes such as 3D printing and molding. These processes often result in deviations from simulated designs, requiring multiple prototypes before achieving a functional system. In this study, we propose a novel methodology to address these limitations by combining advanced rapid prototyping techniques and an efficient optimization strategy. Firstly, we employ rapid prototyping methods typically used for rigid structures, leveraging their precision to fabricate compliant components with reduced manufacturing errors. Secondly, our optimization framework minimizes the need for extensive prototyping, significantly reducing the iterative design process. The methodology enables the identification of stiffness parameters that are more practical and achievable within current manufacturing capabilities. The proposed approach demonstrates a substantial improvement in the efficiency of prototype development while maintaining the desired performance characteristics. This work represents a step forward in bridging the sim-to-real gap in soft robotics, paving the way towards a faster and more reliable deployment of soft robotic systems.

Keywords Soft-rigid grippers · sim-to-real design of soft grippers · compliant mechanisms

1 Introduction

Soft multi-fingers hands [1–3] and grippers [4] have demonstrated to be an efficient solution to the usability problem of the first generation of rigid robotic hands and manipulation mechanisms [5, 6]. The use of flexible and deformable components allow robotic hands to adapt more naturally and effectively to the surrounding environment also addressing challenges arising from sensor uncertainties, particularly in defining the position and shape of the object. In fact, differently than traditional approaches that require precise contact points locations and force application, soft hands

*Humanoids & Human Centered Mechatronics, Istituto Italiano di Tecnologia, Genoa, 16163, Italy

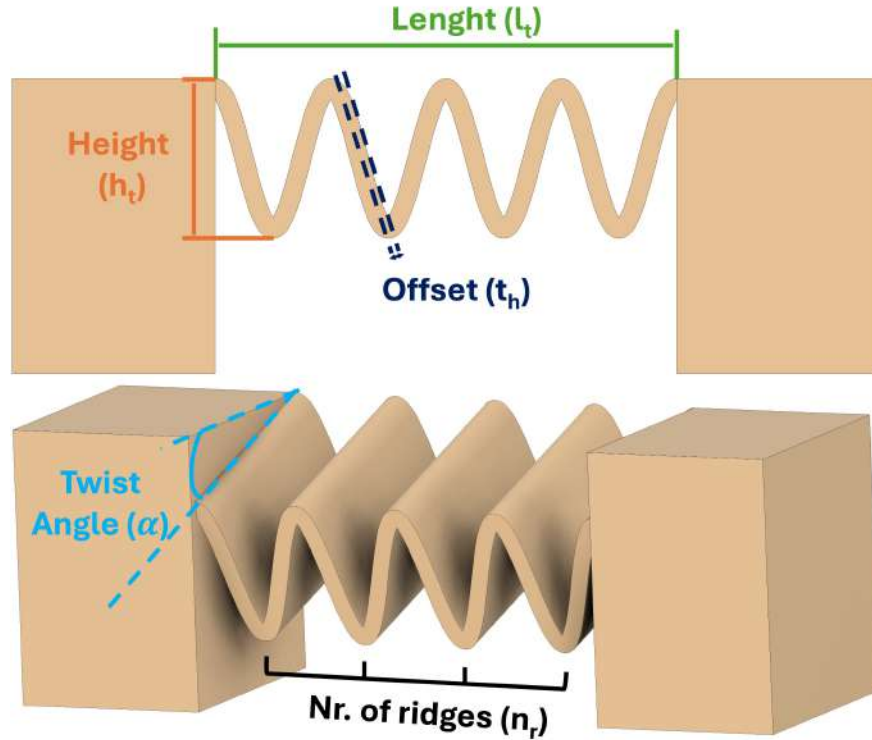


Figure 1: Scheme of the WaveJoint with the parameters that can be varied in the design process based on the desired stiffness.

exploit their flexibility to passively adapt to various objects shape [7].

Using soft materials offer numerous advantages, including improved safety in interactions with humans and reduced damage occurred to fragile objects during manipulation tasks. The compliance in these robotic hands can be achieved through various techniques, such as advanced 3D-printing techniques [8] and the use of deformable bio-materials [9]. However, these fabrication techniques are prone to possible precision errors often leading to prototypes that presents mechanical characteristics different from those designed. Moreover, in more complex manipulation challenges, soft hands may face limitations in terms of dexterity [10]. These limitations arise from underactuation, which implies a reduced ability to apply high forces, and the difficulty in performing precise contact modeling [11].

Soft-rigid grippers have been recently introduced to try to partially solve the above mentioned issues [12–14]. These end-effectors are based on the combination of rigid structures and soft joints to grip and manipulate objects. The underlying principle for the actuation system is to use a tendon transmission system, where the gripper fingers are defined by the concatenation of modules made of rigid material (ABS, PLA) and soft joints [15]. This configuration consent to achieve different flexion trajectories of the fingers by varying the stiffness of the soft joints [16] increasing the dexterity of the grippers. Stiffness of soft joints can be selected in different ways, varying, for instance the geometry or the material used (silicons with different hardness). An interesting opportunity come from 3D printing techniques, where it is possible to act on the *infill density* during the printing process. Commercial 3D printer software generally offers the ability to select infill density as a parameter during the additive manufacturing process. However, the accuracy and repeatability of stiffness values for soft joints is not very high, unless still expensive multi-material 3D printers are used [17].

This limitation underline the importance of advanced design tools, which enable the synthesis of mechanisms using a minimal set of mechanical components, where each component's shape is carefully tailored to ensure the intended mobility functions [18, 19]. Recently, these tools have been leveraged to design compliant robotic grippers and hands [20, 21]. Mutlu et al. [22] modeled and experimentally evaluated the bending behavior of established flexure hinge designs to determine the optimal configuration for a fully compliant prosthetic finger. Liu et al. [23] presented a two-fingered gripper for fruit handling, where each finger features a monolithic, compliant structure shaped through topology optimization. Dragusanu et al. [24] presented a methodology for designing robotic grippers with 3D-printed, tendon-actuated fingers based on an innovative compliant joint, the WaveJoint (see Figure 1). This approach employs

wave-shaped joints to have adjustable finger stiffness that can be modified acting on the wave geometry during design or real-time via pre-compression [25].

The correspondence between stiffness values and WaveJoint design parameters is not straightforward. One possible approach to deal with this issue is to estimate a direct mapping training a neural network [24]. However, this approach presents several significant drawbacks: *i)* generating the training data is complex; *ii)* modeling soft materials is challenging, causing simulations to often diverge from actual behavior; *iii)* neural networks generally require large computational resources, increasing design time and complexity.

In this work, we propose an optimization-based approach to directly search for the geometric properties of the WaveJoint that lead to the desired stiffness. This task is modeled as a black-box global optimization problem, and is then handled by means of a surrogate based approach [26, 27] that requires a low number of measurements. Among the advantages of this method we can list the possibility of handling constraints to the geometry of the gripper, of obtaining a solution with a reasonable number of measurements when no data at start and of obtaining, instead, a direct mapping when the number of measurements becomes large enough. Considering a fixed geometric structure (Figure 1), we show that the proposed algorithm is effective at identifying suitable geometric parameters both in simulated and real world tests.

2 Design of Joint Characteristics by Global Optimization

2.1 Problem statement

A key step in the design of soft-rigid grippers based on WaveJoints amounts to the definition of the flexible joint geometric properties. The correct design of the geometric structure of the joints allows to get suitable stiffness values that lead to the desired fingertip trajectories. For the specific case of WaveJoints, as shown in Figure 1 the geometrical characteristics of the joint are:

- l_t : joint length - a real number within the interval $[l_{\min}, l_{\max}]$;
- n_r : number of ridges - an integer between $n_{\min}$ and $n_{\max}$;
- h_t : wave height (i.e., sinusoidal amplitude) - a real number within the interval $[h_{\min}, h_{\max}]$;
- t_h : offset (i.e., wave thickness) - a real number within the interval $[t_{\min}, t_{\max}]$;
- α : twist angle - an integer between $\alpha_{\min}$ and $\alpha_{\max}$.

The problem consists of detecting a set of parameters $(l_t, n_r, h_t, t_h, \alpha)$ that provides a target set of stiffness values (k_ξ, k_η, k_ζ) , synthesized on the basis of the gripper specifications.

