A Proprioceptive Method for Soft Robots Using Inertial Measurement Units

Yves J. Martin, Daniel Bruder, Robert J. Wood

***Abstract*—Proprioception, or the perception of the configuration of one's body, is challenging to achieve with soft robots due to their infinite degrees of freedom and incompatibility with most off-the-shelf sensors. This work explores the use of inertial measurement units (IMUs), sensors that output orientation with respect to the direction of gravity, to achieve soft robot proprioception. A simple method for estimating the shape of a soft continuum robot arm from IMUs mounted along the arm is presented. The approach approximates a soft arm as a serial chain of rigid links, where the orientation of each link is given by the output of an IMU or by spherical linear interpolation of the output of adjacent IMUs. In experiments conducted on a 660mm long real-world soft arm, this approach provided estimates of its end effector position with a median error of less than 10% of the arm's length. This demonstrates the potential of IMUs to serve as inexpensive off-the-shelf sensors for soft robot proprioception.**

***Index Terms*—Modeling, Control, and Learning for Soft Robots; Kinematics; Sensor Fusion**

I. Introduction

Proprioception, or the perception of the configuration of one's own body, is essential to robot planning and control. The configuration of a rigid-bodied robot can be fully described by a finite set of joint displacements which are readily measured using off-the-shelf sensors such as joint encoders. The configuration of a soft robot, however, is infinite dimensional and cannot readily be measured using off-the-shelf components.

To address this shortcoming, a number of sensing technologies have been developed specifically for soft robots [1]. Flexible resistive sensors infer strain by measuring the change in resistance of channels filled with conductive liquids such as liquid metals [2], [3] or ionic liquids [4]. Flexible capacitive sensors estimate changes in geometry by measuring changes in capacitance of stretchable electrodes separated by an elastomeric dielectric layer [5]. Optical strain sensors detect changes in geometry by measuring variations in intensity, frequency, or phase of light in a light transmission medium [6]–[9]. Magnetic strain sensors infer displacement by measuring the response of a Hall effect sensor embedded in a soft medium relative to a fixed magnetic field [10], [11]. Inductive strain sensors estimate changes in geometry by measuring inductance variations caused by transducer mechanisms such as coil geometry and mutual inductance [12]–[14]. Deformable sensing fabrics or "skins" that incorporate several soft sensing technologies into a single versatile package have also been developed [15], [16].

This work was supported by a Space Technology Research Institutes grant (number 80NSSC19K1076) from NASA's Space Technology Research Grants Program.

Yves J. Martin, Daniel Bruder, and Robert J. Wood are with the John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, 02138 USA (e-mail: yves@martin.yt, dbruder@seas.harvard.edu, rjwood@seas.harvard.edu).

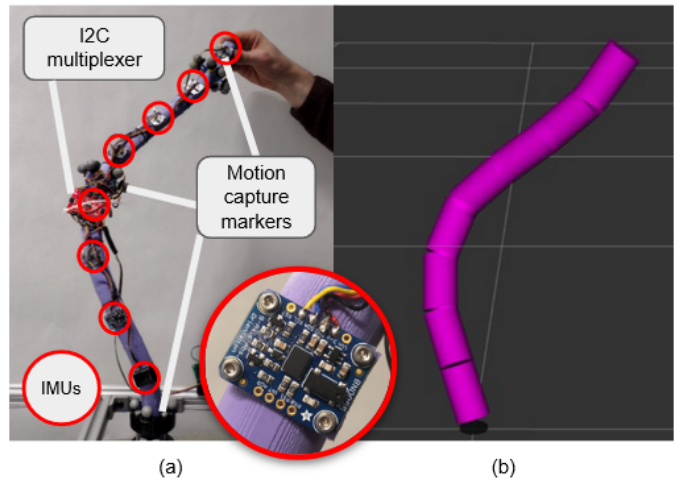

Fig. 1: (a) The soft arm platform, with IMUs labeled, used to evaluate our estimation methods. (b) A serial-chain rigid-body model approximation of the soft arm motion is shown.

