Supplementary Information for Hollow-strut lattice architectures for distributed actuation and sensing in monolithic soft robots

Altair Coutinho[1,†], Trevor Exley[1,†,*], Petr Trunin[1,2], Anderson Brazil Nardin[1], and Lucia Beccai[1,*]

[1]Soft BioRobotics Perception, Istituto Italiano di Tecnologia (IIT), Genova 16163, Italy.
[2]The Open University, Affiliated Research Centre at Istituto Italiano di Tecnologia (ARC@IIT), Genova 16163, Italy
[†]These authors contributed equally. [*]e-mail: trevor.exley@iit.it, lucia.beccai@iit.it

Supplementary Notes

Supplementary Note 1: Computational Routing Pipeline

S 1.1 Voxel representation and classification

The structure is represented as a three-dimensional array of axis-aligned cubic voxels of uniform side length ℓ. Each voxel v_i is characterized by its center position $\mathbf{c}_i \in \mathbb{R}^3$, its class $\kappa_i \in \{\text{active, passive}\}$, and, for active voxels, an orientation code $o_i \in \{+x, -x, +y, -y, +z, -z\}$ identifying the axis of actuation.

The *interface classification* procedure is as follows. For each active voxel v_i with orientation code o_i, let $\hat{\mathbf{n}}_i$ be the unit vector corresponding to o_i. The two adjacent passive voxels at positions $\mathbf{c}_i \pm \ell\,\hat{\mathbf{n}}_i$ (if they exist and are passive) are labelled interface voxels. No other passive neighbours are reclassified. This ensures that only the true inlet and outlet faces of each actuator participate in the dedicated interface topology (edge-octahedron); side faces retain the fill topology and contribute to structural load paths. Fill topology is a per-design choice (BCC or FCC; see Methods).

S 1.2 Lattice graph construction

Lattice curves from Crystallon are interpreted as a spatial graph. All curve endpoints are collected and merged: two endpoints are considered coincident if their Euclidean distance is below a global node tolerance ε ($\varepsilon = 0.1$ mm throughout this work, with voxel size $\ell = 25$ mm). Merged positions are assigned unique vertex indices; each input curve segment becomes an edge (u, v, w) where w is the Euclidean length of the segment. The result is an undirected weighted graph $G_0 = (V_0, E_0)$ representing the raw lattice.

The actuator bridge extends G_0 to $G = (V, E)$ by adding connector edges. For each actuator endpoint $\mathbf{a}_j$, the nearest vertex $v^* \in V_0$ is identified by an axis-aligned search. In axial mode (mode 0), search is restricted to vertices within a cone of half-angle α (default $\alpha = 30°$) centered on $\hat{\mathbf{n}}_i$; this biases connection towards the face-center node of the adjacent edge-octahedron interface voxel. In lateral mode (mode 1), search is unrestricted and finds the nearest node in any direction. A connector edge $(\mathbf{a}_j, v^*)$ of length $\|\mathbf{a}_j - v^*\|$ is added to G.

Connector candidates are screened before insertion: any candidate whose midpoint falls inside the axis-aligned bounding box (shrunk by a factor $\delta = 0.05\,\ell$ per face) of any active voxel is rejected and logged as a blocked connector. This prevents routing lines from passing through actuator bodies.

Series junctions are detected before connector generation. For each actuator endpoint $\mathbf{a}_j$, if any other actuator endpoint $\mathbf{a}_k$ ($k \neq j$) satisfies $\|\mathbf{a}_j - \mathbf{a}_k\| < \varepsilon_s$ (series tolerance, default $\varepsilon_s = 2\varepsilon$), the endpoint $\mathbf{a}_j$ is flagged as a series joint and no connector is generated from it. The actuator tube itself forms the graph connection at these nodes.

S 1.3 BFS channel routing

Given the augmented graph $G = (V, E)$, a source vertex s and a target vertex t (corresponding to the external control port positions), and an ordered sequence of actuator waypoint pairs $W = [(a_1^{\text{in}}, a_1^{\text{out}}), \ldots, (a_k^{\text{in}}, a_k^{\text{out}})]$, the router finds a set of path segments $\mathcal{P} = \{p_1, \ldots, p_{k+1}\}$ such that their concatenation forms a walk from s to t passing through all waypoints. An illustration on a planar lattice is given in Fig. S1.

The algorithm proceeds iteratively. A dynamic forbidden-edge set $F \subseteq E$ is initialised as empty. At each step i, BFS is run on the subgraph $G \setminus F$ from the current position s_i to the next waypoint w_i. BFS expands in order of path length (Dijkstra-style with unit weights) and returns the shortest path p_i if one exists. All edges in p_i are added to F before the next step begins. This is the *edge-only forbidden* rule: vertices are never removed from the search space, only the specific edges already consumed.

The traversal order of actuator waypoints is specified by the user as an ordered index list. For the crawling robot design described in the main text, the intended order is: source → actuators in body sequence (inlet to outlet faces) → target. For the sensing path, the waypoint list is the ordered sequence of sensing zone entry nodes; diameter assignment (2.0 versus 2.5 mm) is applied per segment after routing and does not affect the graph search.