Let us denote by $\phi : \mathbb{R}^5 \rightarrow \mathbb{R}^3$ the function mapping the geometric properties of the joints onto the corresponding stiffness values. In order to accomplish the aforementioned task, the approach followed by Dragusanu et al. [24] consisted in approximating the inverse relation ϕ^{-1} by means of a feed-forward neural network, trained on a small set of data samples. Once such a network is trained, it can conceptually be used to estimate, for any given triple (k_ξ, k_η, k_ζ) , a suitable geometrical configuration of the joint. This approach has a number of significant shortcomings:

1. *large amounts of data* are required to train neural networks with sufficient expressive power to approximate ϕ^{-1} accurately enough;
2. as *new data* is available, the training process (together with hyperparameters validation) has to be *repeated from scratch*: not only this job is computationally expensive, but it also likely produces a model which is significantly different than the starting one;
3. the approach is *incapable of dealing with failure*: assume the configuration of geometric parameters suggested by the neural network model is then found to be wrong, i.e., it actually leads to stiffness values much different than the target ones; then, there is no way of obtaining any kind of second guess;
4. finally, the parameters $(l_t, n_r, h_t, t_h, \alpha)$ have feasible ranges, and some of them have to take integer values; neural network outputs cannot guarantee these constraints to be satisfied, so that truncation or projection of the retrieved values will often be required in practice; this rounding and readjustment operations are rough and reasonably lead to deteriorating the network performance.

We propose a different approach for the design of joints geometrical properties based on mathematical optimization. The function ϕ can be seen as a black box function that can be evaluated, for any setting of $(l_t, n_r, h_t, t_h, \alpha)$, by means of either a structural mechanics analysis based on Finite Element Method (FEM) tools or measurements on a real-world prototype. Given target stiffness values $\hat{k} = (k_\xi, k_\eta, k_\zeta)$, we want to obtain values of $(l_t, n_r, h_t, t_h, \alpha)$ such that:

- each parameter takes a value which is physically consistent and implementable;
- $\phi(l_t, n_r, h_t, t_h, \alpha) = \hat{k}$.

Such values can be obtained as optimal solutions of the following optimization problem

$$\begin{aligned}
 \min_{\substack{l_t, h_c, t_h \in \mathbb{R} \\ n_r, \alpha \in \mathbb{Z}}} f(l_t, n_r, h_t, t_h, \alpha) &= \|\phi(l_t, n_r, h_t, t_h, \alpha) - \hat{k}\|_2^2 \\
 \text{s.t.} \quad & l_{\min} \leq l_t \leq l_{\max}, \quad h_{\min} \leq h_c \leq h_{\max}, \\
 & t_{\min} \leq t_h \leq t_{\max}, \quad n_{\min} \leq n_r \leq n_{\max}, \\
 & \alpha_{\min} \leq \alpha \leq \alpha_{\max}.
 \end{aligned} \tag{1}$$

Using the terminology from the mathematical optimization literature, problem (1) is an instance of global optimization of computationally expensive black-box functions [28, 29]. The key features of this setting are that an analytical expression (and thus derivatives) is not available for the objective function and that its evaluation at some solution has a computational cost that dominates any other operation involved in the optimization process.

Of course, the issue can be generalized to any design process where target properties have to be obtained by properly setting the fundamental characteristics of a mechanical device.

2.2 RBF based optimization of expensive black-box functions

To tackle expensive black-box optimization tasks, approaches based on constructing surrogates of the objective function, with increasing accuracy as the number of observations grows, have proved to be solid choices [30]. Briefly, at each iteration, methods of this kind:

- (i) evaluate the objective function on a new point;
- (ii) if the solution is satisfactory, the procedure stops;
- (iii) otherwise, the new observation is added to previous information and used to construct a more accurate approximation (surrogate) of the objective function;
- (iv) the next point to be evaluated is the global minimizer of an *acquisition function*, i.e., a modified surrogate that privileges unexplored regions of the parameters space.

A popular approach within this paradigm is that of Radial Basis Function methods [26, 27]. For a formal introduction to this class of methods we refer readers to Appendix A.

An efficient RBF surrogate based algorithm, dealing with expensive black-box optimization problems with bound constraints and possibly integer variables, is available with the recent RBFOpt [27] software library. This software is thus well suited to be used to tackle problem (1). In Figure 2, we report a graphical overview of the proposed optimization-based procedure built upon RBFOpt.

This approaches usually require a few tens of function evaluations to reach good quality solutions. Yet, this cost might represent a high computational burden to pay for solving a single gripper design problem, especially if the simulation process were not automated or if we necessarily had to resort to real-world measurements. However, we shall note that, whereas the objective function changes each time we consider a new target $\hat{k}$, the function ϕ that we evaluate by expensive simulation or measurement is always the same. Thus, when we want to find the parameters corresponding to $\hat{k}$ by solving problem (1), we can also employ data gathered for past estimation tasks to construct the surrogate, reducing the number of measurements required to determine each single parameter configuration.

This kind of approach overcomes most of the limitations highlighted for the neural networks based methodology; in particular:

1. the algorithm becomes *increasingly efficient* with time, since the amount of “free” data at the initialization step becomes larger each time an estimation task is solved;
2. *no failure* occurs; if the proposed parameters setting is found to be unsuitable to obtain the desired stiffness values, it can be added as a new observation to the interpolation data; this process can be repeated until a satisfactory solution is retrieved;
3. since we work in the parameters space, we can *explicitly control* the physical *requirements*, treating them as constraints in the optimization problem; all solutions will intrinsically satisfy both bound and integrity constraints.

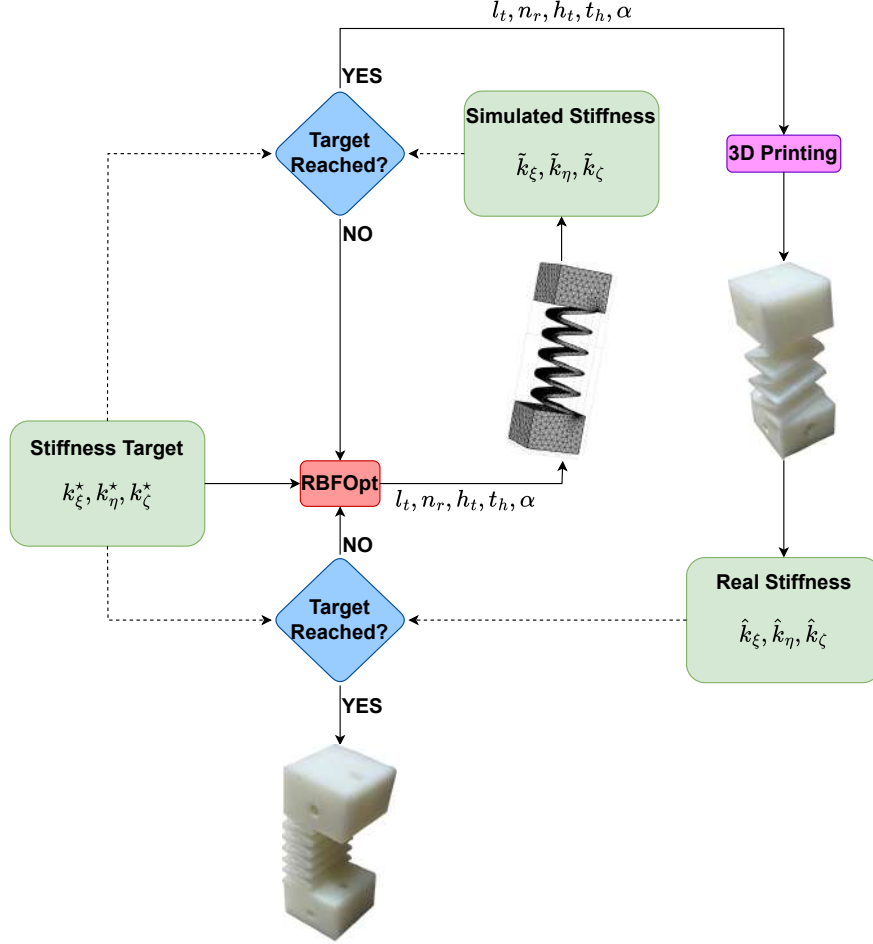


Figure 2: Flowchart of the optimization-based procedure with RBFOpt as black-box optimization software library.

The methodology proposed here is particularly well suited if the entire process can be automated; this is possible, for example, if the model definition in a simulation environment can be parameterized with respect to the values of the geometric parameters to be estimated - $(l_t, n_r, h_t, t_h, \alpha)$ in the case of WaveJoints. Under this condition, users will be able to carry out a larger number of simulations, i.e., more iterations, and consequently obtain higher quality solutions.

