Scalable Plug-and-Play Robotic Fabrics Based on Kilobot Modules

Stanley C. Obilikpa, *Graduate Student Member, IEEE*, Mohamed S. Talamali, *Member, IEEE*, Genki Miyauchi, *Member, IEEE*, John Oyekan, *Member, IEEE*, and Roderich Groß, *Senior Member, IEEE*

***Abstract*—This letter presents a framework for producing robotic fabrics using square lattice formations of interlinked Kilobot modules. The framework supports: (i) fabrics of arbitrary size and shape; (ii) different types of deformable links, namely springs and rods; (iii) easy plug-and-play reconfigurability. Two decentralized straight motion controllers are tested with robotic fabrics comprising up to 81 physical modules: an open-loop controller and a controller from the literature that responds to deformations within the fabric. For spring-based robotic fabrics, the deformation-correcting controller performs best overall, whereas for rod-based robotic fabrics, it is outperformed by the open-loop controller. A decentralized turning motion controller is formally derived and examined for either type of fabric, revealing the ability of the robotic fabrics to move along a curved trajectory using open-loop control. Finally, robotic fabrics are shown to perform basic object manipulation tasks. Robotic fabrics that deploy themselves based on distributed, embodied intelligence could pave the way for novel applications, from patching broken pipes to medical uses within the human body.**

***Index Terms*—Cellular and modular robots, distributed robot systems, soft robot materials and design, swarm robotics.**

I. Introduction

IMAGINE a medical robot that, once administered into a human body via a catheter, deploys itself by moving towards a damaged organ and stretches around it for the purpose of monitoring or stimulating it at high spatial resolution. Although no such robot currently exists, promising technologies have been proposed in a range of fields. Stretchable inorganic electronics [1], [2], for example, could provide sensing information with a high spatial resolution similar to that of human skin. However, current implementations cannot deploy themselves, as they are unable to move autonomously.

Robot swarms [3], [4] can acquire and process information in a distributed manner. However, each member of a swarm is required to locomote on its own within the environment, which could prove challenging if the members are small (e.g., sub-mm) and the environment is complex. Modular reconfigurable robots [5] offer modularity in design, and hence miniaturization potential, as exemplified by studies of programmable matter [6], [7]. As the modules are physically coupled, the locomotion requirements for individual modules are typically less demanding compared to loosely coupled robots within swarms. However, the robots are usually made of rigid components, limiting their ability to deform. Moreover, most lattice-based modular robots require individual modules to change their position within the structure for the robot to move as a whole, rendering locomotion ineffective, with notable exceptions being systems operating in fluidic environments such as air [8], [9] or water [10], [11]. Finally, soft robots [12], [13], [14] are deformable and have been extensively studied for operations within the human body. However, most designs are not modular and feature centralized components, making miniaturization challenging.

Deformable modular robot entities that couple the robots of a swarm using mechanical springs were originally proposed in [15]. A motion controller made these entities move forward while correcting locally-observed deformations within their structure. Although the underlying modules—Kilobots [16]—were highly error-prone, the modular entities could move, and the accuracy to do so increased with the number of modules therein, as validated with up to 49 modules. They could also move along curved trajectories which however required input from a global positioning system. Moreover, as this input was processed by a dedicated agent, which impacted others only implicitly through the deformations it produced, the corrective effect would diminish as the number of modules increases, compromising scalability. Furthermore, the springs were manually produced, compromising production accuracy and rendering reconfiguration tedious, as requiring the user to tie the ends of each spring to the Kilobot rings.

The contributions of this letter are as follows:

- We propose and realize a simple, plug-and-play framework for producing scalable, deformable robot entities, hereafter *robotic fabrics* (see Fig. 1), supporting (i) self-propelling

Received 27 December 2024; accepted 16 April 2025. Date of publication 8 May 2025; date of current version 29 May 2025. This article was recommended for publication by Associate Editor G. A. Sartoretti and Editor M. A. Hsieh upon evaluation of the reviewers' comments. This work was supported in part by OpenSwarm Project (EU's Horizon Europe Framework Programme under Grant 101093046, and UKRI under Grant 10048272 of the U.K. Government's Horizon Europe funding Guarantee), in part by Robotics Institute Germany BMBF under Grant 16ME1001, in part by EPSRC: DigiCORTEX under Grant EP/W014688/2, NanoMan under Grant EP/V055089/1, and Researcher in Residence Scheme under Grant EP/W037009/1 and Grant EP/X528493/1. *(Corresponding author: Roderich Groß.)*

Stanley C. Obilikpa, Mohamed S. Talamali, and Genki Miyauchi are with the School of Electrical and Electronic Engineering, University of Sheffield, S1 4DT Sheffield, U.K. (e-mail: scobilikpa1@sheffield.ac.uk; m.s.talamali@sheffield.ac.uk; g.miyauchi@sheffield.ac.uk).

John Oyekan is with the School of Electrical and Electronic Engineering, University of Sheffield, S1 4DT Sheffield, U.K., and also with the Department of Computer Science, The University of York, YO10 5GH York, U.K. (e-mail: john.oyekan@york.ac.uk).

Roderich Groß is with the School of Electrical and Electronic Engineering, University of Sheffield, S1 4DT Sheffield, U.K., and also with the Department of Computer Science, Technical University of Darmstadt, 64293 Darmstadt, Germany (e-mail: roderich.gross@tu-darmstadt.de).

This letter includes source files available at https://gitlab.com/natural-robotics-lab/robotic-fabrics, and supplementary materials available at https://doi.org/10.15131/shef.data.27021310.

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2025.3568313, provided by the authors.