Each of these sensor types measure the strain of a flexible element. Using strain measurements to construct an estimate of the pose of a soft arm requires integrating strain along the length, imposing accuracy limitations since small inaccuracies in strain along the length will compound into much larger pose errors. Furthermore, while each sensor type has its own unique pros and cons, a common issue among them is nonlinear time-variant behavior and hysteresis. This has motivated the use of machine learning techniques to identify the mapping from raw sensor output to deformation from data [17]–[19].

An alternative to embedded sensors is an external motion capture system. Such systems utilize an array of externally-mounted infrared cameras to track the position of reflective markers in 3D space. By coating a soft robot in reflective markers one can use motion capture data to estimate the robot's shape. Commercial motion capture systems offer an accurate and reliable method for sensing the deformation of soft robots, but they are expensive and impose severe restrictions on the environment in which a robot can operate. Motion capture systems are not portable, are sensitive to lighting conditions, and are susceptible to errors due to occlusions, making it impossible to use them outside of a controlled laboratory setting.

This paper explores the use of Inertial Measurement Units (IMUs) for sensing the shape of continuum soft robots. IMUs are off-the-shelf orientation sensors that combine a three-axis accelerometer, three-axis gyroscope, and three-axis magnetometer with an on-board sensor fusion algorithm. Due to their extensive use in smartphones, tablets, and wearable fitness trackers, IMUs have become widely available and inexpensive, and there are many well-developed computational resources for integrating them into robotic systems [20].

Previous work has explored the use of IMUs for proprioception in soft robots. In [21], two IMUs were used in combination with motor encoders to estimate the pose of a single segment continuum body manipulator assuming constant curvature. In [22] two IMUs were embedded into the ends of an elastomeric liquid metal strain sensor to form a hybrid sensor capable of measuring the angle of deflection of a soft bending actuator and the joint position of a rigid robot arm. In both of these cases, IMUs were combined with other sensor types (i.e. encoders, strain sensors) to estimate the pose of a one segment soft structure.

The contribution of this paper is a simple method for estimating the shape of a full continuum arm using only IMUs. The approach approximates the infinite-dimensional shape of a continuum arm with a finite-dimensional rigid-body model, and does not require any prior training. In real-world validation experiments, it is shown to estimate end-effector position to within 10% of arm length of median error, putting its accuracy on par with that of other soft sensors while avoiding the limitations imposed by external motion capture systems.

The remainder of this paper is organized as follows. Section II describes how IMU data is utilized to construct an estimate of the shape of a robot arm. Section III describes how the performance of the IMU-based sensing approach was evaluated on a real-world soft arm platform. Section IV presents the results of these experiments. Section V discusses the results, likely sources of error, and potential improvements. Section VI offers concluding remarks and proposes several avenues for future work.

II. Methods

IMU measurements provide the arm's orientation at given spots along the arm. This section will discuss how to use this data to estimate the full arm shape using two different arm parametrizations. For both of the presented models, the arm is parameterized as a series of rigid segments.

In part II-A, one segment per IMU is used. In part II-B, we use spherical linear interpolation to create "virtual IMUs" to increase the number of segments without increasing the number of IMUs. In the limit, this method approximates a Piecewise Constant Curvature model (PCC) [23]. Both models are constructed from IMU orientation data in the form of quaternions, which are singularity free, as opposed to Euler angles which suffer from the Gimbal Lock issue [24]. Quaternions are also broadly used in computer science and as a default IMU output format.

The models specify the shape of the arm as a collection of joint coordinates. However, because both models represent the arm as a series of rigid segments, it is straightforward to compute the coordinates of any point along the length of the arm using linear interpolation between two adjacent joints.

A. Rigid-body model

The arm is modeled by $N+1$ straight links of length L_i^R, $i \in \{0, ..., N\}$, called segments (see Fig. 2(a)). An IMU is fixed to each segment, ideally on the center. The ith segment is oriented along the IMU frame defined by its quaternion $\mathbf{q_i}$, $i \in \{0, ..., N\}$ where $\mathbf{q_0}$ is the orientation of the arm's base. We define $R_q(\mathbf{q})$ as the 3x3 rotation matrix extracted from a quaternion $\mathbf{q} = a + ib + jc + kd$ [25],

$$R_q(\mathbf{q}) = \begin{pmatrix} 1-2(c^2+d^2) & 2(bc-ad) & 2(bd+ac) \\ 2(bc+ad) & 1-2(b^2+d^2) & 2(cd-ab) \\ 2(bd-ac) & 2(cd+ab) & 1-2(b^2+c^2) \end{pmatrix} \tag{1}$$