If BFS fails to find a path at any step (i.e., $G \setminus F$ is disconnected between s_i and w_i), the router reports the failed segment index and the remaining unrouted waypoints. In practice, failures indicate either that the lattice has insufficient edge-redundancy for the requested routing order, or that a blocked connector has isolated a waypoint from the main graph; both are diagnosable from the `rejected_lines` and `series_info` debug outputs.

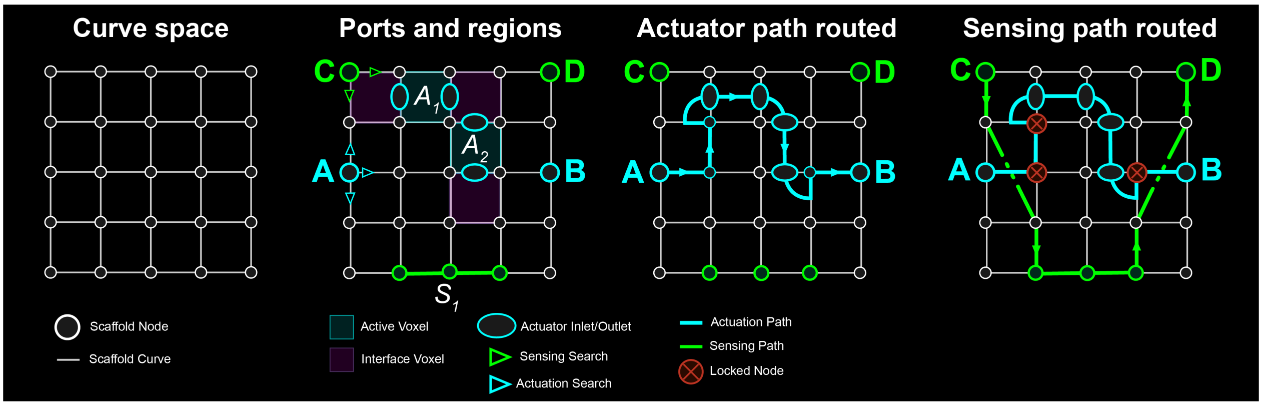

Figure S1 BFS multi-path routing pipeline illustrated on a planar lattice. (*left*) Raw curve space: scaffold nodes (open circles) and strut curves. (*second*) Port and region definition: active voxels A_1/A_2 (teal), interface voxels (purple), external ports A1, A2 (actuation) and S1, S2 (sensing), with actuator-search and sensing-search cones indicated. (*third*) Actuation path (cyan) after BFS from A1 through actuator inlets/outlets; consumed edges are added to the forbidden set. (*right*) Sensing path (green) routed on the residual graph; locked nodes (×) mark vertices whose incident edges are fully consumed, demonstrating the edge-only forbidden rule that permits node reuse via distinct struts.

S 1.4 Output and downstream use

The router outputs three curve sets that feed directly into MultiPipe (McNeel) for solid channel generation: (i) `route_curves`, the BFS-found path segments; (ii) `scaffold_curves`, the connector stubs from actuator endpoints to their lattice entry nodes; and (iii) `actuator_curves`, the actuator body centerlines. Actuation curves are assigned a single nominal inner diameter d_{ch}; sensing curves are assigned per-segment diameters (2.0 or 2.5 mm) according to the sensitization plan, producing a continuous, watertight channel network through the monolithic printed body. The resulting geometry is exported as a STEP file for slicing and SLA fabrication.

Supplementary Note 2: Hollow struts post-processing

The cleaning methodology was inspired by a previously reported procedure [1] for the post-processing of SLA-printed structures with complex internal geometries. The syringe-based approach adopted in this work provides a simple and effective method for removing uncured resin from enclosed channels prior to post-curing.

After printing, the structures were removed from the build platform and any supports were removed prior to the cleaning procedure. The external surfaces were then washed using a Form Wash system. To remove uncured resin trapped inside the hollow struts, isopropyl alcohol (IPA) was injected through the internal channels using a syringe connected to one of the inlets. The lattice was simultaneously rotated to allow gravity to assist in flushing out the resin. The flushing process was repeated until IPA flowed freely through the entire structure and no residual resin was observed exiting the opposite side. Compressed air was subsequently applied to expel any remaining IPA and residual resin from the internal channels. A final water rinse was then performed, followed by an additional compressed air step to completely empty and rapidly dry the structure. The duration of the cleaning procedure depended on the complexity of the internal channel network.

After cleaning, the samples were post-cured using a Form Cure system at 70 ℃ for 40 min. This combination of manual flushing, gravitational assistance, and compressed air enabled reliable cleaning of the hollow struts while preserving the internal geometry of the printed lattice structures. In practice, cleaning efficiency was strongly influenced by the hollow-strut diameter, with larger channels providing lower flow resistance and facilitating the removal of uncured resin through complex routing networks. Consequently, a 2.5 mm inner diameter was selected for the actuation channels to improve manufacturing reliability and reduce the likelihood of residual resin obstructing pneumatic connections between multiple actuators. The complete procedure is summarized in Fig. S2.