Digital Object Identifier 10.1109/LRA.2025.3568313

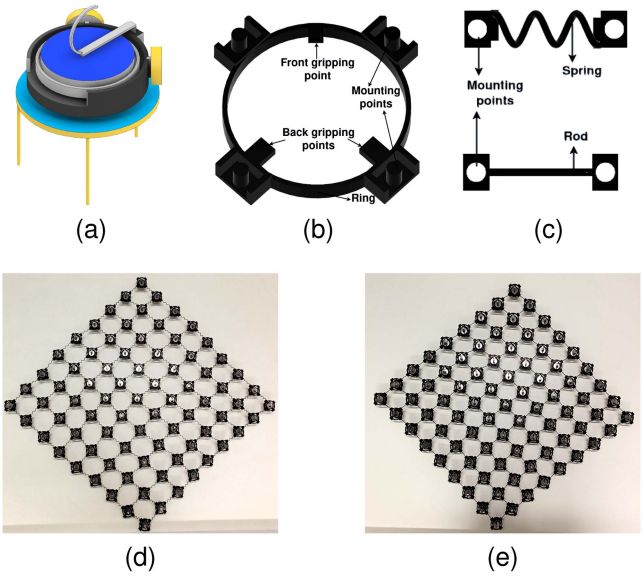

Fig. 1. Scalable plug-and-play robotic fabrics using Kilobot modules. (a) 3D model of a Kilobot; (b) Kilobot holding ring with four mounting points; (c) two types of deformable links (spring and rod); (d) 9×9 robotic fabric using spring links; (e) 9×9 robotic fabric using rod links.

entities of arbitrary size and shape, subject to resolution constraints; (ii) different types of elastic links which can be 3D printed using a range of materials; and (iii) easy plug-and-play reconfigurability.
- We report physical experiments that examine the accuracy and precision of forward motion with spring-based and rod-based robotic fabrics of up to 81 physical modules. These, to the best of our knowledge, are the biggest soft-bodied aggregates made of autonomous modules to date.
- We analyze and formally derive a novel turning motion controller which is decentralized and requires no global inputs unlike other controllers in literature. We evaluated the controller through an extensive set of 80 experimental trials, demonstrating its effectiveness and scalability in coordinating the movement of robotic fabrics without relying on a central leader.
- We investigate basic manipulation capabilities of robotic fabrics to displace various objects in the environment.

This letter is organized as follows: Section II reviews related work. Section III presents the robotic fabric, including concept, design, physical realization and motion controllers. Section IV, V and VI present experiments with physical robotic fabrics moving straight, along a curved trajectory, or manipulating objects, respectively. Section VII concludes the letter.

II. Related Work

Robotic materials [17] are of interest to scientists of many fields, including biomimetics, robotics, and micromedicine. By tightly coupling actuation, communication, computation, and/or sensing, they offer high versatility [2].

Robotic fabrics are robotic materials that resemble cloth, and hence are flexible in form [18]. Robotic fabrics have the potential to enable smart textiles and interactive surfaces [19]. For example, Buckner et al. [20] presented a robotic fabric integrated with functional fibers that sensed strain and actuated through self-rolling. Morin et al. [21] explored soft machines with microfluidic networks that change color, contrast, pattern and temperature for camouflage. Although these studies demonstrate sensing and communication capabilities, robotic fabrics typically can not self-propel and are not fully modular and reconfigurable.

To overcome these limitations, researchers considered autonomous robotic units that can assemble into modular mobile robotic entities. For example, Mondada et al. [22] considered mobile robots capable of latching onto each other. Connected robots could move together, avoid or cross gaps [23] and transport objects [24]. Murray et al. [25] equipped off-the-shelf e-puck mobile robots [26] with magnetic attachment points, allowing multiple robots to be linked and move together. Oung et al. [27] proposed the Distributed Flight Array, where propeller-based modules could link themselves on the ground, and take-off and fly as a modular entity. This concept was refined in [9] enabling individual modules to self-assemble mid-air. Doyle et al. [11] presented modular, aquatic robots that can be linked and move together by routing fluid through their combined bodies. Saintyves et al. [28] presented self-organizing robot aggregates that exhibit solid and liquid-like collective states by enabling cylindrical modules to roll over one another. Although all of these modular systems are reconfigurable and capable of collective motion, their bodies and links are essentially rigid in design.

The proposed work combines the advantages of other works in the literature by offering a simple-to-construct, flexible, plug-and-play framework for creating a range of modular robot entities that can move and adapt their shapes.

III. Robotic Fabric

In this section, we present the robotic fabric's concept, design, physical realization, and motion control algorithms.

A. Concept

The *robotic fabric* can be described as a planar graph, $G = (V, E)$. Each node $v \in \{1, 2, \ldots, |V|\}$ represents a robotic module. Within its local coordinate system, module $v \in V$ occupies a unique point $p_v = (i, j) \in \mathcal{R}$ on the diagonal square lattice, $\mathcal{R} = i\mathbf{a}_1 + j\mathbf{a}_2$ where i and j are integers, $\mathbf{a}_1 = \frac{L}{\sqrt{2}}\begin{bmatrix}1 & -1\end{bmatrix}^\top$, $\mathbf{a}_2 = \frac{L}{\sqrt{2}}\begin{bmatrix}1 & 1\end{bmatrix}^\top$, and $L > 0$. Parameter L can be interpreted as the link length.

Modules v and w are connected via a link $\{v, w\} \in E$ if and only if the pair of modules occupy adjacent points within the lattice, implying $|p_v - p_w| = L$. We assume G to be connected.

The robotic fabric can be controlled by directing the movement of each individual module. Each module is capable of executing a discrete set of motions $\mathcal{U}$. Let $u_v(t) \in \mathcal{U}$ denote the motion of the module v at time t, which is dictated by its control algorithm. Unless otherwise stated, we consider robotic fabrics of squared shape, denoted as $n \times n$. An $n \times n$ robotic fabric comprises $|V| = n^2$ modules and $2n(n-1)$ links.

Fig. 2 illustrates a 5×5 robotic fabric with 25 modules and 40 links. We assume that the modules are rigidly attached to the links. In the absence of any link deformation, a module's forward motion direction is aligned with the y-axis.

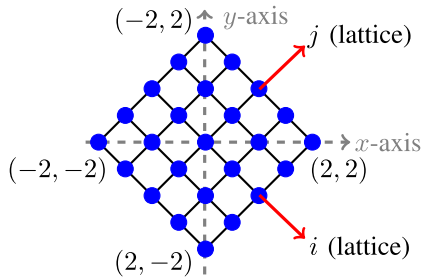

Fig. 2. Illustration of an $n \times n$ robotic fabric for $n = 5$. The modules' positions within the square lattice are denoted as (i, j) with module $(0, 0)$ being located in the center of the fabric (for odd n). Here, the robotic fabric is oriented towards the top (y-axis).