Let $\hat{\mathbf{x}}_\mathbf{i}^\mathbf{R}$ be an estimate of the position of the joint between segment i and segment $(i-1)$ according to this model. It is computed by summing the displacements of the first $i-1$ links, according to the following expression,

$$\hat{\mathbf{x}}_\mathbf{i}^\mathbf{R} = \sum_{j=0}^{i-1} R_q(\mathbf{q_j}) \begin{pmatrix} 0 \\ 0 \\ L_j^R \end{pmatrix} \tag{2}$$

Note that the superscript $(\cdot)^R$ denotes a variable related to the rigid-body model.

B. PCC-extended rigid-body model

To make the rigid-body model smoother and more accurate, we divide each segment into n sub-segments. Each sub-segment orientation is computed under the assumption that the segment's curvature is constant. As n increases, this model approximates a PCC model.

Let us define L_i^E as the distance between $\mathbf{q_i}$ and $\mathbf{q_{i+1}}$, and $\mathbf{q_{i,k}}$ as the orientation quaternion of the kth sub-segment of the ith segment (see Fig. 2(b)), which is computed using spherical linear interpolation between $\mathbf{q_i}$ and $\mathbf{q_{i+1}}$.

$$\mathbf{q_{i,k}} = slerp\left(\mathbf{q_i}, \mathbf{q_{i+1}}, \frac{2k+1}{2n}\right) \tag{3}$$

The function $slerp$ represents spherical linear interpolation [26] and is defined as,

$$slerp(\mathbf{q_a}, \mathbf{q_b}, t) = (\mathbf{q_b}\mathbf{q_a}^{-1})^t \mathbf{q_a} \tag{4}$$

where $t \in [0, 1]$ is the interpolation variable such that $slerp(\mathbf{q_a}, \mathbf{q_b}, 0) = \mathbf{q_a}$ and $slerp(\mathbf{q_a}, \mathbf{q_b}, 1) = \mathbf{q_b}$.

Next, the position of the kth joint of the ith segment, denoted $\hat{\mathbf{x}}_\mathbf{i,k}^\mathbf{E}$, is computed by summing the displacement of all preceding joints according to the following expression,

$$\hat{\mathbf{x}}_\mathbf{i,k}^\mathbf{E} = \hat{\mathbf{x}}_\mathbf{i,0}^\mathbf{E} + \sum_{m=0}^{k-1} R_q(\mathbf{q_{i,m}}) \begin{pmatrix} 0 \\ 0 \\ L_i^E/n \end{pmatrix} \tag{5}$$

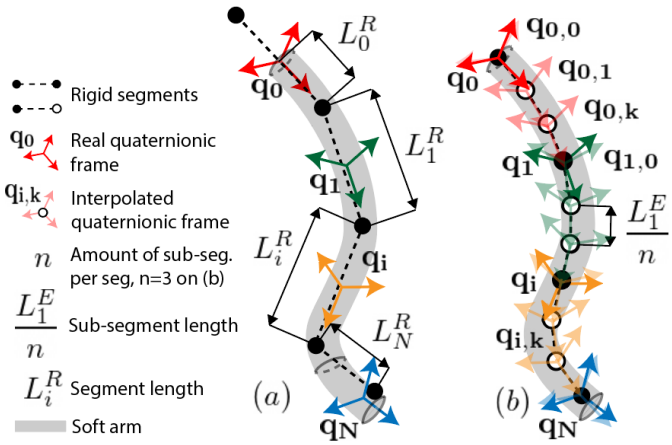

Fig. 2: Different models to approximate the state of a soft continuum arm using IMUs: (a) rigid-body model, (b) and PCC-extended rigid-body model.

Note that the superscript $(\cdot)^E$ denotes a variable related to the PCC-extended rigid-body model.