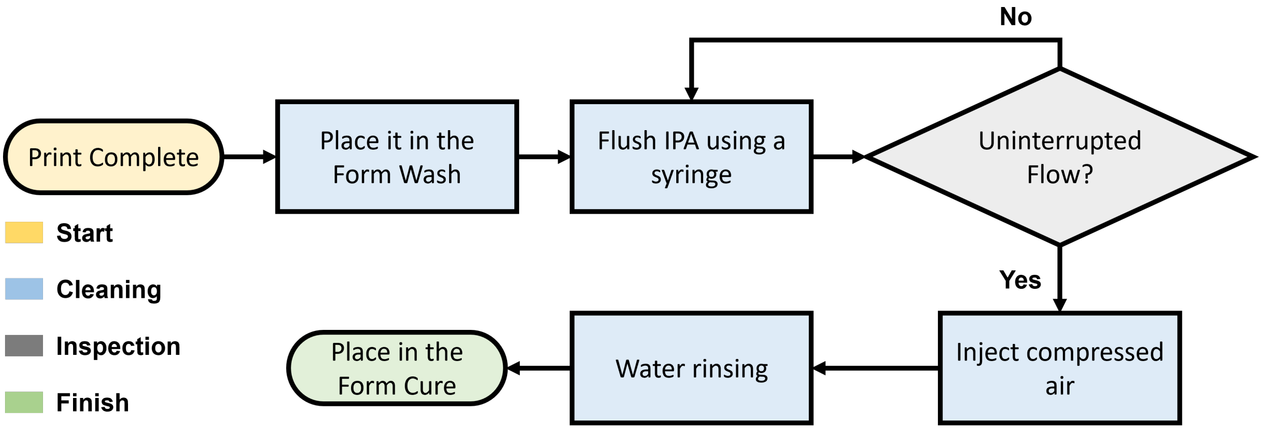

Figure S2 Post-processing workflow for the hollow struts. Rounded rectangles indicate the start and end of the workflow, blue rectangles denote cleaning operations, and the gray diamond represents the decision step used to verify complete channel cleaning.

Supplementary Note 3: Sensing characterization setup and cyclic tests

Three straight sensing-channel geometries were used for the sensing characterization, all with a constant outer diameter of 3.5 mm and hollow inner diameter d_i of 1.5, 2.0, or 2.5 mm (nominal wall thicknesses 1.0, 0.75, and 0.5 mm).

For indentation experiments (Fig. S3a), the hollow sensing channel was positioned beneath a rounded indenter mounted on a linear stage. The applied force was recorded using an inline load cell and used to regulate the indentation process, as well as for the characterization curves. The resulting pressure variation inside the channel was recorded using differential pressure sensors connected to both ends of the hollow strut. This configuration enabled synchronized acquisition of force and pressure throughout each indentation cycle, allowing direct evaluation of the pressure response generated by localized contact.

To evaluate the response under repeated loading, cyclic indentation tests were performed using the same experimental setup (Fig. S3a). The linear stage repeatedly compressed and released the sensing channel while force and pressure were continuously recorded. The resulting pressure traces (Fig. S4) compare the repeatability of the most sensitive hollow channel (2.5 mm) design under identical cyclic loading conditions.

Bending characterization was performed using the setup shown in Fig. S3b. In this configuration, the sensing channel was subjected to controlled bending deformation while the internal pressure response was recorded. This test enabled evaluation of the sensor response under distributed mechanical loading conditions, complementing the localized indentation measurements and providing insight into the behavior of the pneumatic channels under bending deformation (Fig. S5).

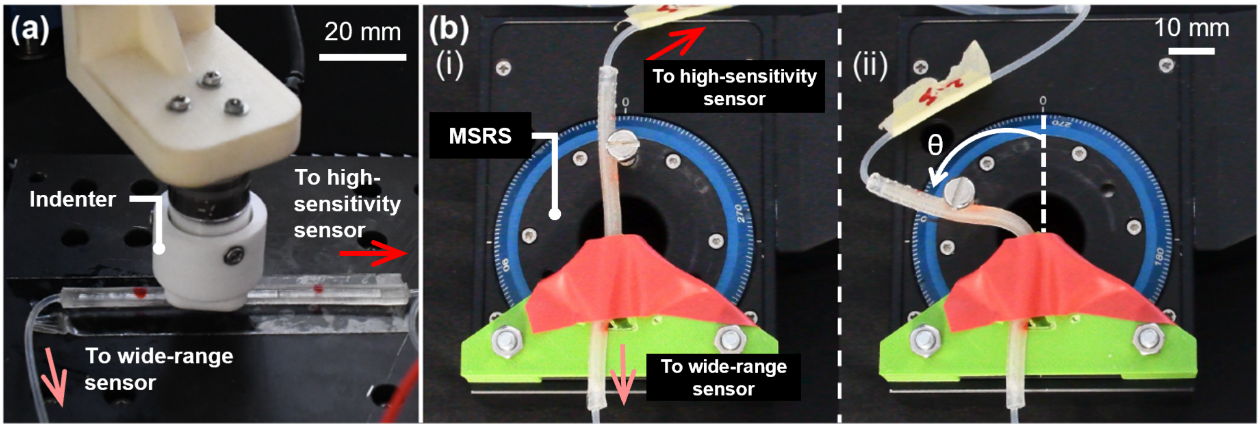

Figure S3 Benchtop characterization of the diameter-programmed sensing channel. (a) Localised indentation test: a rounded indenter mounted on a materials-testing stage compresses the channel at a sensitized segment while pressure is recorded simultaneously at both ends of the through-channel, the high-sensitivity sensor (SSC; red arrow) and the wide-range sensor (ABP; pink arrow). Scale bar, 20 mm. (b) Bending test: the channel is clamped to a micrometric servo-controlled rotary stage (MSRS) and deflected through angle θ, shown (i) in the undeformed reference state ($\theta = 0$) and (ii) bent to angle θ read from the dial, with the channel read at both ends. Scale bar, 10 mm.