Advanced versions of RBF algorithms - such as the one in RBFOpt library - are also capable of dealing with noise; in particular, RBFOpt is able to distinguish exact observations, assumed to be obtained by a more expensive oracle, and noisy observations from a cheaper oracle. This is particularly useful in a setting where we have access to both a simulator and real-world prototypes: the results of a simulation might be imprecise (i.e., noisy), whereas the direct feedback from a real-world prototype should be considered fully reliable; on the other hand, it is obvious that building a prototype is much more expensive than running a simulation. The RBFOpt [27] algorithm then operates in such a way that it collects larger amounts of data from the cheaper oracle: these data points, however, are not necessarily interpolated by the surrogate, which rather passes through within confidence interval of the observed noisy value.

2.3 Incremental Neural Network approach

If we are open to storing observation results - whether they are from simulations or real-world measurements - and give up having a fixed and definitive network, we might consider adopting an incremental-type scheme, similar to that of RBF approaches, based on neural networks. In particular, one may consider a procedure that, given a set of past observations $\mathcal{D}$ and target values T to obtain with a new configuration x , follows these steps:

- (i) Train, on dataset $\mathcal{D}$, a neural net N to approximate ϕ^{-1} ;
- (ii) Check output $x = N(T)$ of the net given target values T ;

- (iii) Measure the true value of $\phi(x)$;
- (iv) If the value of $\phi(x)$ is close enough to T , then stop; otherwise, add the pair (T, x) to the dataset $\mathcal{D}$ and go back to step (i).

As we will show numerically, this approach is actually employable, and shows performance close to those of the RBF optimization approach. However, contrarily to the latter method, this approach cannot

- manage constraints on the values of the parameters x to be estimated;
- manage failures possibly occurring due to physically impossible values of x returned by the network as an output; in those cases, it would not be clear where to place the next observation;
- distinguish between exact and noisy observations.

Taking into account the above points, the optimization-based approach appears thus to be the most robust methodology, both conceptually and practically.

3 Experiments

Throughout this section, we will require that the geometric parameters of the joint satisfy the following constraints related to physical viability: $l_t \in [15, 30]$, $n_r \in [3, 6]$, $h_c \in [8, 14]$, $t_h \in [0.5, 0.6]$, $\alpha \in [0, 24]$. We carried out all computations on a computer with the following specifics: Ubuntu 22.04, Intel(R) Core(TM) i5-10600KF CPU @ 4.10GHz, 32 GB RAM. The code implementing the optimization process was written in Python3, exploiting the RBFOpt library. For Neural Networks, we employed the `scikit-learn` Python library. When not carried out on a real-world prototype, the computation of $k = \phi(l_t, n_r, h_c, t_h, \alpha)$ is done by a structural mechanics analysis based on FEM, with simulations done on the COMSOL Multiphysics 6.0 platform, accessed through the MATLAB interface. The parametric design of WaveJoints within COMSOL was done exploiting the following parametric equation, which describes a cosine function characterized by amplitude, frequency, and phase

$$y(t) = - \left[\left(\frac{h_t}{2} - t_h \right) \cos \left(2\pi \frac{n_r}{l_t} t \right) \right]. \quad (2)$$

For the rigid parts of the WaveJoint, a simple geometry with fixed parameters was used, specifically a rectangular parallelepiped with dimensions of $11 \times 15 \times 16$ mm. The component was modeled in COMSOL using the Geometry workspace, where two distinct work planes were defined: one for creating the wave using (2) and the other for the two parallelepipeds. The wave's parameters and the dimensions of the parallelepipeds were defined through the Parameters section in the Global Definitions workspace. Finally, the 3D component of the WaveJoint was obtained by applying a series of operations: the extrusion of the wave and the parallelepipeds, the rotation of the wave based on α , and the union of the three elements into a single body.

By varying the parameters shown in Figure 1, different configurations of the joint can be achieved. The material was set to ABS (Acrylonitrile Butadiene Styrene) with a Young's modulus $E = 2.30$ GPa, density $\rho = 1.06$ g/cm³, and Poisson's ratio $\nu = 0.35$. Experiments were carried out setting the mesh element size in COMSOL to `fine`.

Each FEM simulation was evaluated in terms of displacement, using point probes positioned on the parallelepiped vertices (Figure 3). Three loading cases were simulated: bending in the longitudinal plane (Figure 3a), bending in the lateral plane (Figure 3b), and torsion (Figure 3c). In each case, a force of 1 N was applied. For the torsion force simulation, two equal and opposite forces were applied on the lateral parallelepiped faces, as shown in Figure 3c, making an output momentum of 1 Nmm. The corresponding stiffness was evaluated based on the final displacement obtained and the parallelepiped geometry. The result, in terms of displacement, of a representative simulation with $n_r = 4$, $l_t = 25$ mm, $t_h = 0.5$ mm, $\alpha = 10^\circ$, and $h_t = 5$ mm, is shown in Figure 3. Notice that some parameters configurations might lead to COMSOL failures, due to physical values inconsistencies. In these cases, the objective value within RBFOpt is considered to be $+\infty$.

Concerning the physically fabricated joints, all components were realized using FDM (Fused Deposition Modeling) 3D printing with ABS material, the same assumed in simulations. Each sample was experimentally tested using the motorized force test rig MARK-10 F305 shown in Figure 4. This setup consists of a vertical testing station equipped with a FS05-200 dynamometer and a EMP001-1 distance measurement package. During testing, deformation was induced by a flexion test, where applied forces and corresponding displacements were recorded. The testing force was set to 1 N, consistent with simulation parameters.

To measure deformation along the three axes and calculate joint stiffness, k_ξ, k_η, k_ζ , supports were made. Figures 4a and 4b illustrate how the module was evaluated for bending in both the longitudinal (k_ξ) and lateral planes (k_η). For

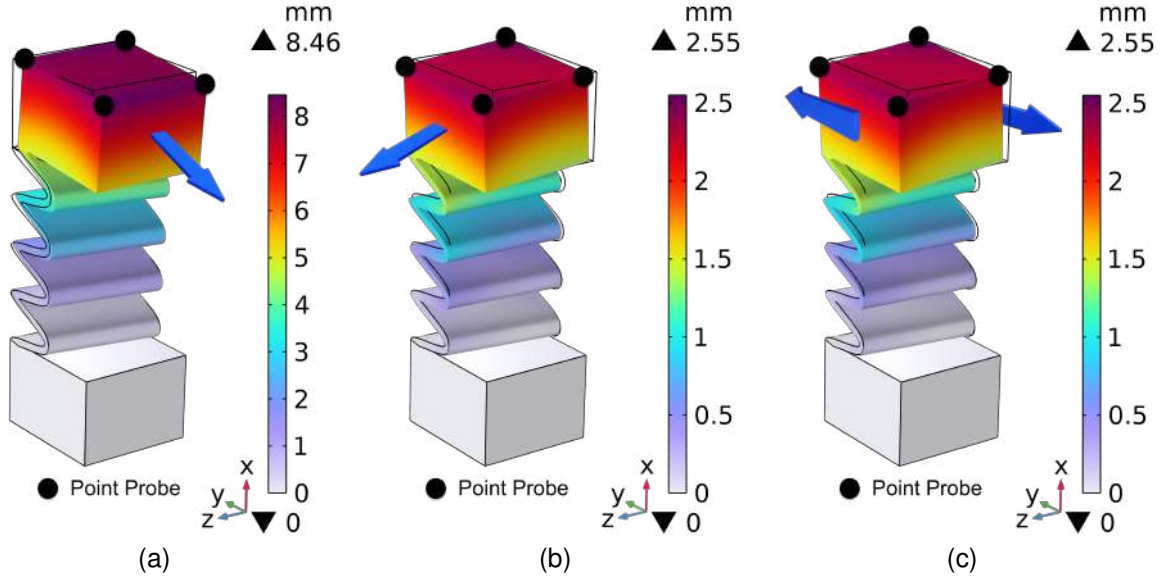


Figure 3: Results in terms of displacements of a representative simulation for stiffness evaluation based on FEM structural analysis. For each simulation a total of 1 N force was applied with the following parameters of the WaveJoint: $n_r = 4$, $l_t = 25\text{mm}$, $t_h = 0.5\text{mm}$, $\alpha = 10^\circ$, and $h_t = 5\text{mm}$. (a) Longitudinal (k_ξ) bending simulation. (b) Lateral (k_η) bending simulation. (c) Torsional (k_ζ) simulation.

torsional testing (k_ζ), we adapted the linear movement of the test rig to rotational motion through a rack and pinion mechanism, as shown in Figure 4c. The pinion gear and the WaveJoint were rigidly connected via a custom housing part on the gear, to ensure the rotational motion of the gear was transferred to the joint.

For each tested module, the k_ξ , k_η , and k_ζ evaluation involved three distinct tests, each consisting of five consecutive trials. During these trials, the test rig applied a holding test force, and the resulting displacements was recorded. The joint stiffness, along the three axes, was then calculated based on the average displacements values.