We define the 0th joint of the 0th segment to be located at the origin,

$$\hat{\mathbf{x}}^{\mathbf{E}}_{\mathbf{0,0}} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{6}$$

and since the nth joint of segment i is the same as the 0th joint of segment $i+1$ the following are equivalent,

$$\hat{\mathbf{x}}^{\mathbf{E}}_{\mathbf{i+1,0}} = \hat{\mathbf{x}}^{\mathbf{E}}_{\mathbf{i,n}} \tag{7}$$

III. Experiments

A. Description of continuum arm

To validate the IMU-based modeling approach presented in Section II, we constructed a soft continuum arm and outfitted it with IMU sensors. The arm is a cylinder of silicon (Smooth Sil 945) with a diameter of 25mm and of length 660mm. The structure is purely passive, it does not have any actuators. Eight IMUs (bno055 Bosch) were evenly spaced along the length of the arm and fixed in place with screws into flat slots on the arm. We selected this geometry to make the arm as flexible as possible while accommodating the 25x20mm footprint of the IMU breakout boards and allowing them to be spaced 80-100mm apart. Empirically, this spacing permits a maximum angle of $\pi/2$ between adjacent IMUs. The update frequency of the IMUs is 31Hz, and they are read by an Arduino Mega 2560 using an I2C channel and an I2C multiplexer. Two motion capture markers were fixed to the arm at lengths $M_1 = 360$mm and $M_2 = 600$mm. There was only two motion capture markers because when we tried to use a higher density of markers, the motion capture system would mix up the markers. The motion capture system (Vicon) has a precision of 1mm and an update frequency of 100Hz.

B. Validation experiments: Motion capture vs. IMU model

To evaluate the accuracy of our models, motion capture is used as ground truth. While the arm is moved, measurements from the IMUs and the motion capture system are recorded. The IMU data is then used to construct models according to Section II offline. The positions of the motion capture markers are estimated from the models and compared to the actual positions recorded by the motion capture system. In all experiments, error is defined as the Cartesian distance between the estimated positions of the motion capture markers and their actual positions. To evaluate how model performance is affected by the density of IMUs along the arm, both models are constructed using the data from two, four, and eight IMUs.

1) Rigid-body model experiment: The rigid-body model is implemented using two, four, and eight IMUs (and thus with an equivalent number of segments). The position of the IMUs along the arm are shown in Table I. We set $L_0 = 0$mm since the first segment of length L_0 (see Fig 2) empirically induces a vertical error offset. When using two and four IMUs, the length of the arm can be divided into two and four equal sections, respectively, with an IMU mounted to the midpoint of each section. For eight IMUs, when the length of the arm is divided into 8 equal sections, the segmentation is made such that the IMUs lie at the endpoints of the sections instead of the center due to their physical spacing.

2) PCC-extended rigid-body model experiment: The PCC-extended rigid-body model is tested w.r.t. two variables: the number of segments, and the number of IMUs. First, we fixed the number of IMUs to the maximum, i.e. eight IMUs, and used 8, 16, 32, and 64 segments. Results shown in Fig. 4 suggests that 16 segments are sufficient to approximate the maximum curvature exhibited by our arm. To explore the impact of the number of IMUs on pose estimation accuracy, we generated a 16 segments model using data from two, four, and eight IMUs. The position of the IMUs used along the arm are shown in Table I. Results shown in Fig. 5 indicate that even for the same number of segments (i.e., 16), the error decreases as the number of IMUs increases.

IMU #	Base*	1	2	3	4	5	6	7	8
pos [mm]	0	80	160	240	320	400	480	560	640
2 IMUs	E		R		E		R		E
4 IMUs	E		X		X		X		X
8 IMUs	E	X	X	X	X	X	X	X	X

TABLE I: This table shows which IMUs are selected for a given number of IMUs used to feed each model. Pos corresponds to the position of the IMU along the arm. "R" stands for "rigid-body only", "E" stands for "PCC-extended rigid-body model only", and "X" stands for both.
*base is the orientation of the arm's base, not counted as an IMU but used in the PCC-extended rigid-body model.

3) Timing synchronization: To ensure the same sampling frequency on both the IMU model and motion capture signals, linear interpolation across time is processed on the model output. The time delay between both signals is computed using cross-correlation and removed.