B. Design

The modules are based on the Kilobot [16], [29], a popular open-source hardware platform. The Kilobot is illustrated in Fig. 1(a). It has a diameter of 3.3 cm and weighs 17.2 g. It moves using a pair of vibration motors: When unconstrained, the Kilobot can remain immobile, move forward at the nominal speed (1 cm/s), turn left or turn right at a nominal angular speed (45 °/s), but cannot move backwards or rotate on the spot. Hence, $\mathcal{U} = \{\texttt{STOP}, \texttt{RIGHT}, \texttt{LEFT}, \texttt{FORWARD}\}$. The Kilobot can broadcast and receive infra-red (IR) messages at rates up to 30 kb/s within a local neighborhood of 10 cm. A Kilobot module comprises a holding ring, shown in Fig. 1(b) with deformable links used to connect individual modules. The requirements for the holding rings are as follows:

- The holding ring shall fit firmly on the Kilobot while not making physical contact with the vibrating motors.
- Its mass shall not exceed 2.5 g.
- When fitted on a Kilobot, it shall enable plug-and-play attachment of up to four neighboring modules via dedicated mounting points.
- The mounting point shall not allow translation or rotation of the link's attachment point relative to itself, and withstand reasonable forces experienced in a robotic fabric.
- The mounting points shall be located at angles 45°, 135°, 225°, 315°, where 0° corresponds to the module's forward orientation (see Fig. 2). This ensures that the module encounters the least force when attempting to move by a small quantity forward (see Figure S5 of Supplementary Information in [15]).

The requirements for the links are:

- The link shall ensure mechanical coupling between adjacent modules within the robotic fabric.
- The length of the link shall be such that the distance between adjacent modules (center to center) is 7 cm. This means that a module is at most about 10 cm away from any module within its Moore neighborhood, which is still in communication range.
- It shall be deformable, enabling at least one degree of freedom in motion while avoiding contact with the ground at all times.
- It shall feature attachment points at either end, enabling easy plugging and unplugging from the mounting points of the holding rings without backlash.
- For spring-type links, the wire diameter of the coil shall not exceed 0.2 cm to ease their fabrication.
- Its mass shall not exceed 1.0 g.

C. Physical Realization

Both the holding ring and links are produced using additive manufacturing techniques with CAD models designed in Autodesk Inventor. The holding ring models were sliced using CURA and 3D-printed using PLA Black material with an Ultimaker S5. The deformable link models were sliced using GrabCAD and 3D-printed using a high-fidelity Stratasys J35 Pro multi-material printer. This made it possible to digitally combine rigid and elastic materials, in our case, VeroUltraBlackS and ElasticoBlack. VeroUltraBlackS [30] has a tensile strength of 7250–9430 psi and breaks at an elongation of around 5–20%, whereas ElasticoBlack [31] has a tensile strength of 435–725 psi and breaks at an elongation of around 360–400%. Printing of all components was performed using a dissoluble support material (PVA for the S5 and WSS150 for the J35 Pro). Processing of WSS150 required treatment of wastewater with L2S powder. Printing with the J35 Pro also used small quantities of Elastico-Clear to produce a removable base layer.

In preliminary investigations, links for five digital combinations of rigid and flexible materials were produced, realizing materials with Shore A values of 50, 60, 70, 85, and 95. Links with Shore A value 95 were finally chosen due to a reasonable level of tensile strength. It should be noted that the springs mostly deform along their principle axis, whereas rods mostly deform by bending sideways.

The holding ring (see Fig. 1(b)) had a mass of 2.1 g with inner and outer diameters of 3.6 cm and 4.0 cm respectively. However, the center of each mounting point was a further 0.5 cm away from the module's center. The front and back gripping points prevented translation along the z-axis and on-the-spot rotation. As a result, the holding ring was firmly fitted on the Kilobot. Note that the front gripping point indicates the module's forward direction.

The deformable links (see Fig. 1(c)), springs and rods, had a mass of 0.3 g and 0.2 g, respectively. Both links had the same rest length, which was 2 cm when excluding the attachment points (which will coincide with the mounting points). Both links had two cuboid attachment elements with holes that match the protruding elements on the mounting points of the holding ring. The stiffness of the spring link was 0.85 N/m. This was determined with the aid of a tensile tester (Shimadzu EZ Test). The distance between adjacent modules, measured center to center with the link at rest (L), was 7 cm.

D. Motion Control Algorithms

The movement of the robotic fabric is achieved by controlling the motion of each individual module $v \in \{1, \ldots, n^2\}$, that is, by defining $u_v(t) \in \mathcal{U}$. In the following, we present three distributed algorithms for module motion control. The corresponding source code is available in the Git repository provided earlier.

1) Straight Motion Using Open-Loop Controller: The controller is executed on each module of the robotic fabric. Each module is tasked to simply move forward, formally:

$$u_v(t) = \texttt{FORWARD}, \forall t \tag{1}$$

The movements of individual Kilobots are highly error-prone. This controller tests whether mechanical coupling, in the absence of intelligent coordination like that in [15], can enable the collective to achieve coherent movement.

2) Straight Motion Using Deformation-Correcting Controller: This controller, from the literature [15], determines how each module moves based on its position in the square lattice and its distance to neighboring modules within its Moore neighborhood. To enable this, each module is assigned a unique ID in a systematic manner across the fabric, ensuring that it can identify all other modules within its neighborhood.

The motion controller lets each module move forward by default. If a module v detects a deformation within its local neighborhood, it performs corrective actions based on its position p_v in the square lattice and the distances d_v to its neighboring modules. These actions may include stopping to allow trailing neighbors to catch up, or turning to the left or right to better align with lateral neighbors. This behavior can be formally expressed as:

$$u_v(t) = f_v(p_v, d_v), \text{ with } f_v : (p_v, d_v) \to \mathcal{U} \tag{2}$$

the definition of $f_v(p_v, d_v)$ is provided in [15].

3) Turning Motion Using Open-Loop Controller: We formally derive a decentralized turning motion algorithm that is designed to be scalable. In particular, no matter how large n, the set of (distinct) speed values that the modules are required to produce is bounded.[1] By adopting a probabilistic controller, each module merely needs to decide whether to move forward at the nominal speed or remain stationary.