3.1 Preliminary Experiments

In a first round of experiments, we set RBFOpt to stop when either a maximum number of iterations of 500 was reached, or a solution with an objective value $< 10^{-5}$ was obtained (note that we know that the value of the global optimum is 0).

As a first experiment, we chose different “Target” configurations of geometric parameters of the WaveJoint ($l_t, n_r, h_t, t_h, \alpha$) and computed the corresponding stiffness values (k_ξ, k_η, k_ζ) by FEM simulations in COMSOL. For each of these test cases, we ran RBFOpt to solve the corresponding instance of problem (1). Note that, instead of the simpler objective function from (1), we used the squared norm of relative errors, i.e., $\mathcal{R} = \|R\|_2^2$, with $R_i = \frac{\phi_i(x) - T_i}{T_i}$. In the following, we will refer to this quantity as “residual”. Considering relative errors is beneficial to get uniformly accurate solutions, since the scales of values for (k_ξ, k_η, k_ζ) might be different.

Demanding residuals on the order of 10^{-5} may be excessive in real-world applications. Firstly, achieving precision up to several decimal places in stiffness values is often unnecessary to attain the desired physical behavior from the joint. Secondly, simulators introduce non-negligible errors compared to real-world observations, rendering the pursuit of such precision from a simulator nonsensical. The chosen value of desired accuracy allows us to evaluate how our approach’s precision evolves over time and the extent to which we can advance it in an ideal setting.

The experiment was repeated for 7 problem instances. We report in Table 1 (left) the results for 3 representative cases, the full table can be found in Appendix B. We can look at the final residual values to assess the quality of the obtained solutions. The process always ended with stiffness values very close to the target ones. Interestingly, the returned configuration of geometric parameters is equal to the target one only in 1 case out of 7. It is in fact not surprising that different geometric settings of the joint lead to equal stiffness values (ϕ is not injective, as it maps points to a lower dimensional space).

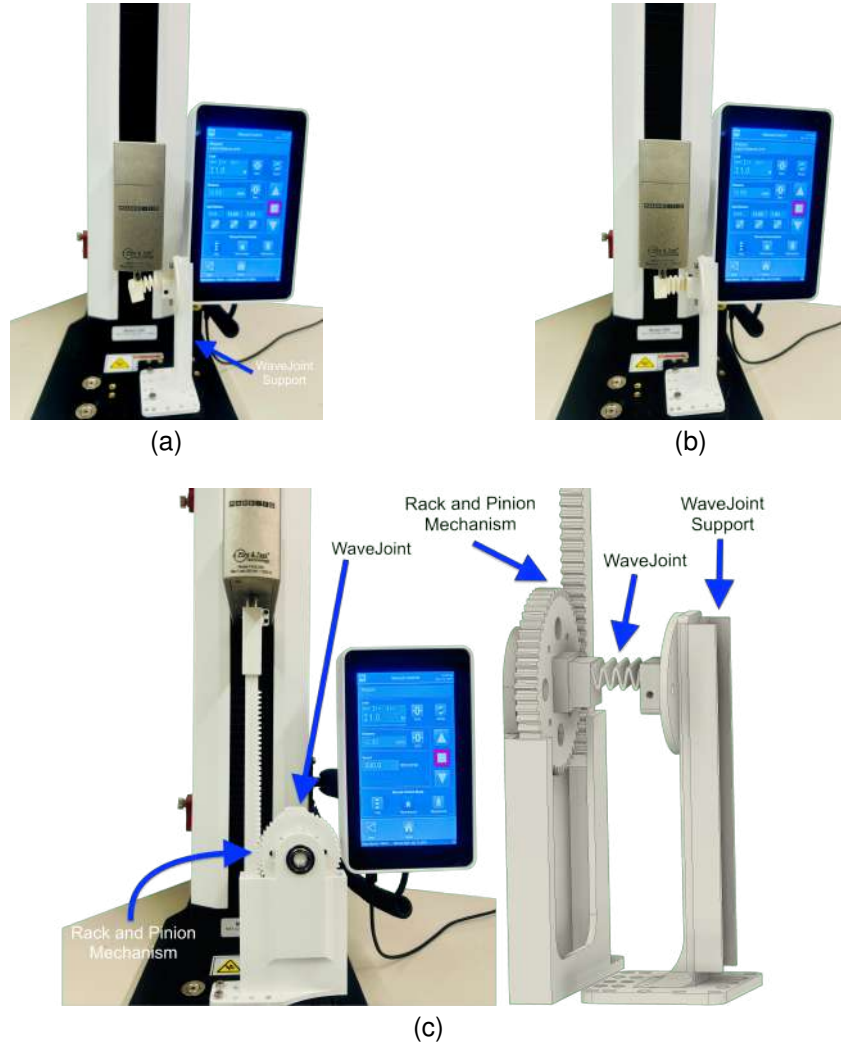


Figure 4: Test on bench for a WaveJoint sample. (a) Longitudinal (k_ξ) bending test. (b) Lateral (k_η) bending test. (c) Torsional (k_ζ) test.

Table 1: Results of WaveJoints design by RBFOpt, according to FEM simulations. On the left: no data at initialization. On the right: 3043 data points at initialization. Further results can be found in Tables B.I-B.II of Appendix B.

Type	Configuration					Stiffness			$\mathcal{R}$	Time(h)
	l_t	n_r	h_t	t_h	α	k_ξ	k_η	k_ζ		
Target	15	3	8.0	0.5	0	990.3	35.28	5.09	5e-5	2.59
Final	15.03	3	8.03	0.5	0	994.77	35.15	5.11		
Target	20	3	14	0.5	0	741.28	6	1.97	3e-5	2.63
Final	27.95	4	11.66	0.59	11	744.99	6	1.98		
Target	20	3	10	0.5	21	635.47	6.22	2.8	0	1.1
Final	27.42	3	10.64	0.6	20	635.91	6.21	2.79		

Type	Configuration					Stiffness			$\mathcal{R}$	Time(h)
	l_t	n_r	h_t	t_h	α	k_ξ	k_η	k_ζ		
Target	20	4	10	0.5	0	803.39	12.49	2.38	8e-4	4.89
Final	27.05	5	9.08	0.58	5	818.76	12.7	2.34		
Target	20	5	10	0.5	6	932.27	14.26	2.45	3e-5	4.79
Final	27.86	6	8.99	0.6	7	936.3	14.22	2.45		
Target	20	3	10	0.5	24	630.04	5.73	2.86	1e-5	4.61
Final	20.01	3	9.98	0.5	24	629.24	5.72	2.87		

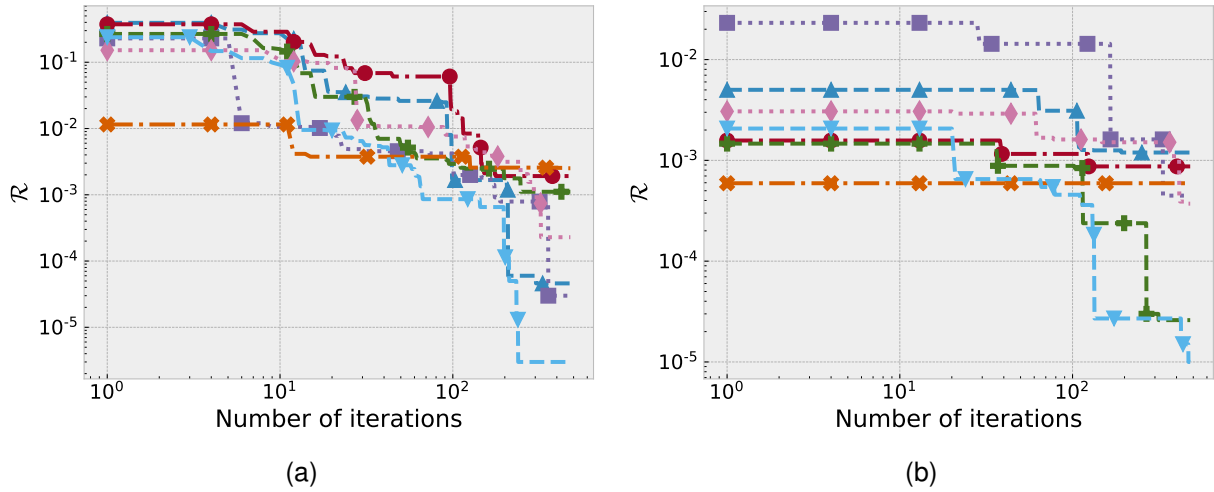


Figure 5: Progress of residual along RBFOpt iterations for the 14 problem instances related to Table 1. (a) 7 instances – No data point provided at RBFOpt initialization. (b) 7 instances – 3043 data points provided at RBFOpt initialization.