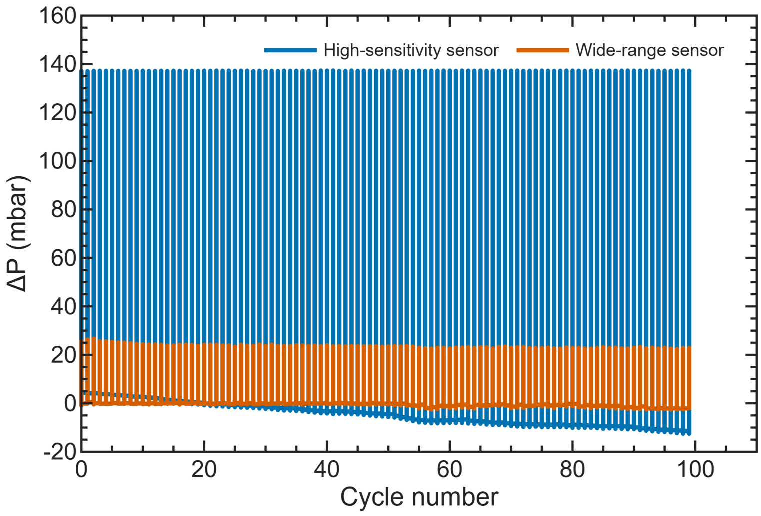

Figure S4 Repeatability of the pressure response under cyclic indentation. Peak pressure variation, ΔP, measured over 100 consecutive indentation cycles for the sensing channel using the high-sensitivity sensor (blue) and the wide-range sensor (orange), demonstrating repeatable pressure measurements under repeated localized loading.

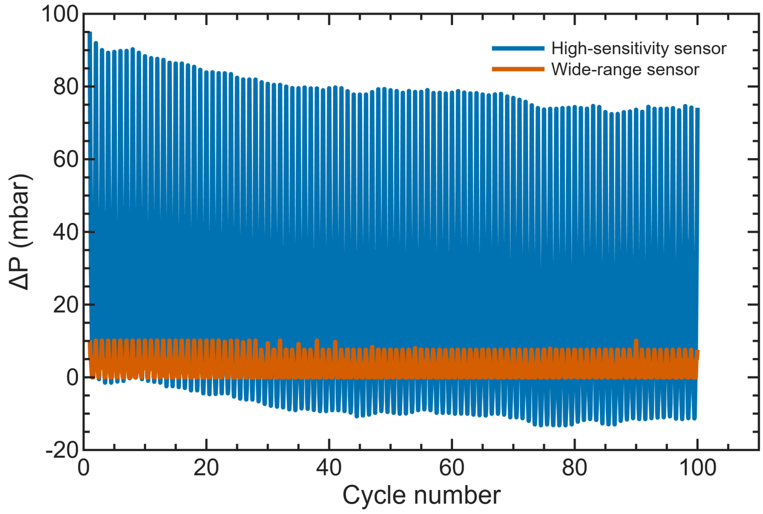

Figure S5 Repeatability of the pressure response under cyclic bending. Peak pressure variation, ΔP, recorded over 100 consecutive bending cycles for the sensing channel using the high-sensitivity sensor (blue) and the wide-range sensor (orange), demonstrating repeatable pressure responses throughout repeated bending loading.

Supplementary Note 4: Lattice-cube sensing verification and event detection

Two $2 \times 2 \times 2$ voxel specimens ($U = 25$ mm) verify that diameter-programmed sensing transfers from straight tubes to channels routed through a three-dimensional lattice (Fig. S6). Both are intended as verification rather than full characterization: the aim is to confirm that sensitized and routing segments remain distinguishable when channels follow non-trivial paths through BCC and FCC cells, and, for the active cube, while the lattice is being actuated.

The passive cube (BCC fill, no actuators) routes two independent sensing channels through the shared scaffold, each read at both ends by the high-sensitivity and wide-range sensors, with sensitized zones placed on distinct structural regions of the two channels (Fig. S6a). Three protocols were applied: bending of a sensitized region; pinching at a sensitized zone compared with an identical pinch at a non-sensitized routing segment; and a sustained pinch held on one channel while the other is repeatedly pinched, probing channel independence under simultaneous loading (see also Supplementary Movie S2).

The active cube (edge-octahedron interface, FCC fill) combines a single sensing channel, carrying one bending zone and one pinching zone, with three actuators at orthogonal orientations driven from a single shared actuation path. The sensing channel is confined to the passive scaffold subgraph, so its signal reflects lattice deformation rather than direct actuator pressure. With the actuators cycled (40 kPa to -100 kPa at 0.2 Hz), three protocols were applied: bending; pinching, as a sustained hold and as repeated taps; and combined bending and pinching during actuation (see also Supplementary Movie S3).