Consider an $n \times n$ robotic fabric as illustrated in Fig. 2, where we assume that n is odd[2] and that the distance between adjacent modules is L. At time 0, module $(0, 0)$ resides at position $[0\ 0]^T$ of the global x-y coordinate system, with the robot being oriented towards the positive y-axis (see Fig. 2). The global position of module (i, j) is given by:

$$p_{i,j} = i \begin{bmatrix} \frac{L}{\sqrt{2}} \\ -\frac{L}{\sqrt{2}} \end{bmatrix} + j \begin{bmatrix} \frac{L}{\sqrt{2}} \\ \frac{L}{\sqrt{2}} \end{bmatrix} = \frac{L}{\sqrt{2}} \begin{bmatrix} j + i \\ j - i \end{bmatrix}$$

Each module (i, j) is required to orbit around a designated reference point while maintaining a fixed distance from it.

a) Right-turning motion: We first derive a controller that enables the robotic fabric to turn right, steering towards the positive x direction in Fig. 2. Specifically, we aim for the center of module (0,0) to orbit around the point $[R\ 0]^T$ with $R > 0$, while the robotic fabric retains its formation. Over a full revolution, this module travels a total distance of $2\pi R$.

The distance of module (i, j) from the orbiting center $[R\ \ 0]^T$ at time $t = 0$ is $R_{\text{right}}(i, j) = \sqrt{(\frac{L}{\sqrt{2}}(j + i) - R)^2 + (\frac{L}{\sqrt{2}}(j - i))^2}$. To maintain a constant curvature during turning, this distance must remain unchanged. Therefore, module (i, j) travels a total distance of $2\pi R_{\text{right}}(i, j)$ per revolution.

The module $(-\frac{n-1}{2}, -\frac{n-1}{2})$ needs to travel the maximum distance, $2\pi R_{\text{right}}(-\frac{n-1}{2}, -\frac{n-1}{2})$, and thus must move the highest speed. Let the speed of this module be the nominal speed of the Kilobot, hereafter 1. To realize this speed on average, module $(-\frac{n-1}{2}, -\frac{n-1}{2})$ moves forward with the nominal speed with probability $P_{\text{right}}(-\frac{n-1}{2}, -\frac{n-1}{2}) = 1$ during each decision interval. For module (i, j), the corresponding probability is given by:

$$P_{\text{right}}(i, j) = \frac{2\pi R_{\text{right}}(i, j)}{2\pi R_{\text{right}}(-\frac{n-1}{2}, -\frac{n-1}{2})} = \frac{R_{\text{right}}(i, j)}{R_{\text{right}}(-\frac{n-1}{2}, -\frac{n-1}{2})}$$

b) Left-turning motion: Similarly, to derive a controller for left-turning motion (steering toward the negative x-axis in Fig. 2), we redefine the orbit center as $[-R\ 0]^T$. In this case, the initial distance of module (i, j) from the orbit center is $R_{\text{left}}(i, j) = \sqrt{(\frac{L}{\sqrt{2}}(j + i) + R)^2 + (\frac{L}{\sqrt{2}}(j - i))^2}$. To maintain a constant curvature, this distance must remain unchanged, meaning module (i, j) travels a total distance of $2\pi R_{\text{left}}(i, j)$ per revolution. Hence, the probability of module (i, j) moving at nominal speed is given by:

$$P_{\text{left}} = \frac{R_{\text{left}}(i, j)}{R_{\text{left}}(\frac{n-1}{2}, \frac{n-1}{2})}$$

c) Motion control implementation: To execute the turning motion, the movement of each module v at position $p_v = (i, j)$ is updated every τ time units. The module's motion is determined by:

$$u_v(t) = \begin{cases} \texttt{FORWARD}, & \text{if } X_v(t) < P_{\text{turn}}(i, j), \\ \texttt{STOP}, & \text{otherwise}, \end{cases} \tag{3}$$

where $X_v(t) \sim \text{Uniform}(0, 1)$ is a uniformly distributed random variable at time t, ensuring that the module moves forward with probability $P_{\text{turn}}(i, j)$ and stops otherwise. The function $P_{\text{turn}}(i, j)$ defines the probability of moving forward and depends on the position of the module (i, j). The probability function $P_{\text{turn}}(i, j)$ is set to $P_{\text{right}}(i, j)$ for right turning and to $P_{\text{left}}(i, j)$ for left turning. In our validations, we set the decision interval to $\tau = 0.1$ s

IV. Straight Motion Experiments

In this section, we examine the ability of robotic fabrics to consistently propel themselves in the forward direction. Given that all of actuation, sensing, computation, and communication are fully decentralized, this is a non-trivial task.

A. Experimental Setup

The experiments are facilitated using the ARK (Augmented Reality for Kilobots) setup [32]. ARK consists of a 2 m×2 m flat bounded arena made of Moonlight White S21T41, overhead cameras for tracking the positions of the robots in real time, overhead infrared transmitters for sending addressed real-time messages to individual modules, and a base control station, which is used to assign IDs.

To support post-analysis of the experiments, ARK records camera images of the arena and determines the positions of all modules, as well as the robot's center of mass (CoM) through all experimental trials. Prior to starting a new experiment, the Kilobots' vibration motors are calibrated. Moreover, the batteries are fully charged (indicated by the green LED). Where the battery indicator drops to yellow or red, the process is repeated before commencing the next trial.

At the beginning of each trial, the robotic fabric is organized as shown in Fig. 2 and placed in the environment so that it is pointing towards the x-axis of the global coordinate system. At this moment, the deformable links are at rest, resulting in a center-to-center distance of 7 cm. Where an algorithm relies on

[1] No real robot module could produce arbitrarily many speed values.

[2] The case of n being even can be derived similarly.

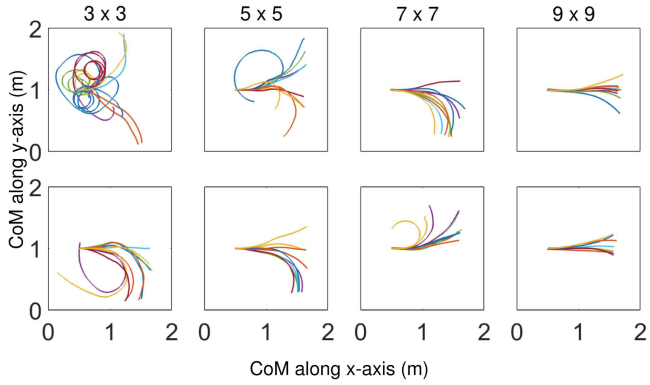

Fig. 3. Trajectories of the robotic fabric with deformable spring links. Open-loop motion controller (top), and deformation-correcting motion controller (bottom) for robotic fabrics of size 3×3, 5×5, 7×7, and 9×9.