We then tested the case where past simulations results are available when we begin the optimization process for the design of a new joint. In particular, the 3043 simulations carried out in the first phase were used for initializing the process in 7 new instances. The full corresponding results can be found in Appendix B; here we show in Table 1 (right) 3 representative cases. As expected, solutions get more accurate; the runtimes however seem to double, even though the budget of function evaluations (i.e., simulations) is still 500. Indeed, constructing and then optimizing a surrogate based on thousands nodes is no more computationally negligible and the runtime is no more completely dominated by COMSOL simulations. Arguably, a good compromise might consist of only using for the initialization of the algorithm a subset of all the available past observations.

Employing available information to start the optimization process have beneficial effects on the cost-quality trade-off: the surrogate function shall provide a nicer proxy for the true objective function from early iterations, so that the error might be brought sufficiently down in far fewer steps.

To analyze this aspect, we report in Figure 5 the trend of residuals as new simulations are carried out, for each test instance (note log scale for both axes). When no observation is initially fed to the RBFOpt algorithm (Figure 5a), the process naturally starts with larger errors, and acceptable solutions are only reached after a few dozen simulations, while hundreds of iterations are often necessary to reach a higher accuracy. When past data samples are used in the initialization phase, the initial residual is often well below 10^{-2} , indicating that the first proposed solution might already be acceptable; moreover, the error can be brought down below 10^{-3} with a number of iterations ranging from few tens to a hundred.

Accuracy up to 10^{-2} - 10^{-3} might be sufficient in most robotics applications. When sufficiently large amounts of data have been collected, the design problem might be possibly considered solved and no additional attempts might be required before finding a suitable solution. This issue is investigated in the next section.

3.2 Evaluation of zero-shot approaches

In this section we verify if, and eventually at what point, incremental approaches are no more necessary to find suitable configurations for WaveJoints. We verify the behavior, as the number of the available past observations grows, of two “zero-shot” approaches that do not require users to carry out additional simulations/measurements:

- the neural network approach that was originally employed by Dragusanu et al. [24];
- a nearest-neighbor greedy approach that, given a target k , returns the configuration in the available dataset with the closest corresponding values of stiffness.

We report in Figure 6 the cumulative distribution of residuals obtained by the two zero-shot approaches upon a benchmark of 25 WaveJoint design problems, given sets of available past observations of 1000, 2000 and 6000 samples respectively.

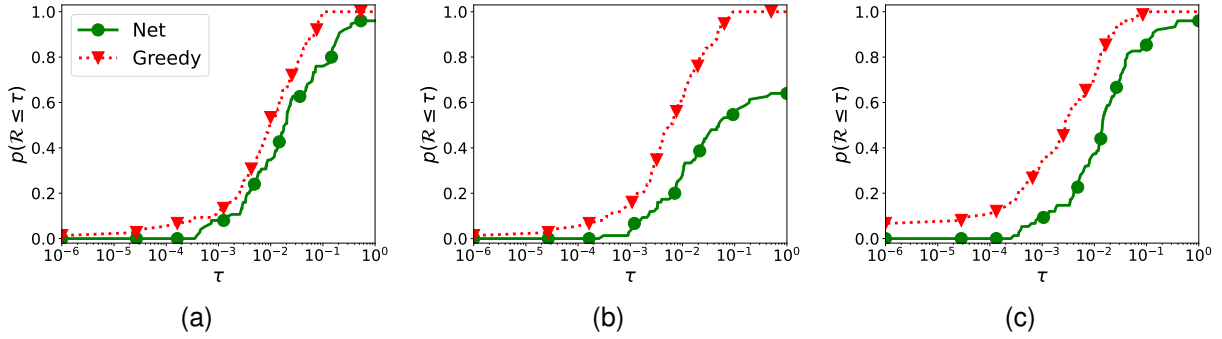


Figure 6: Cumulative distribution of residuals attained by zero-shot methods over 25 different targets, for different dataset sizes. (a) 1000 observations. (b) 2000 observations. (c) 6000 observations.

The nearest-neighbor heuristic exhibits a consistently more robust behavior than the neural network. The latter method, however, still appears to work decently if it is provided with enough training data. In absolute terms, examining the results of the nearest-neighbor heuristic, we observe that with 6000 examples in the dataset, very high levels of accuracy are frequently achieved, with very few instances where the residual values are likely unacceptable. With 2000 examples, there are fewer cases of very high accuracy levels; nevertheless, the methodology consistently yields acceptable solutions. However, with only 1000 data points, the approach appears somewhat less reliable.

We might conclude that, when a few thousands samples have been collected, the estimation problem could be considered solved employing one of the aforementioned methods. Our interest, for the rest of the Section, will thus lie in the case where few, or even zero information is available at the beginning.

3.3 Incremental approaches in fully simulated setup

We report the results of experiments carried out under two main assumptions:

- few amounts of data are available at the start of the design process;
- we work in a fully simulated setting: calculations are only done by means of the simulator, the quality of the solution is measured based on the output of the simulator; numerical errors introduced by the simulating environment w.r.t. the physical world are thus ignored.

We distinguish four scenarios, based on the number of past observations available at the initialization of the process: 25, 50, 100 and 200 samples. For each scenario, we will consider 15 problem instances. We consider the RBFOpt algorithm with a budget of 5, 10 and 20 additional function evaluations (i.e., simulations) to carry out. We also do the same for the incremental-Net approach.

First, we show the results, in the form of cumulative distribution of final residual, upon the entire benchmark of 15 problems for the RBFOpt approaches (Figure 7a) and the incremental-Net approaches (Figure 7b). In both figures, we also report the performance of the two zero-shot approaches analyzed in Section 3.2.

As expected, the incremental approaches lead to improved solutions w.r.t. the zero-shot baselines. Then, the other expected yet notable result is that providing the incremental approaches with a larger budget of function evaluations allows to improve the robustness of the method. Finally, while few tens of data points are not enough to produce high accuracy solutions, we can see that acceptable levels of residuals can be attained consistently with the incremental approaches. This aspect is the most important one, recalling that high precision is usually not needed in robotics applications, especially with simulated results.

In Figure 8, more detailed results for the four different scenarios were provided showing a direct comparison between RBFOpt and incremental-Net approaches (we only report the cases with a budget of 5 and 20 functions evaluations for the sake of clarity). The performance of the two different methodologies are comparable; there seems to be an interesting, slight advantage of the RBF optimization approach in terms of robustness, as in each of the 4 cases we can observe that the RBF curves are above their incremental-Net counterparts on the right end of the plot.

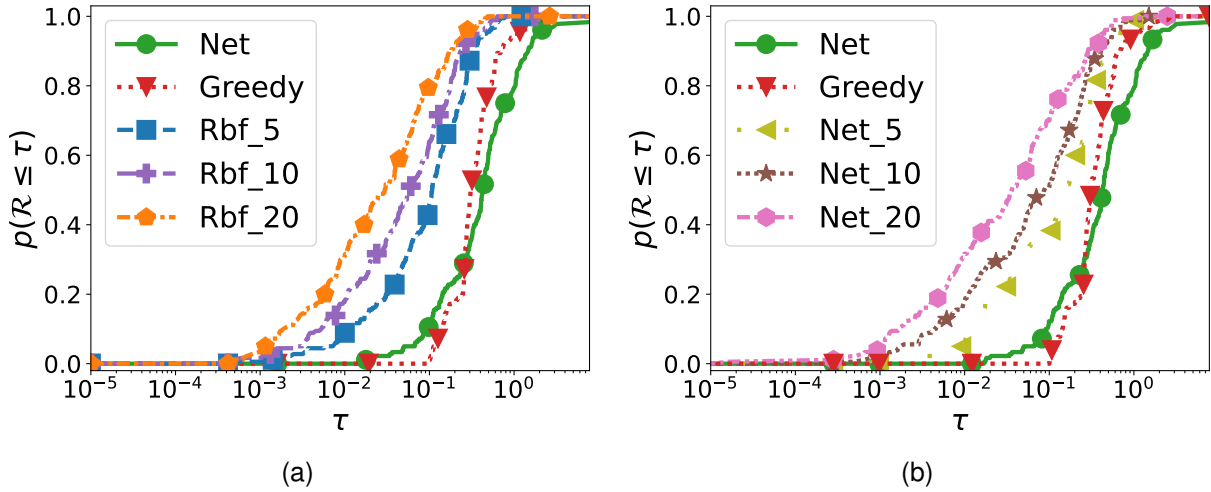


Figure 7: Cumulative distribution of residuals attained upon a benchmark of 60 instances with 25, 50, 100 and 200 data points at initialization. (a) RBFOpt vs. baselines (b) Incremental-Net vs. baselines