Event detection uses a single rule applied to both cubes and the crawler. Bending is read continuously from the high-sensitivity sensor; a pinch (tactile event) is flagged when the high-sensitivity sensor saturates and the wide-range sensor exceeds a fixed threshold (10 mbar). Because the two modes separate by amplitude, no additional signal processing is required to distinguish them.

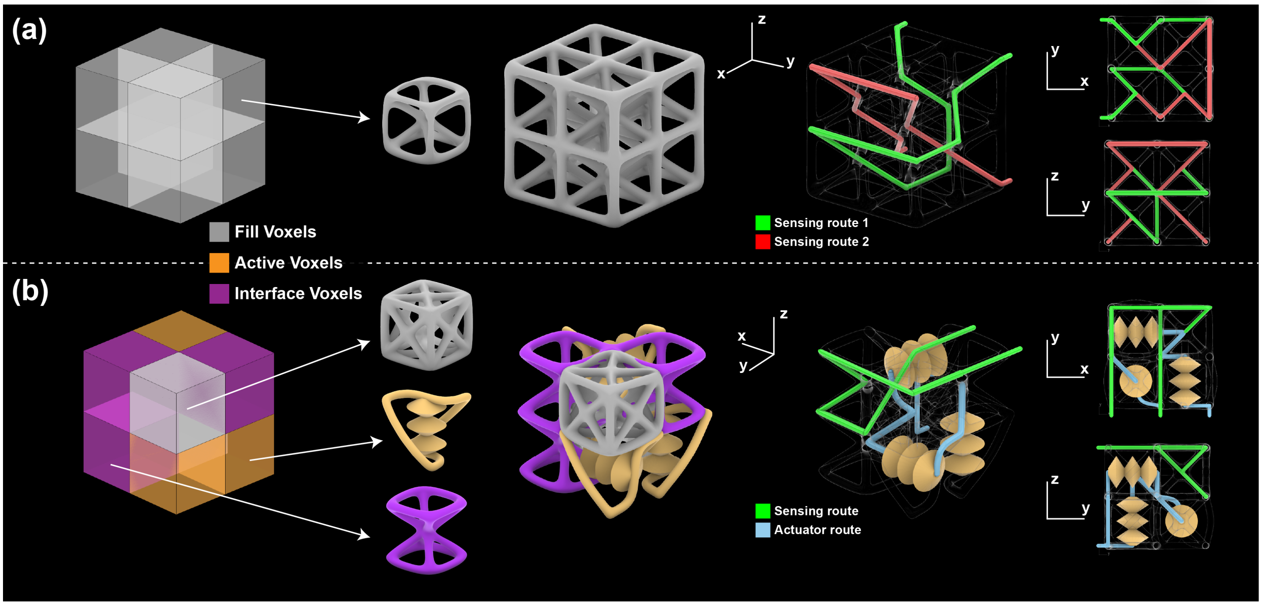

Figure S6 Routed channel paths in the $2 \times 2 \times 2$ verification cubes. (a) Passive cube (BCC fill, no actuators): voxel decomposition (left), render (center-left), and two independent sensing routes (green: channel 1, red: channel 2) in three-dimensional and orthographic projections (right); both routes terminate at the proximal face and sensitized zones are placed on distinct structural regions to enable the multi-channel verification of Fig. 3a. (b) Active cube (edge-octahedron interface, FCC fill): voxel decomposition highlighting active (orange) and interface (purple) voxels (left), render (center-left), and co-routed sensing (green) and actuation (blue) paths in three-dimensional and orthographic projections (right); the three actuators at orthogonal orientations share the single actuation path and the sensing path is confined to the passive scaffold subgraph.

Supplementary Note 5: Crawler locomotion simulation

The crawler locomotion simulation was implemented in the Simulation Open Framework Architecture (SOFA) [2, 3]. The finite-element files were generated from the design geometry using Gmsh [4]. The simulated crawler retained the full lattice architecture of the physical design, but the struts were represented as solid, non-hollow members. This approximation preserves the external geometry and actuation layout used to evaluate the locomotion mechanism, while simplifying the internal hollow-strut structure for finite-element simulation. The deformable body was modeled with a tetrahedral mechanical mesh, an implicit integration scheme, and a large-deformation tetrahedral finite-element formulation. The material was represented as an elastic solid with Young's modulus of 2 MPa and total body mass of 0.084 kg.

Three geometric representations were used in the simulation workflow (Fig. S7). The visual representation was used for rendering the crawler geometry, the collision representation defined the surface used for contact detection with the substrate, and the mechanical representation defined the tetrahedral deformable body solved during the simulation. The visual and collision representations include the external surface of the crawler, whereas the mechanical representation highlights the effective solid-strut lattice used for the deformable model.