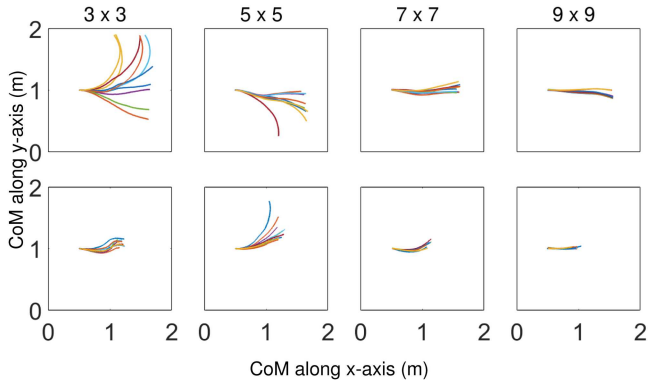

Fig. 4. Trajectories of the robotic fabric with deformable rod links. Open-loop motion controller (top), and deformation-correcting motion controller (bottom) for robotic fabrics of size 3×3, 5×5, 7×7, and 9×9.

unique IDs, ARK assigns them to each module. The objective is to reach the finish line that is 100 cm away when moving perfectly straight. To reach it, the robotic fabrics' CoM needs to move forward by 100 cm.

We report a series of experiments performed with both spring-based and rod-based robotic fabrics of dimensions 3×3, 5×5, 7×7, and 9×9. Both types of straight motion controllers were examined. For each of the $2 \cdot 4 \cdot 2 = 16$ configurations, 10 trials were performed. Hence, a total of 160 experimental trials were conducted. Trials were deemed successful, once the robotic fabric (i.e., its CoM) crossed the finish line. A trial is considered unsuccessful if the finish line isn't reached within the experimental time of 800 seconds or if the robot's fabric contacts the arena wall before the finish line, in both open-loop and deformation-correcting conditions.

B. Results

Figs. 3 and 4 depict the motion trajectories of the spring-based and rod-based robotic fabrics, respectively. Overall, the more modules within the robotic fabric, the more accurately and precisely it tends to move (see Table I).

Spring-based robotic fabrics controlled by the open-loop controller (top row of Fig. 3) did not reliably reach the target except for the largest ones, that is, 9×9 (see also left bars in Fig. 5(a)). When controlled by the deformation-correcting controller (bottom row of Fig. 3), the finish line was reached in more trials, but still only the 9×9 robotic fabrics proved reliably (see also right bars in Fig. 5(a)). Completion times remained fairly consistent across robotic fabric sizes; except for the open-loop controller performing particularly well for 9×9 robotic fabrics. It should be noted that individual springs offer little resistance to the noisy motion of individual Kilobot modules. However, for a robotic fabric of size $n \times n$, where $n > 1$, $(n-2)^2$ modules reside in the interior; therefore four springs to neighboring modules restrain their movement. The percentage of such interior modules varies substantially—from 11% to 60%—for the robotic fabrics we considered. Modules in larger robotic fabrics are therefore on average better supported and, in turn, are able to offer on average better support to their peers. In some cases, a module would topple over during a trial; however, this would not necessarily prevent the fabric from reaching the finish line. The frequency of modules toppling over was 0.98% for the open-loop controller and 0.24% for the deformation-correcting controller. For the open-loop controller, it should be noted that the robotic fabrics exhibited significant deformations.

TABLE I
ACCURACY AND PRECISION (IN MM) OF THE OPEN-LOOP (OL) AND DEFORMATION-CORRECTING (DC) CONTROLLERS AT THE TIME THE SPRING-BASED OR ROD-BASED ROBOTIC FABRICS REACHED THE FINISH LINE (SUCCESSFUL TRIALS ONLY)

		3×3	5×5	7×7	9×9
Spring	OL	-	290 / 283	-	66 / 94
	DC	191 / 113	299 / 267	260 / 114	86 / 106
Rod	OL	265 / 311	183 / 103	46 / 49	85 / 42
	DC	-	-	-	-

Accuracy is computed as the mean absolute difference between the fabric's x coordinates at start and when crossing the finish line. Precision is computed as the standard deviation of the x coordinates when crossing the finish line. Results are omitted (-) if at least half of the 10 corresponding trials are unsuccessful.

Rod-based robotic fabrics controlled by the open-loop controller (top row of Fig. 4) succeeded most often (see also left bars in Fig. 5(b)) albeit at the expense of increased completion times for the 9×9 robotic fabrics. When controlled by the deformation-correcting controller (bottom row of Fig. 4) none of the trials succeeded (see also right bars in Fig. 5(b)). This can be attributed to the reduced elasticity of the rod-shaped links as it makes it difficult for a module to correct deformations, including *perceived* deformations, which a module incorrectly detects as the result of systematic biases in its low-cost sensing hardware. For neither controller did rod-based fabrics significantly deform. In none of the trials, modules toppled over.

Fig. 6 depicts a sequence of snapshots taken from typical trials with 9×9 robotic fabrics while executing the two algorithms for the two deformable links.

V. TURNING MOTION EXPERIMENTS

A. Experimental Setup

We use the same ARK setup and position the robotic fabric as described in Section IV-A. The objective is for the robotic fabric to orbit around a point that is initially $R \in \{1, 2, 3, \infty\}$ (in m) to its right or left, in either a clockwise or anti-clockwise manner, respectively. We report experiments performed with

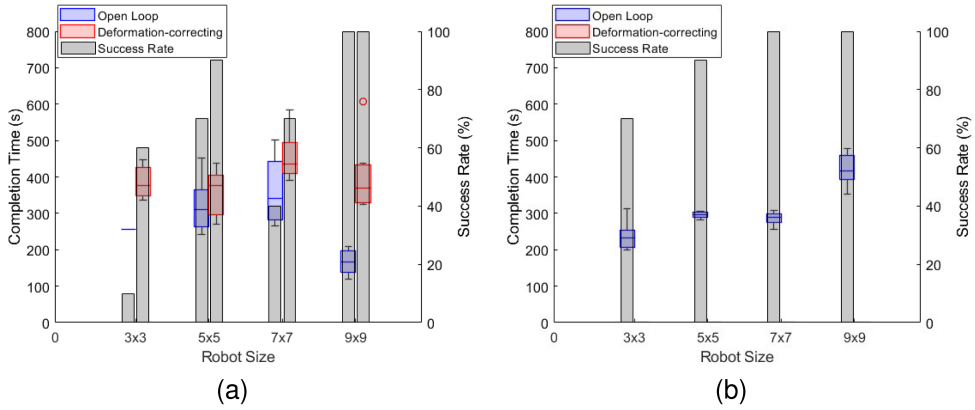

Fig. 5. Ability of robotic fabrics to move straight without external feedback. (a) Modules linked with deformable springs; (b) modules linked with deformable rods. The completion times and success rates were based on 160 experimental trials—10 trials per setting—for robotic fabrics of sizes 3×3, 5×5, 7×7, and 9×9 using the open-loop and deformation-correcting motion controllers. All trials of the rod-based fabrics with the deformation-correcting motion controller were unsuccessful in reaching the goal before the timeout, hence only a single bar and boxplot per size are shown in panel (b).