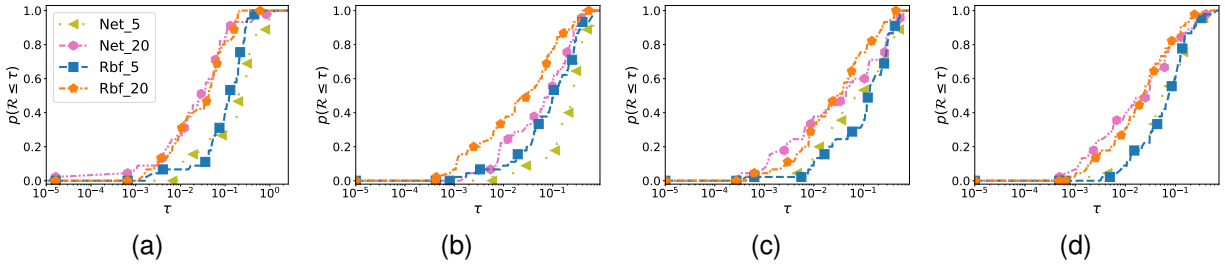


Figure 8: Cumulative distribution of residuals attained upon 4 benchmarks of 15 instances each. (a) 25 data points at initialization (b) 50 data points at initialization (c) 100 data points at initialization (d) 200 data points at initialization

3.4 Real world measures and performance of the noisy RBF approach

The results from Section 3.3 would make us confident about the proposed approach being a solid choice for the design of devices with target properties, if only simulators perfectly reproduced real-world behavior. Of course, this is not the case; we thus validate the approach on a realistic case study. We have to set the values of l_t, n_r, h_c, t, α for a WaveJoint so that the real-world measures of stiffness values will be close enough to some targets. We assume we have no prior information about the problem; in other words, we address the task starting from scratch, as it usually happens in applications.

To deal with both simulated and real-world measures obtained by 3D-printing a module and evaluating it on the test rig, we exploited the option (mentioned in 2.2) of RBFOpt for handle noise: real-world measures can be considered observations from an expensive exact oracle, whereas simulated measures can be set as observations from a cheap, yet noisy oracle. The RBFOpt solver is informed about cheap observations being noisy, with a noise cap that, based on numerous empirical tests, was set to 30% relative error.

We ran the RBFOpt based procedure to find a configuration leading to stiffness values close to $(k_\xi, k_\eta, k_\zeta) = (1238.27, 50.84, 5)$. We considered it reasonable to carry out 25 simulations between two real-world measures. After 25 new simulated, noisy observations, the algorithm asks to check the real-world measure corresponding to the best value obtained so far by the simulator; if the obtained module is considered not satisfactory from an engineering standpoint, the exact measure is added to the set of noiseless observations used by RBFOpt to build the surrogate objective function. In Table 2, we show the stiffness values (both according to the simulator and the empirical measure) of the modules that have been 3D-printed throughout the process. We can observe that the first two attempts are far from the target, because RBFOpt is still exploring parts of the space related to physically extreme configurations where there is a large gap between the simulated and the true measure. After just four attempts the proposed solution starts being

Table 2: Evolution of RBFOpt process at “milestone” iterations where a module is physically 3D-printed. The target stiffness values are $(k_\xi, k_\eta, k_\zeta) = (1238.27, 50.84, 5)$. Information for printed modules n. 3,5,6,7 can be found in the Table B.III of Appendix B.

N° Simul.	N° Real	Type	Stiffness			$\mathcal{R}$
			k_ξ	k_η	k_ζ	
25	1	Simul.	1652.89	31.07	4.63	0.268979
		Real	2683.5	47.46	11.25	2.922401
50	2	Simul.	1590.97	25.5	4.3	0.349188
		Real	1451.7	23.89	11.29	1.887757
100	4	Simul.	1641.87	30.18	5.16	0.272236
		Real	1383.7	30.43	6.21	0.232495
200	8	Simul.	1353.96	37.0	5.56	0.09524
		Real	1118.4	32.85	5.05	0.134661

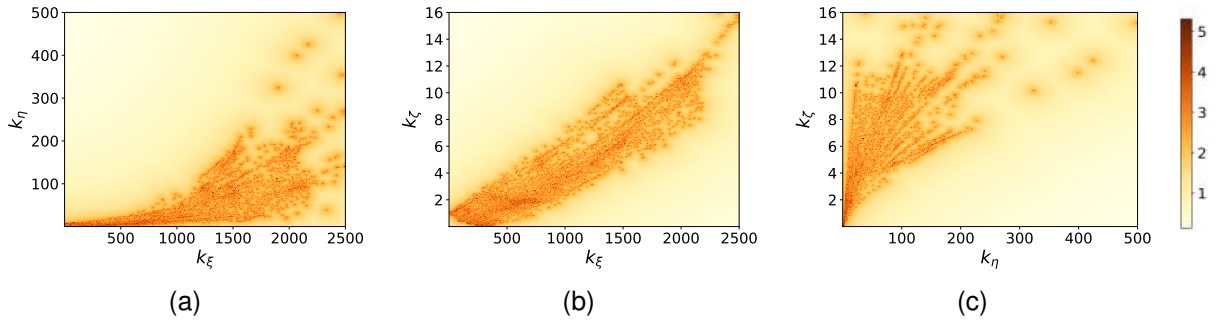


Figure 9: Heatmap showing, for each stiffness target, the quantity $\frac{1}{d_{\text{closest}}}$, with d_{closest} being the distance from the closest of the collected 18285 simulations. Each sub-figure is related to a pair of stiffness values: (a) k_ξ, k_η ; (b) k_ξ, k_ζ ; (c) k_η, k_ζ .

close to the target and the residual keeps getting reduced until $\mathcal{R} = 0.13$. At that point, we have reached a reasonably accurate solution, and arguably we cannot hope to go any further due to unavoidable imperfections both in 3D-printing and forces measures.

We finally report the result of an additional experiment, carried out to investigate the extreme case where we are willing to only build one module, at the end of the process, after a reasonable number of simulations is carried out. In other words, we test how the approach purely based on simulations performs in practice. We considered a scenario with target $(k_\xi, k_\eta, k_\zeta) = (982.36, 37.87, 5.87)$ and with a budget of 170 simulations. The final result gave simulated stiffness values of $(1037.32, 35.9, 5.53)$ - $\mathcal{R} = 0.009$ - translating to real-world stiffness measures of $(1054.2, 33.33, 5.94)$, with $\mathcal{R} = 0.01$, that are acceptable for design purposes.

3.5 Study of obtainable stiffness values

Our approach interestingly allows also to evaluate the space of stiffness values and the region of configurations that are obtainable with physically realizable WaveJoints. Using the approach proposed in this work with a target configuration that is actually unreachable, we observe that after a large number of RBFOpt iterations the error remains stable on high values, even if the parameters space has been thoroughly explored.

We collected the results of all simulations carried out plus additional runs aiming at target configurations that were never observed, so as to explore in depth the parameters space and confirm that missing target configurations are actually unreachable. With an overall collection of 18285 simulations results, we were finally able to plot (Figure 9) which parts of the three-dimensional target space are ultimately reachable. In particular, some regions where at least one of the three stiffness values is extreme are not physically obtainable; moreover, some stiffness values are obtainable, but with very precise configurations that are on the edge of physical feasibility (e.g., $(k_\xi, k_\eta, k_\zeta) = (2522.46, 139.86, 15.58)$).

4 Conclusion

In this work, we presented an optimization-based approach to identify the geometric properties of the WaveJoint that yield the desired stiffness. By modeling the task as a black-box global optimization problem and leveraging a surrogate-based strategy, we demonstrated the method's effectiveness in minimizing the number of required measurements while handling geometric constraints.

We show that the proposed algorithm identifies optimal parameters for the WaveJoint in both simulated and real-world scenarios, even with no initial data. Furthermore, as the number of measurements increases, the method transitions from iterative optimization to providing a direct mapping, enhancing its applicability in different experimental contexts.

This approach provides a flexible and efficient tool for exploring complex geometric parameter spaces, with potential applications extending beyond the WaveJoint to other soft robotic systems. Future work will focus on extending this methodology to other types of joints and exploring its integration into broader design frameworks for robotic systems.

Competing Interest

The authors have no competing interests to declare that are relevant to the content of this article.