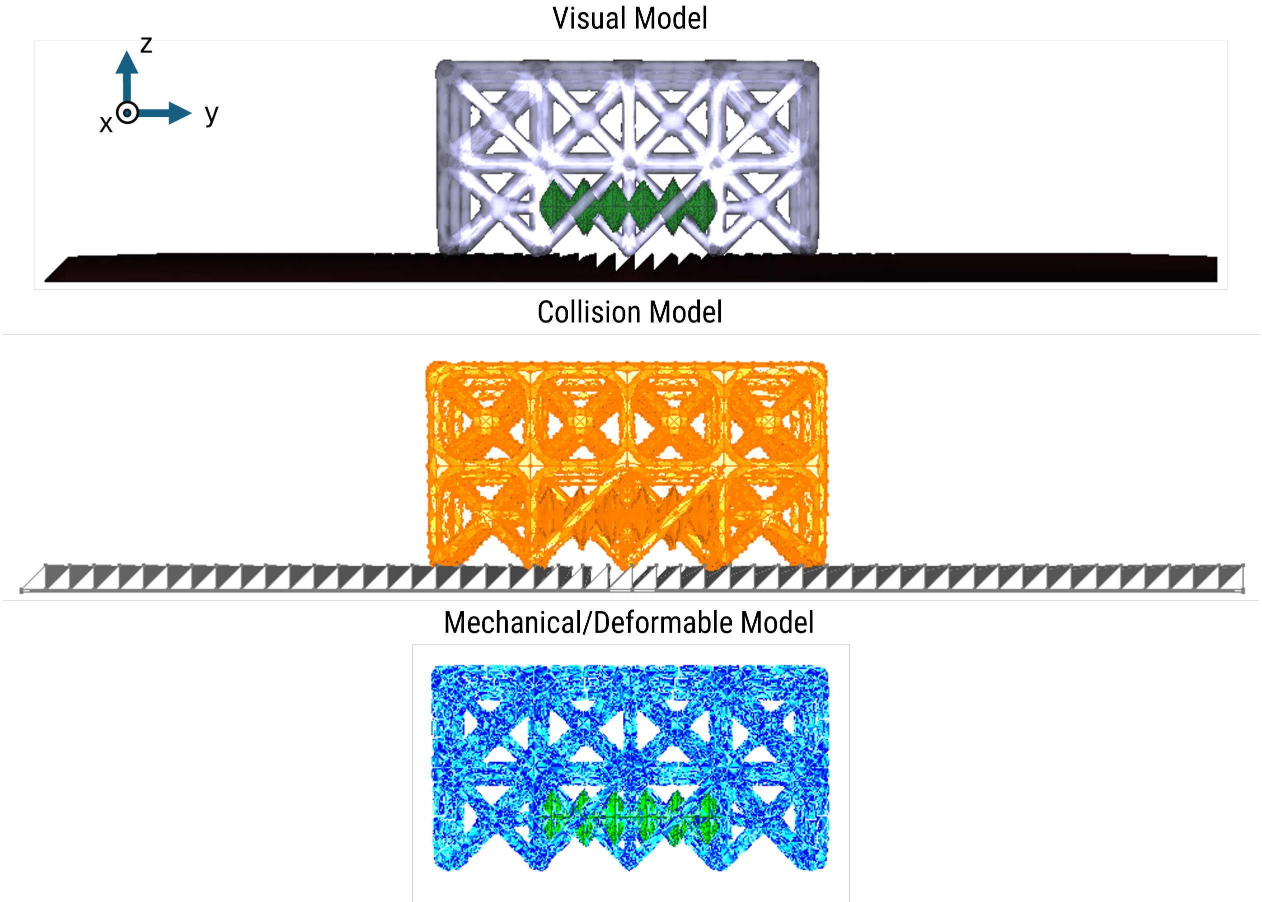

Figure S7 Simulation model representations. Visual, collision, and mechanical representations of the crawler. The mechanical model highlights the tetrahedral lattice with solid, non-hollow struts used for deformation simulation.

Pneumatic actuation was prescribed through surface pressure constraints applied to the internal chamber surfaces. The two active chambers were driven in phase, so that both sides of the crawler received the same pressure command. The input waveform spanned from positive pressure to vacuum, reaching a minimum pressure of -100 kPa and a maximum pressure of 20 kPa. The waveform included short plateaus at the pressure extrema and was repeated over multiple cycles. Contact with the sawtooth substrate was modeled using frictional contact constraints with a nominal coefficient of friction $\mu = 0.5$, chosen as a representative value for dry rubber-plastic contact and as a reasonable approximation of the material interface used in the experiments. Directional locomotion arises from the asymmetric substrate geometry rather than from anisotropic friction.

Net locomotion was quantified by tracking the y-position of the tetrahedral node closest to the initial body centroid. The signed displacement was computed as the change in y-position relative to the first recorded frame, $\Delta y(t) = y(t) - y(0)$, and plotted over repeated actuation cycles (Fig. S8). The primary validation claim is therefore that the simulated gait produces net displacement in the same direction as observed experimentally; per-cycle displacement magnitude is treated as a secondary consistency metric. Simulated and experimental net displacements are compared in Fig. S8.