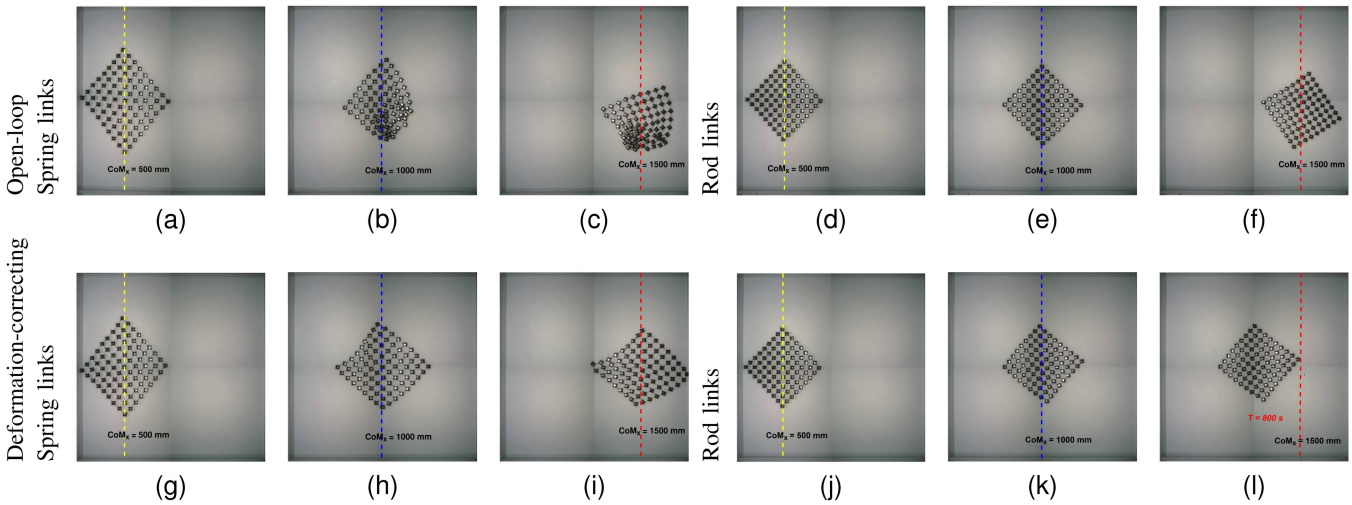

Fig. 6. Example trials with 9×9 robotic fabrics moving forward using the open-loop and deformation-correcting motion controllers. Snapshots taken once the fabrics's CoM reached 500 mm, 1000 mm and 1500 mm. (a)–(c) Open-loop motion controller with spring links; (d)–(f) open-loop motion controller with rod links; (g)–(i) deformation-correcting motion controller with spring links; (j)–(l) deformation-correcting motion controller with rod links.

both spring-based and rod-based robotic fabrics of size 7×7. For each of the $2 \cdot 8 = 16$ configurations, 5 trials were performed. Hence, a total of 80 trials were conducted. Trials continued until a robotic fabric collided with a wall or a timeout of 10 minutes was reached, whichever occurred first.

B. Results

Fig. 7 shows snapshots from experiments with $R = 3\,\text{m}$. When using the open-loop controller, rod-based robotic fabrics were better at maintaining the lattice structure and achieved a more consistent turn, albeit at a reduced average speed.

Fig. 8 shows the trajectories of all trials. We can observe that the robotic fabric successfully turns either left or right as intended and adapts its curvature according to the chosen parameter of R. We also verify that when $R = \infty$, the robotic fabric exhibits a straight motion. However, the accuracy at which it follows the orbiting arc could be improved.

Lastly, we conducted an analysis of the tracking performance for fabrics of both types (see Fig. 9), quantifying their deviation from the defined circular path. For each recorded position (i.e., once per second), the closest point on the circular trajectory was identified by projecting the robot's position onto the circle's circumference. The error was calculated as the Euclidean distance between the recorded position and its corresponding closest point. The results, summarized in Fig. 9, show the mean absolute error across various curvature radii (R). The rod-linked fabrics exhibited lower mean deviations than the spring-based fabrics, but the differences seem not significant for $R = 3\,\text{m}$ and $R = \infty$. The higher deviations observed for spring-based fabrics are likely due to the increased elasticity of their links. Nevertheless, for fabrics of both link types, the observed deviations seem acceptable given the open-loop nature of the controller, and are expected to decrease as the size of the fabric increases. For fabrics of both link types, the variance of the mean absolute error increases monotonically with R (see supplementary materials).

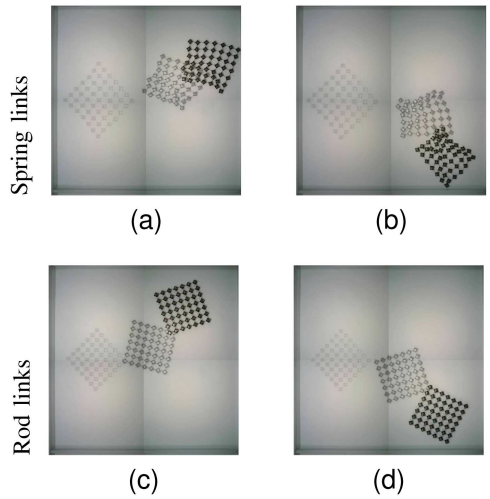

Fig. 7. Superimposed snapshots of 7×7 robotic fabrics turning left or right using a decentralized open-loop controller with rotation radius $R = 3$ m. Robotic fabrics with deformable links of type (a)–(b) spring and (c)–(d) rod.