Acknowledgements

We acknowledge the support of the European Union by the Next Generation EU project ECS00000017 "Ecosistema dell'Innovazione" Tuscany Health Ecosystem (THE, PNRR: Spoke 9 - Robotics and Automation for Health), and by the Horizon Europe project "HARIA - Human-Robot Sensorimotor Augmentation - Wearable Sensorimotor Interfaces and Super- numerary Robotic Limbs for Humans with Upper-limb Disabilities" (GA No. 101070292), and Italian Ministry of Research, under the complementary actions to the NRRP "Fit4MedRob - Fit for Medical Robotics" Grant (PNC0000007, CUP B53C22006960001).

References

- [1] R. Deimel and O. Brock, "A compliant hand based on a novel pneumatic actuator," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 2047–2053, IEEE, 2013.
- [2] M. G. Catalano, G. Grioli, E. Farnioli, A. Serio, C. Piazza, and A. Bicchi, "Adaptive synergies for the design and control of the pisa/iit soffhand," *The International Journal of Robotics Research*, vol. 33, no. 5, pp. 768–782, 2014.
- [3] N. Feng, Q. Shi, H. Wang, J. Gong, C. Liu, and Z. Lu, "A soft robotic hand: design, analysis, semg control, and experiment.," *International Journal of Advanced Manufacturing Technology*, vol. 97, 2018.
- [4] J. Shintake, V. Cacucciolo, D. Floreano, and H. Shea, "Soft robotic grippers," *Advanced Materials*, vol. 30, no. 29, p. 1707035, 2018.
- [5] A. Bicchi, "Hands for dextrous manipulation and robust grasping: a difficult road towards simplicity," *IEEE Trans. on Robotics and Automation*, vol. 16, pp. 652–662, December 2000.
- [6] D. Prattichizzo and J. Trinkle, "Grasping," in *Handbook on Robotics* (B. Siciliano and O. Kathib, eds.), pp. 671–700, Springer, 2008.
- [7] M. Pozzi, S. Marullo, G. Salvietti, J. Bimbo, M. Malvezzi, and D. Prattichizzo, "Hand closure model for planning top grasps with soft robotic hands," *The International Journal of Robotics Research*, vol. 39, no. 14, pp. 1706–1723, 2020.
- [8] I. Hussain, O. Al-Ketan, F. Renda, M. Malvezzi, D. Prattichizzo, L. Seneviratne, R. K. Abu Al-Rub, and D. Gan, "Design and prototyping soft-rigid tendon-driven modular grippers using interpenetrating phase composites materials," *The International Journal of Robotics Research*, vol. 39, no. 14, pp. 1635–1646, 2020.
- [9] T. J. Jones, E. Jambon-Puillet, J. Marthelot, and P.-T. Brun, "Bubble casting soft robotics," *Nature*, vol. 599, no. 7884, pp. 229–233, 2021.
- [10] C. Della Santina, C. Piazza, G. Grioli, M. G. Catalano, and A. Bicchi, "Toward dextrous manipulation with augmented adaptive synergies: The pisa/iit soffhand 2," *IEEE Transactions on Robotics*, vol. 34, no. 5, pp. 1141–1156, 2018.

- [11] M. Pozzi, G. Salvietti, J. Bimbo, M. Malvezzi, and D. Prattichizzo, “The closure signature: A functional approach to model underactuated compliant robotic hands,” *IEEE Robotics and Automation Letters*, vol. 3, pp. 2206–2213, July 2018.
- [12] L. U. Odhner, L. P. Jentoft, M. R. Claffee, N. Corson, Y. Tenzer, R. R. Ma, M. Buehler, R. Kohout, R. D. Howe, and A. M. Dollar, “A compliant, underactuated hand for robust manipulation,” *The International Journal of Robotics Research*, vol. 33, no. 5, pp. 736–752, 2014.
- [13] A. M. Dollar and R. D. Howe, “The highly adaptive sdm hand: Design and performance evaluation,” *The international journal of robotics research*, vol. 29, no. 5, pp. 585–597, 2010.
- [14] A. Gafer, D. Heymans, D. Prattichizzo, and G. Salvietti, “The quad-spatula gripper: A novel soft-rigid gripper for food handling,” in *Proc. IEEE Int. Conf. on Soft Robotics (RoboSoft)*, (New Haven, USA), pp. 39–45, 2020.
- [15] M. Malvezzi, Z. Iqbal, M. C. Valigi, M. Pozzi, D. Prattichizzo, and G. Salvietti, “Design of multiple wearable robotic extra fingers for human hand augmentation,” *Robotics*, vol. 8, no. 4, p. 102, 2019.
- [16] G. Salvietti, I. Hussain, M. Malvezzi, and D. Prattichizzo, “Design of the passive joints of underactuated modular soft hands for fingertip trajectory tracking,” *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2008–2015, 2017.
- [17] I. Hussain, Z. Iqbal, M. Malvezzi, D. Prattichizzo, and G. Salvietti, “How to 3d-print compliant joints with a selected stiffness for cooperative underactuated soft grippers,” in *Human-Friendly Robotics 2019. HFR 2019* (F. Ferraguti, V. Villani, L. Sabattini, and M. Bonfé, eds.), vol. 12, pp. 139–153, Springer, Cham, 2020.
- [18] H. Xu, E. Knoop, S. Coros, and M. Bächer, “Bend-it: design and fabrication of kinetic wire characters,” *ACM Transactions on Graphics (TOG)*, vol. 37, no. 6, pp. 1–15, 2018.
- [19] V. Megaro, J. Zehnder, M. Bächer, S. Coros, M. Gross, and B. Thomaszewski, “A computational design tool for compliant mechanisms,” *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–12, 2017.
- [20] R. Mutlu, C. Tawk, G. Alici, and E. Sariyildiz, “A 3d printed monolithic soft gripper with adjustable stiffness,” in *IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society*, pp. 6235–6240, IEEE, 2017.
- [21] C. Tawk, Y. Gao, R. Mutlu, and G. Alici, “Fully 3d printed monolithic soft gripper with high conformal grasping capability,” in *2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1139–1144, 2019.
- [22] R. Mutlu, G. Alici, M. in het Panhuis, and G. Spinks, “Effect of flexure hinge type on a 3d printed fully compliant prosthetic finger,” in *2015 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 790–795, IEEE, 2015.
- [23] C.-H. Liu, C.-H. Chiu, T.-L. Chen, T.-Y. Pai, Y. Chen, and M.-C. Hsu, “A soft robotic gripper module with 3d printed compliant fingers for grasping fruits,” in *2018 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 736–741, IEEE, 2018.
- [24] M. Dragusanu, G. M. Achilli, M. C. Valigi, D. Prattichizzo, M. Malvezzi, and G. Salvietti, “The wavejoints: A novel methodology to design soft-rigid grippers made by monolithic 3d printed fingers with adjustable joint stiffness,” in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 6173–6179, IEEE, 2022.
- [25] M. Dragusanu, D. Troisi, D. Prattichizzo, and M. Malvezzi, “Compliant finger joint with controlled variable stiffness based on twisted strings actuation,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7378–7384, 2023.
- [26] H.-M. Gutmann, “A radial basis function method for global optimization,” *Journal of Global Optimization*, vol. 19, no. 3, pp. 201–227, 2001.
- [27] A. Costa and G. Nannicini, “Rbfopt: an open-source library for black-box optimization with costly function evaluations,” *Mathematical Programming Computation*, vol. 10, pp. 597–629, 2018.
- [28] A. Cassioli and F. Schoen, “Global optimization of expensive black box problems with a known lower bound,” *Journal of Global Optimization*, vol. 57, pp. 177–190, 2013.
- [29] D. R. Jones, M. Schonlau, and W. J. Welch, “Efficient global optimization of expensive black-box functions,” *Journal of Global optimization*, vol. 13, pp. 455–492, 1998.
- [30] D. R. Jones, “A taxonomy of global optimization methods based on response surfaces,” *Journal of Global Optimization*, vol. 21, no. 4, pp. 345–383, 2001.

Appendix A Radial Basis Function methods for global optimization

A *radial function* $s(x)$ [26] is defined as

$$s(x) := \sum_{i=1}^h \lambda_i \sigma(\|x - \bar{x}_i\|), \quad (\text{A.1})$$

where $\lambda_1, \dots, \lambda_h \in \mathbb{R}$ are scalar coefficients, $\bar{x}_1, \dots, \bar{x}_h \in \Omega$ are the centers of the radial bases, Ω is the feasible set, $\|\cdot\|$ is the Euclidean norm and $\sigma : \mathbb{R}^+ \rightarrow \mathbb{R}$ is a univariate function usually referred to as *radial basis function* (RBF) - the Gaussian RBF $\sigma(r) = \exp(-\gamma r^2)$ is arguably the most popular one. A radial function is thus expressed as a linear combination of RBFs, each term being symmetric around its center. A very common choice in interpolation tasks is to choose the set of centers to coincide with the interpolation points.