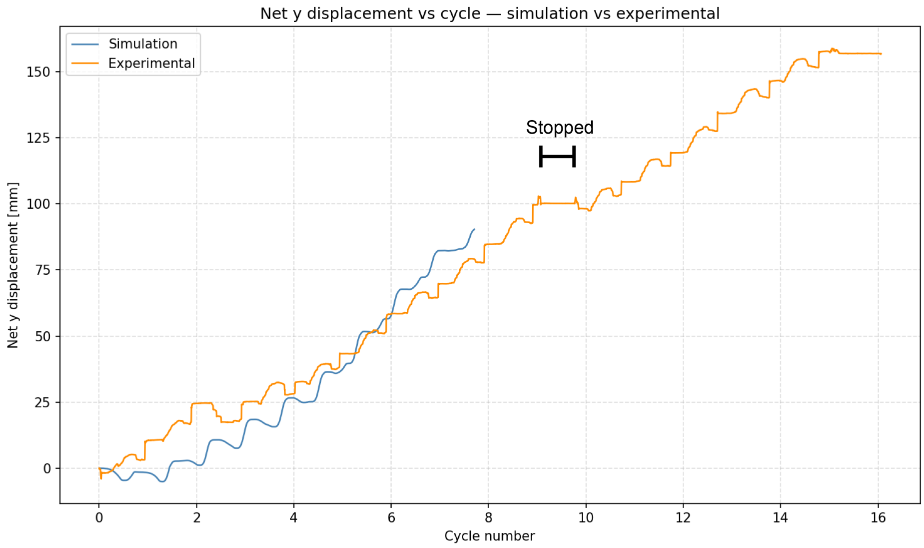

Figure S8 Comparison of net crawler displacement between simulation and experiment. Net y-displacement of the tracked centroid-proximal node over locomotion cycles for the SOFA simulation (blue) and the physical crawler (orange); both advance in the same forward direction, with the experiment reaching ≈ 158 mm over 16 cycles.

To examine the deformation associated with locomotion, a line of material points (namely track1 spine) was also tracked along the dorsal region of the crawler. These target locations were mapped to the nearest tetrahedral nodes at the beginning of the simulation, and their y-z positions were exported over time. For visualization, the tracked points were projected onto the sagittal plane and fitted at each sampled time point with a second-order polynomial curve, $z = f(y)$.

Fig. S9 shows the fitted dorsal curves sampled at equivalent phases across the first eight cycles. The selected cycle values, 1.7, 2.7, . . . , 7.7, correspond to a recurring phase of the actuation waveform chosen to emphasize the curvature obtained during vacuum application. This representation highlights the repeatability of the deformation pattern across cycles and provides a compact view of the bending behavior generated by the simulated actuation (see also Supplementary Movie S4).

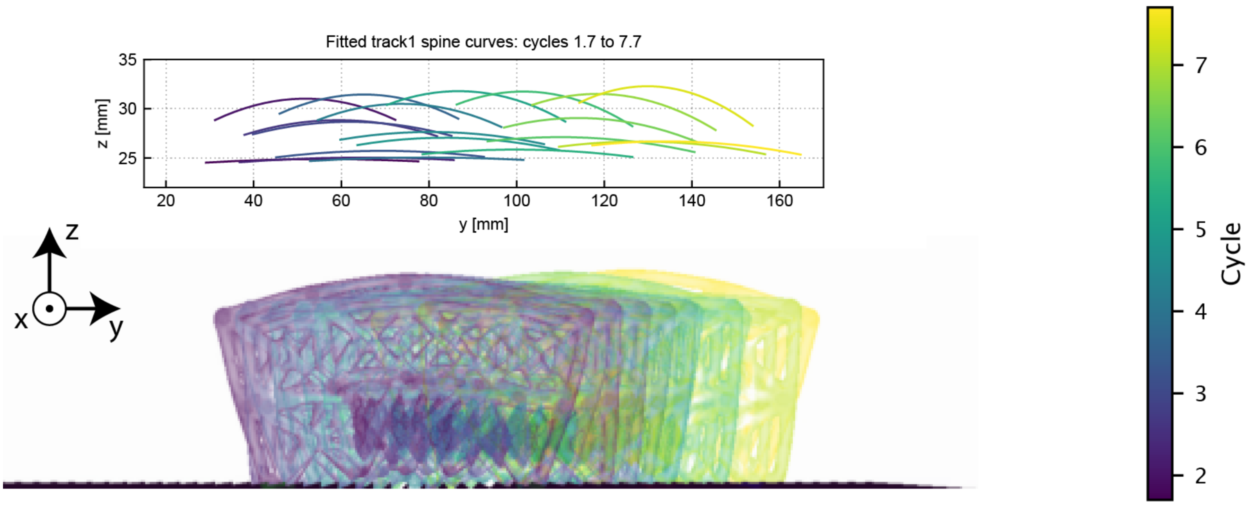

Figure S9 Dorsal bending curves sampled across locomotion cycles. Fitted y-z curves obtained from the tracked dorsal material points at equivalent phases across successive cycles.

Fig. S10 shows the same fitted dorsal curve over a single representative actuation cycle, from cycle 5.0 to cycle 6.0. This view illustrates the deformation sequence within one cycle, including the development and relaxation of curvature as the pressure command evolves. Together, the centroid tracking and dorsal-curve analysis provide complementary simulation outputs: the former quantifies net locomotion, while the latter describes the bending behavior that underlies the gait.