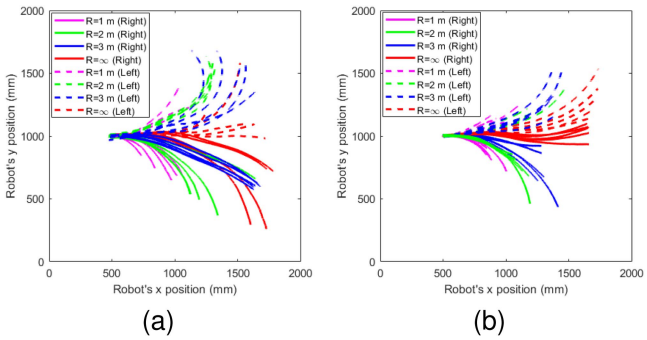

Fig. 8. Trajectories of 7×7 robotic fabrics turning left or right using (a) spring or (b) rod links. R (in m) denotes the distance of the center module from the point the fabric orbits around. Dashed and solid lines represent the trajectories of the robotic fabric turning left and right, respectively.

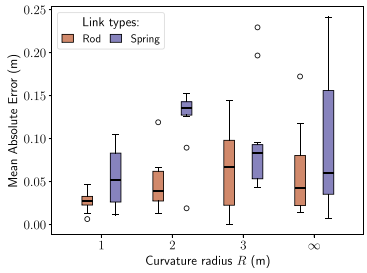

Fig. 9. Tracking errors exhibited by 7×7 robotic fabrics attempting to turn at a constant rate using the proposed probabilistic open-loop controller for four radii of curvature (i.e., 1 m, 2 m, 3 m, and ∞) for both types of links.

VI. Basic Object Manipulation Experiment

In this section, we examine a robotic fabric that collides with an object in the environment. We are particularly interested in the potential uses of robotic fabrics to grasp and/or displace the objects they encounter. The object was a 3D-printed cylinder of height 4.5 cm, radius 6.5 cm, and mass 58 g.

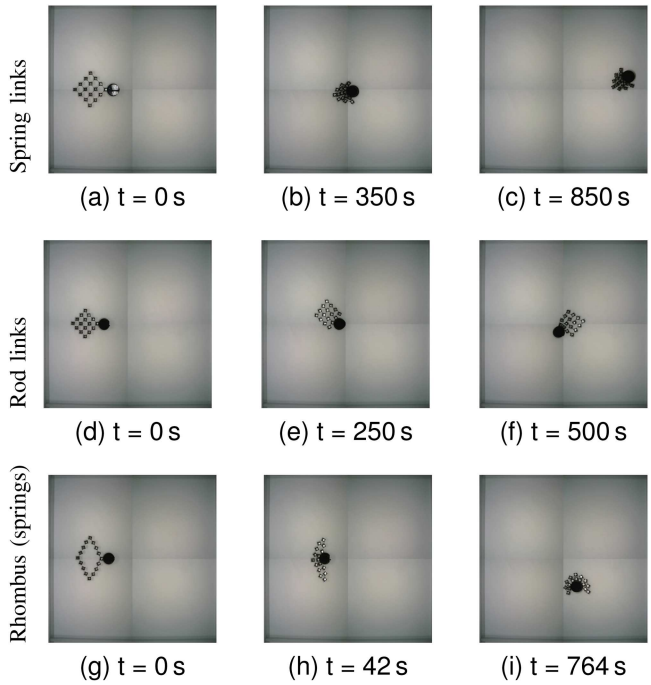

Fig. 10. Collective manipulation of a 58 g cylindrical object using a 4×4 robotic fabric with (a)–(c) spring links and (d)–(f) rod links. In (g)–(i), a hollow 5×5 robotic fabric based on spring links is shown to grasp an object.

Both spring-based and rod-based robotic fabrics were tested for 10 times. In each experiment, the robotic fabric was initially positioned with a CoM of 500 mm and was tasked with transporting the object until the CoM of the fabric reached the finish line, now placed at 1750 mm, that 125 cm away from the start. Fig. 10(a)–(f) shows two of the trials. Six trials with spring-based robotic fabrics successfully transported the object to the finish line. These fabrics tend to assume a grasping configuration as a result of colliding with the object (see Fig. 10(b)–(c)). Rod-based robotic fabrics were not unsuccessful in any trial. They typically pushed the object to about the center of the test table (i.e., 1000 mm).

Finally, Fig. 10(g)–(i) shows a hollow 5×5 robotic fabric initially placed as a rhombus. Five experimental trials were conducted. Although the object did not reach the finish line, the robotic fabric showed good promise in caging/grasping it.

VII. Conclusion

We presented a novel framework for realizing robotic fabrics based on an open-source hardware swarm robotics platform, the Kilobots. The framework makes it possible to create robotic fabrics of arbitrary size, subject to resolution constraints. We designed two types of deformable links, namely spring links and rod links, which are fabricated using additive manufacturing technologies. This allows a wide range of links to be produced, for example, by mixing digitally both rigid and elastic materials. Moreover, due to the easy plug-and-play reconfigurability, novel robotic fabrics of arbitrary shapes can be created at ease.

We implemented two decentralized control strategies to enable the forward motion of robotic fabrics from small 3×3 to large 9×9 configurations. Our results demonstrated that the larger the fabric, the more accurate and precise their motion.

We formally derived a decentralized, stochastic control strategy to enable the turning motion of fabrics. We tested the strategy with 7×7 robotic fabrics, revealing that the controller achieved its objective, despite being open-loop. It should be noted however that while the spring-based fabrics were faster, the rod-based fabrics incurred fewer deformations, which resulted in a more consistent movement. Finally, we explored the use of the robotic fabrics in basic manipulation tasks.

Our robotic fabrics concept can be generalized to other robotic systems, with feasibility depending on technological constraints. The modular design could be adapted to fit other robotic platforms such as Zooids [33], upon adjusting the link lengths and holding ring. We expect the empirical principles derived here to remain valid as long as the fundamental link properties are preserved. From a broader perspective, the concept could potentially be extended to medical applications, such as nanoscale robots navigating toward damaged tissue. This, however, is subject to technological advancements in nanofabrication, actuation, and sensing.

Future work will consider fabrics made of a combination of rigid and elastic links. It will also consider refining the control strategies to track dynamically moving targets and accomplish a wider range of object manipulation tasks.

ACKNOWLEDGMENT

The authors thank Mengguang Li for suggesting rod-based links.

REFERENCES

[1] J. Song, X. Feng, and Y. Huang, "Mechanics and thermal management of stretchable inorganic electronics," *Nat. Sci. Rev.*, vol. 3, no. 1, pp. 128–143, 2015.