Let $\mathcal{D} = \{(x_i, y_i) \in \Omega \times \mathbb{R} \mid i = 1, \dots, N\}$ be a set of N observations for which $y_i = f(x_i)$, $i = 1, \dots, N$. Denoting by Ξ the (symmetric) matrix whose (i, j) -th entry is $\sigma(\|x_i - x_j\|)$, $\forall i, j = 1, \dots, N$, the problem of finding the interpolant (A.1) can be formulated as a linear system:

$$\Xi \lambda = y. \quad (\text{A.2})$$

A uniquely defined radial basis interpolant exists if and only if the linear system has a unique solution in λ , i.e., if and only if matrix Ξ is invertible. Depending on the choice of the radial basis function, this condition might not be satisfied. The common way used to guarantee the existence of a solution of the system is to add a low degree polynomial to the interpolant.

Approaches based on RBFs can be used to tackle problems of the general form

$$\begin{aligned} \min_x f(x) &= \|\phi(x) - T\|^2 \\ \text{s.t. } x &\in [\ell, \mu], \quad x \in \mathbb{R}^n \times \mathbb{Z}^m, \end{aligned} \quad (\text{A.3})$$

where $\ell, \mu \in \mathbb{R}^{n+m}$ and we denote the overall feasible set by $\Omega = [\ell, \mu] \cap \mathbb{R}^n \times \mathbb{Z}^m$. Algorithm A.1 shows the general procedure adopted in global optimization based on RBFs. Usually the algorithm ends when a certain number of function evaluations has been performed or a time limit is reached, returning the best point, i.e., the observation with the lowest function value found so far.

Algorithm A.1: Global Optimization Algorithm Based on RBF surrogate

Input: Lower and upper bounds $\ell \in \mathbb{R}^{n+m}$, $\mu \in \mathbb{R}^{n+m}$, N number of initial points

- 1: Choose initial points $x_1, \dots, x_N \in \Omega$ and evaluate f obtaining $y_1, \dots, y_N$
 - 2: Let $\mathcal{D}_x = \{x_i \mid i = 1, \dots, N\}$ and $\mathcal{D}_y = \{y_i \mid i = 1, \dots, N\}$
 - 3: Set $k \leftarrow N$
 - 4: **while** *stopping criterion not satisfied* **do**
 - 5: Compute RBF surrogate $s_k(x)$ on $\mathcal{D}_x$ and $\mathcal{D}_y$ by solving system (A.2)
 - 6: Minimize $s_k(x)$ and get x_{k+1}
 - 7: Evaluate $f(x_{k+1})$ obtaining y_{k+1}
 - 8: Update $\mathcal{D}_x = \mathcal{D}_x \cup \{x_{k+1}\}$ and $\mathcal{D}_y = \mathcal{D}_y \cup \{y_{k+1}\}$
 - 9: Set $k \leftarrow k + 1$
 - 10: **end while**
 - 11: Let $x^* = \arg \min_{x \in \mathcal{D}_x} f(x)$
 - 12: **return** $x^*, f(x^*)$
-

In fact, instead of minimizing $s_k(x)$, more advanced variants of Algorithm A.1 employ merit (or acquisition) functions to be minimized for obtaining x_{k+1} . An acquisition function $m_k(x) = s_k(x) + u_k(x)$ shall calibrate the trade-off between *exploration* of the feasible region (measured by an uncertainty function $u_k(x)$) and the *exploitation* of the surrogate model $s_k(x)$ of $f(x)$.

Appendix B Extended Numerical Results

In this section, we report numerical results which did not find space in the main body of the manuscript. In particular: all the experiments for Table 1 of the manuscript are reported in Tables B.I-B.II; Table B.III contains the results of the entire experimentation performed for Table 2 of the paper.

Table B.I: Results of WaveJoints design by RBOpt (no data at initialization), according to FEM simulations.

Type	Configuration					Stiffness			$\mathcal{R}$	Time (h)
	l_t	n_r	h_t	t_h	α	k_ξ	k_η	k_ζ		
Target	15.0	3.0	8.0	0.5	0.0	990.2987435	35.2796	5.0932	5e-05	2.59
Final	15.03	3.0	8.03	0.5	0.0	994.7744533	35.1521	5.1109		
Target	15.0	4.0	12.0	0.5	0.0	1368.4751095	25.5473	3.8731	0.00192	2.63
Final	18.91	5.0	10.91	0.54	5.0	1335.141762	26.4364	3.8322		
Target	20.0	3.0	14.0	0.5	0.0	741.2807955	6.0003	1.9733	3e-05	2.63
Final	27.95	4.0	11.66	0.59	11.0	744.9876164	6.0016	1.9776		
Target	20.0	5.0	10.0	0.5	3.0	933.9547194	16.6476	2.4329	0.00111	2.56
Final	27.55	6.0	9.25	0.6	5.0	962.0831036	16.4722	2.4563		
Target	20.0	4.0	10.0	0.5	12.0	794.4973146	8.5653	2.4449	0.00251	2.5
Final	22.73	4.0	9.62	0.54	12.0	761.6266629	8.6578	2.5089		
Target	20.0	4.0	10.0	0.5	21.0	769.2113467	5.9468	2.5417	0.00023	2.6
Final	20.24	4.0	10.16	0.51	21.0	779.0096133	5.9011	2.5487		
Target	20.0	3.0	10.0	0.5	21.0	635.4721141	6.2179	2.7956	0.0	1.1
Final	27.42	3.0	10.64	0.6	20.0	635.9104309	6.2123	2.7925		

Table B.II: Results of WaveJoints design by RBOpt (3043 data points at initialization), according to FEM simulations.

Type	Configuration					Stiffness			$\mathcal{R}$	Time (h)
	l_t	n_r	h_t	t_h	α	k_ξ	k_η	k_ζ		
Target	15.0	3.0	12.0	0.5	0.0	1153.1955404	15.8598	3.7215	0.0012	5.8
Final	15.26	3.0	11.57	0.5	1.0	1121.7993543	16.1922	3.7361		
Target	20.0	4.0	10.0	0.5	0.0	803.3871624	12.4934	2.3771	0.00087	4.89
Final	27.05	5.0	9.08	0.58	5.0	818.7625301	12.6985	2.3406		
Target	25.0	3.0	10.0	0.5	0.0	448.0967673	7.1477	1.8049	0.00045	3.5
Final	28.67	4.0	8.0	0.53	10.0	456.494794	7.0798	1.8093		
Target	20.0	5.0	10.0	0.5	6.0	932.272608	14.2617	2.4533	3e-05	4.79
Final	27.86	6.0	8.99	0.6	7.0	936.2972516	14.2246	2.4512		
Target	20.0	4.0	10.0	0.5	15.0	785.8311582	7.5199	2.4704	0.00059	3.67
Final	22.91	5.0	8.64	0.52	16.0	776.9848613	7.5952	2.4232		
Target	20.0	4.0	10.0	0.5	24.0	760.010985	5.3672	2.5943	0.00037	5.0
Final	24.39	3.0	13.57	0.59	20.0	771.3502553	5.3016	2.5941		
Target	20.0	3.0	10.0	0.5	24.0	630.0396666	5.7302	2.8581	1e-05	4.61
Final	20.01	3.0	9.98	0.5	24.0	629.2406172	5.724	2.866		

Table B.III: Evolution of RBOpt process at “milestone” iterations where a module is physically 3D-printed. The target stiffness values are $(k_\xi, k_\eta, k_\zeta) = (1238.27, 50.84, 5)$.

N° Simul.	N° Real	Type	Stiffness			$\mathcal{R}$
			k_ξ	k_η	k_ζ	
25	1	Simul.	1652.89	31.07	4.63	0.268979
		Real	2683.5	47.46	11.25	2.922401
50	2	Simul.	1590.97	25.5	4.3	0.349188
		Real	1451.7	23.89	11.29	1.887757
75	3	Simul.	1641.87	30.18	5.16	0.272236
		Real	1383.7	30.43	6.21	0.232495
100	4	Simul.	1641.87	30.18	5.16	0.272236
		Real	1383.7	30.43	6.21	0.232495
125	5	Simul.	1517.04	29.58	4.96	0.225565
		Real	1581.4	28.57	5.71	0.288711
150	6	Simul.	1428.97	28.59	4.51	0.225254
		Real	1369.8	27.45	5.32	0.226888
175	7	Simul.	1507.89	42.84	5.9	0.104348
		Real	1272.0	28.57	5.5	0.202557
200	8	Simul.	1353.96	37.0	5.56	0.09524
		Real	1118.4	32.85	5.05	0.134661