Marker	Quartile	Rigid-body	PCC-extended
M1 (L=360mm)	Median	13.3%	10.2%
M1 (L=360mm)	Q_3	17.7%	13.5%
M2 (L=600mm)	Median	11.8%	9.1%
M2 (L=600mm)	Q_3	16.25%	13.8%

TABLE II: Median and the third quartile (Q3) of the relative error for 16 segment models generated using using eight IMUs.

C. Conditions of the data acquisition

In a motion capture environment, the arm is fixed, base up, on an elevating structure (See Fig. 1). Over a 250sec trial, the arm is moved manually with a stick attached to the arm between the two motion capture markers. During the trial 8,000 samples are collected, The median of the arm's moving speed is 0.1ms^{-1} at marker 1 and 0.2ms^{-1} at marker 2, reaching speeds up to 0.6ms^{-1}.

The arm is moved in different modes such as oscillating (going back and forth on each side of the structure, the tip reaching positions below the base w.r.t the z-axis), buckling (pseudo vertical position and contractions along the z-axis), and "dog chasing its tail" (the arm making circles around the z-axis while its shape forms a question mark).

IV. Results

A. Main takeaways

1) Overall estimation performance for both models: The lowest error for the rigid-body model was found using eight IMUs with a median error of 48mm. The lowest error for the PCC-extended rigid-body model was found using eight IMUs with a median error of 37mm. As described in Table II, the error normalized by the distance along the arm, which we refer to as *relative error*, decreases from $L = 360$mm to $L = 600$mm for both models. The PCC-extended rigid-body model is consistently more accurate than the rigid-body model, with approximately 25% lower error.

2) Computation considerations: The rigid-body model can be computed in real time for two, four, and eight IMUs using a personal computer with an Intel® Core™ i7-10510U Processor (4.90 GHz) and 16GB of RAM. The PCC-extended rigid-body model could not be constructed in real-time on the same machine, due to the added computational burden of repeatedly performing spherical linear interpolation. It took 18 minutes to compute each model for 4 min 10 secs of recording, which equates to roughly a factor of 4.4 slow-down relative to real time. It should be noted, however, that a more efficient implementation could lower this computation time significantly.

B. Results for the rigid-body model

As expected, accuracy increases with the density of IMUs for both models. However, the accuracy doesn't increase linearly: for marker #1, the median relative error for two, four, and eight IMUS is 44.4%, 15.0%, and 13.6%, respectively (Fig. 3). Furthermore, the evolution of the relative error decreases along the arm. For eight IMUs at $L = 360$mm, the relative error is 13.3% whereas at $L = 600$mm, the relative error is 11.5% (Tab. II).

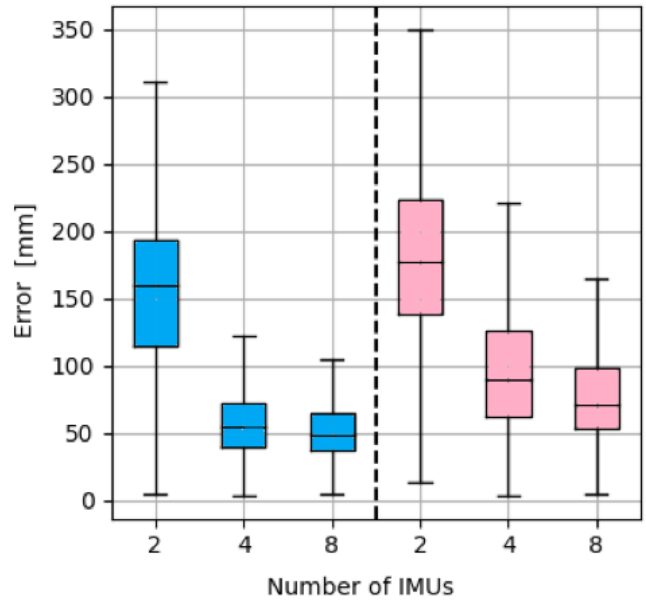

Fig. 3: Error of the rigid-body model with two, four, eight IMUs. The blue box-plots corresponds to the error on the marker 1. The pink box-plots corresponds to the error on the marker 2.