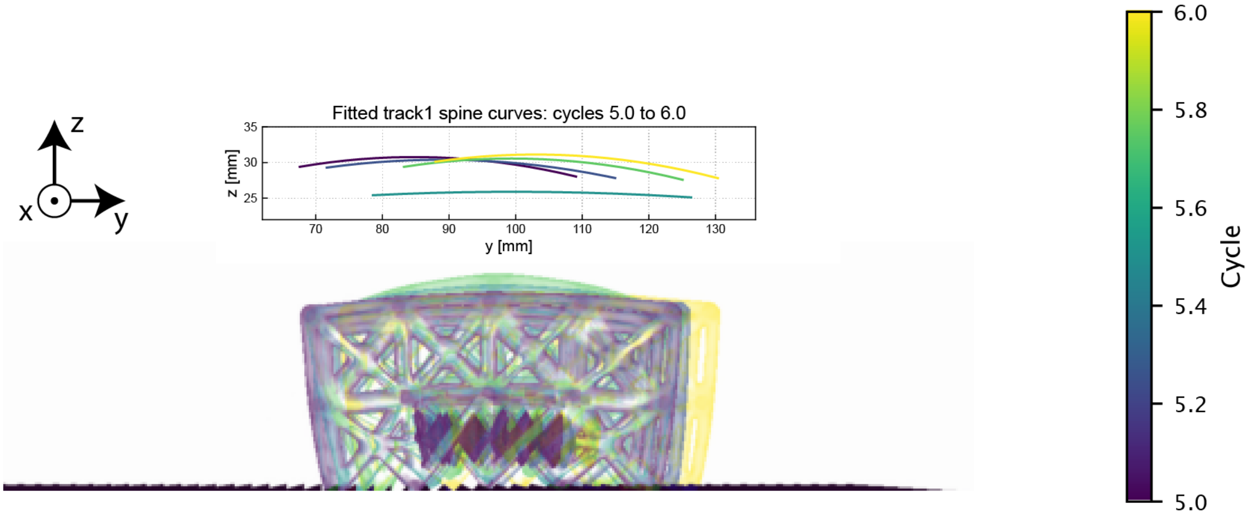

Figure S10 Dorsal bending evolution over one actuation cycle. Fitted y-z curves obtained from the tracked dorsal material points along an entire cycle.

Supplementary Movies

Movie S1. 1D Sensing Demonstration of Bending and Pinching: Demonstration of a hollow sensing channel with an inner diameter $(d_i = 2.5mm)$ undergoing repeated bending and localized pinching. The channel is first subjected to bending, followed by pinching at $0°$ and $90°$ orientations to illustrate its response under different loading conditions.

Movie S2. Passive Lattice with Distributed Sensing: Demonstration of a $2 \times 2 \times 2$ passive lattice with two embedded sensing channels. Individual and simultaneous bending and pinching are applied to the sensing channels to illustrate distributed tactile sensing, including concurrent loading and sensor saturation scenarios.

Movie S3. Active Lattice with Distributed Sensing: Demonstration of a $2 \times 2 \times 2$ active lattice integrating pneumatic actuation and distributed sensing. One hollow network is used for sensing, while the second actuates the embedded bellows. Bending and pinching experiments are performed with the actuators at rest 0 kPa and repeated during cyclic pneumatic actuation driven by a sinusoidal pressure input between +40 kPa and -100 kPa at 0.2 Hz.

Movie S4. Crawler Simulation with Tracked Backbone: SOFA simulation of the crawler undergoing ten actuation cycles. The video shows the predicted deformation, backbone evolution, and forward displacement throughout the locomotion sequence.

Movie S5. Crawler Demonstrator with Pinch Control: Experimental demonstration of the crawler under tactile control. A localized pinch command is used to start, pause, restart, and stop locomotion while the crawler is pneumatically actuated with a sinusoidal pressure input between -100 kPa and +20 kPa at 0.1 Hz. Real-time displays show both the actuation pressure signal and the sensing response used to detect bending and pinch commands.

References

[1] Benn W.B. Proper, Brandon J. Caasenbrood, and Irene A. Kuling. Easy Cleaning of 3D SLA/DLP Printed Soft Fluidic Actuators with Complex Internal Geometry. In *2023 IEEE International Conference on Soft Robotics (RoboSoft)*, pages 01–06, Singapore, Singapore, April 2023. IEEE. ISBN 979-8-3503-3222-3. doi: 10.1109/RoboSoft55895.2023.10122113.

[2] C. Duriez, F. Dubois, A. Kheddar, and C. Andriot. Realistic haptic rendering of interacting deformable objects in virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 12(1):36–47, January 2006. ISSN 1077-2626. doi: 10.1109/TVCG.2006.13.

[3] François Faure, Christian Duriez, Hervé Delingette, Jérémie Allard, Benjamin Gilles, Stéphanie Marchesseau, Hugo Talbot, Hadrien Courtecuisse, Guillaume Bousquet, Igor Peterlik, and Stéphane Cotin. SOFA: A Multi-Model Framework for Interactive Physical Simulation. In Yohan Payan, editor, *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*, volume 11, pages 283–321. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-29013-8 978-3-642-29014-5. doi: 10.1007/8415_2012_125.

[4] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, September 2009. ISSN 0029-5981, 1097-0207. doi: 10.1002/nme.2579.