[2] A. Vázquez -Guardado, Y. Yang, and J. A. Rogers, "Challenges and opportunities in flexible, stretchable and morphable bio-interfaced technologies," *Nat. Sci. Rev.*, vol. 9, no. 10, 2022, Art. no. nwac016.

[3] M. Schranz, M. Umlauft, M. Sende, and W. Elmenreich, "Swarm robotic behaviors and current applications," *Front. Robot. AI*, vol. 7, p. 36, Apr. 2020.

[4] D. Floreano and H. Lipson, "From individual robots to robot societies," *Sci. Robot.*, vol. 6, no. 56, 2021, Art. no. eabk2787.

[5] E. Bray and R. Groß, "Recent developments in self-assembling multi-robot systems," *Curr. Robot. Reports*, vol. 4, no. 4, pp. 101–116, 2023.

[6] K. Gilpin, A. Knaian, and D. Rus, "Robot pebbles: One centimeter modules for programmable matter through self-disassembly," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2010, pp. 2485–2492.

[7] Y. Peng et al., "A high-voltage generator and multiplexer for electrostatic actuation in programmable matter," *IEEE J. Solid-State Circuits*, vol. 58, no. 4, pp. 915–928, Apr. 2023.

[8] R. Oung, F. Bourgault, M. Donovan, and R. D'Andrea, "The distributed flight array," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2010, pp. 601–607.

[9] G. Li, B. Gabrich, D. Saldaña, J. Das, V. Kumar, and M. Yim, "ModQuad-Vi: A vision-based self-assembling modular quadrotor," in *Proc. Int. Conf. Robot. Automat.*, May 2019, pp. 346–352.

[10] W. Wang et al., "Roboat II: A novel autonomous surface vessel for urban environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 1740–1747.

[11] M. J. Doyle et al., "Modular fluidic propulsion robots," *IEEE Trans. Robot.*, vol. 37, no. 2, pp. 532–549, Apr. 2021.

[12] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, 2015.

[13] C. Laschi, B. Mazzolai, and M. Cianchetti, "Soft robotics: Technologies and systems pushing the boundaries of robot abilities," *Sci. Robot.*, vol. 1, no. 1, 2016, Art. no. eaah3690.

[14] G. M. Whitesides, "Soft robotics," *Angewandte Chemie Int. Ed.*, vol. 57, no. 16, pp. 4258–4273, 2018.

[15] F. Pratissoli et al., "Coherent movement of error-prone individuals through mechanical coupling," *Nature Commun.*, vol. 14, no. 1, 2023, Art. no. 4063.

[16] M. Rubenstein, A. Cornejo, and R. Nagpal, "Programmable self-assembly in a thousand-robot swarm," *Science*, vol. 345, no. 6198, pp. 795–799, 2014.

[17] M. A. McEvoy and N. Correll, "Materials that couple sensing, actuation, computation, and communication," *Science*, vol. 347, no. 6228, 2015, Art. no. 1261689.

[18] M. Yuen, A. Cherian, J. C. Case, J. Seipel, and R. K. Kramer, "Conformable actuation and sensing with robotic fabric," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 580–586.

[19] J. Qu et al., "Advanced flexible sensing technologies for soft robots," *Adv. Funct. Mater.*, vol. 34, no. 29, 2024, Art. no. 2401311.

[20] T. L. Buckner, R. A. Bilodeau, S. Y. Kim, and R. K.-Bottiglio, "Roboticizing fabric by integrating functional fibers," *Proc. Nat. Acad. Sci.*, vol. 117, no. 41, pp. 25360–25369, 2020.

[21] S. A. Morin, R. F. Shepherd, S. W. Kwok, A. A. Stokes, A. Nemiroski, and G. M. Whitesides, "Camouflage and display for soft machines," *Science*, vol. 337, no. 6096, pp. 828–832, 2012.

[22] F. Mondada et al., "SWARM-BOT: A new distributed robotic concept," *Auton. Robots*, vol. 17, pp. 193–221, 2004.

[23] V. Trianni, S. Nolfi, and M. Dorigo, "Cooperative hole avoidance in a swarm-bot," *Robot. Auton. Syst.*, vol. 54, no. 2, pp. 97–103, 2006.

[24] R. Groß, F. Mondada, and M. Dorigo, "Transport of an object by six pre-attached robots interacting via physical links," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2006, pp. 1317–1323.

[25] L. Murray, J. Timmis, and A. Tyrrell, "Modular self-assembling and self-reconfiguring e-pucks," *Swarm Intell.*, vol. 7, pp. 83–113, 2013.

[26] F. Mondada et al., "The e-puck, a robot designed for education in engineering," in *Proc. 9th Conf. Auton. Robot Syst. Competitions*, 2009, vol. 1, pp. 59–65.

[27] R. Oung and R. D'Andrea, "The distributed flight array: Design, implementation, and analysis of a modular vertical take-off and landing vehicle," *Int. J. Robot. Res.*, vol. 33, no. 3, pp. 375–400, 2014.

[28] B. Saintyves, M. Spenko, and H. M. Jaeger, "A self-organizing robotic aggregate using solid and liquid-like collective states," *Sci. Robot.*, vol. 9, no. 86, 2024, Art. no. eadh4130.

[29] M. Rubenstein, C. Ahler, and R. Nagpal, "Kilobot: A low cost scalable robot system for collective behaviors," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2012, pp. 3293–3298.

[30] Stratasys US & Canada, "Veroultra: The next level of full color modeling," 2025. [Online]. Available: https://www.stratasys.com/en/materials/materials-catalog/polyjet-materials/veroultra

[31] Stratasys US & Canada, "Elastico," 2025. [Online]. Available: https://www.stratasys.com/en/materials/materials-catalog/polyjet-materials/elastico

[32] A. Reina, A. J. Cope, E. Nikolaidis, J. A. R. Marshall, and C. Sabo, "ARK: Augmented reality for Kilobots," *IEEE Robot. Automat. Lett.*, vol. 2, no. 3, pp. 1755–1761, Jul. 2017.

[33] M. Le Goc, L. H. Kim, A. Parsaei, J.-D. Fekete, P. Dragicevic, and S. Follmer, "Zooids: Building blocks for swarm user interfaces," in *Proc. 29th Annu. Symp. User Interface Softw. Technol.*, 2016, pp. 97–109.