C. Results for the PCC-extended rigid-body model

The PCC-extended rigid-body model has two parameters to define: the number of segments and the number of IMUs. We will first present the effects of changing the number of segments using eight IMUs, and then present the effects of changing the number of IMUs using 16 segments.

1) Effects of changing the number of segments using eight IMUs (Fig. 4): The accuracy of the model increases with the number of segments until a plateau is reached. For example, while the Q3 of the error for eight segments is 16.9%, for 16 segments and more, the Q3 of the error remains similar at 13.5%.

2) Effects of changing the number of IMUs using 16 segments (Fig. 5): Similar to the rigid-body model, error decreases as the number of IMUs increases. However, unlike the rigid-body model, the error continues decreasing significantly between four and eight IMUs. The median of the relative error decreases by 10% between markers 1 and 2.

3) The error in pose estimation along the arm: The absolute error increases along the length of the arm. For eight IMUs and 16 segments, at $L = 360$mm, the Q3 of the absolute error is 48mm whereas at $L = 600$mm, the Q3 of the absolute error is 83mm (Tab. II). Fig 6 shows a 3D hull containing the Q3 of the PCC-extended rigid-body model estimation.

However, if we consider the error normalized by the length moving from the base along the arm, the relative error decreases. For eight IMUs and 16 segments, at $L = 360$mm, the

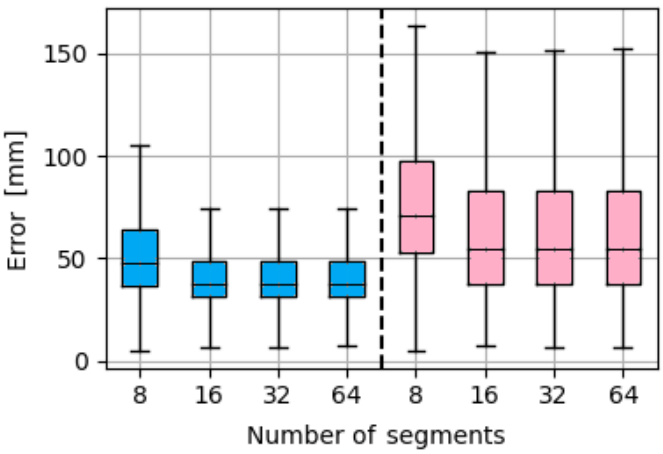

Fig. 4: Using PCC-extended rigid-body model with eight IMUs and 8, 16, 32, and 64 segments. The blue box-plots corresponds to the error on the marker 1. The pink box-plots corresponds to the error on the marker 2.

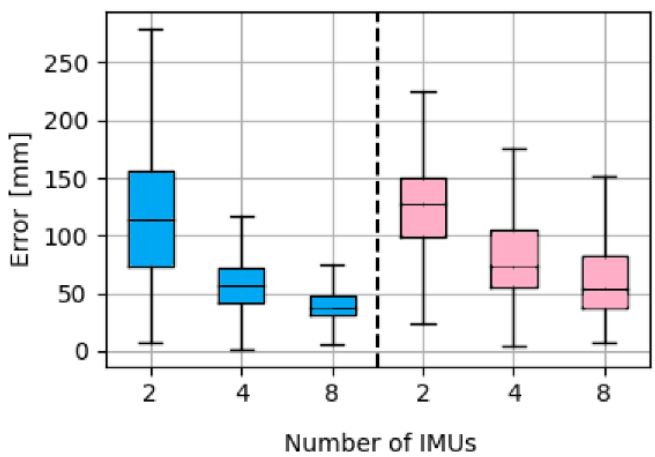

Fig. 5: Error of the 16 segments PCC-extended rigid-body model with two, four, and eight IMUs. The blue box-plots corresponds to the error on the marker 1. The pink box-plots corresponds to the error on the marker 2.

Q3 of the relative error is 10.2% whereas at $L = 600$mm, the Q3 of the relative error is 9.1% (Tab. II).

V. Discussion

A likely cause of error for the rigid-body model is the length of the segments for high bending values. Bending reduces the distance between the joints, whereas the rigid-body model keeps it constant. The PCC-extended rigid-body model captures the overall shape better, but since the model is a serial chain, a small error near the base propagates through the entire arm. This is why the error at the end-effector tends to be larger than that at the motion capture marker near the middle of the arm (see Fig. 6).

Another likely source of error is the orientation of the arm relative to gravity. In our experiments the arm was mounted

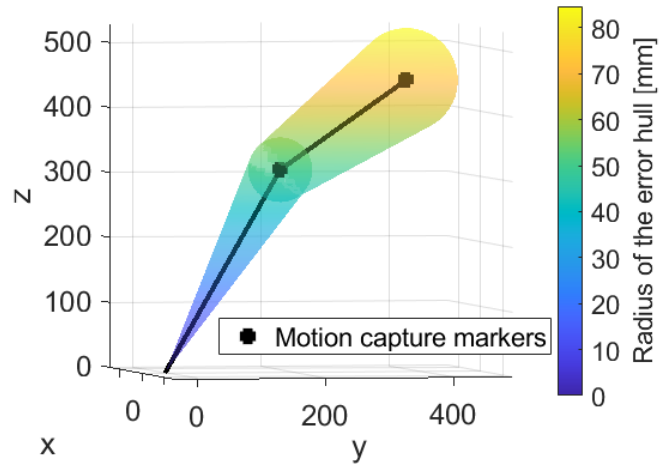

Fig. 6: 3D hull containing 75% of the PCC-extended rigid-body model estimation (error below the Q3) for eight IMUs and 16 segments. The arm pose is reconstructed using motion capture data.

such that it points upwards from the base, causing the weight of the arm to induce larger non-constant curvatures near the base. This curvature near the base leads to errors that propagate and grow along the length of the arm. We could potentially reduce this error by mounting the arm such that it points downward instead, or by changing the IMU positions along the arm such that the density of IMUs increases as we get closer to the arm's base.

IMU drift might also generate errors in the IMU measurements, especially at the base. The IMU we used, the bno055, auto-calibrates in the background by reaching different positions [27]. The first IMUs along the arm have a limited range of positions, which might reduce the quality of their calibration and therefore increase the error.

Ideally, the shape of the arm would be computed from IMU data quickly enough to provide a feedback signal to a real-time closed-loop controller. In our experiments the rigid-body model was able to provide estimates in real-time, but the PCC-extended rigid-body model was not. Because this work is meant only to be a proof of concept for IMU based proprioception, achieving real-time computation was not our focus. However, there are many ways in which computational efficiency could be improved in the future. One such approach could be to ignore the real part of the quaternions and keep only a vector along which the arm would be aligned. Then linear interpolation could be used between these vectors, which is computationally 3 times lighter than spherical linear interpolation using MATLAB implementation [28].

Fig. 5 shows that increasing the number of actual IMU sensors used to generate a 16 segment model reduces the model's error. However, Fig. 4 shows that increasing the number of segments beyond 16 does not noticeably reduce the error. We interpret this to mean that a 16 segment rigid-body model is a suitable approximation of a PCC model of this

arm. Therefore, we expect that for 16 or more IMUs, a rigid-body model and a continuous PCC model will approximate the shape of the arm with roughly equal accuracy. This "maximum number of IMUs" will vary from arm to arm depending on its geometry and bending stiffness. Future work could develop a method to compute this number for any soft arm as a function of its maximum curvature and some tolerated error threshold. There are computation concerns when adding additional IMUs, however. Adding more IMUs could reduce the reading frequency, especially if using a Micro Controller Unit. This could be compensated by using higher speed protocols than I2C such as SPI.

VI. Conclusion

In this paper we show that IMUs – inexpensive, available off-the-shelf, and easy-to-integrate – can be used for proprioception of soft robot arms. Using a rigid-body model generated from IMU data and spherical linear interpolation, we were able to estimate the position of the end-effector of a soft robot arm with a median error of less than 10% of the arm's length. Future work could employ methods to improve the accuracy of IMU-based shape estimates. Future work could also investigate ways to make IMUs easier to integrate into soft robots. This could be done by mounting the IMUs on small custom PCBs, and by integrating an entire string of IMUs as a self-contained sensor.

References

[1] H. Wang, M. Totaro, and L. Beccai, "Toward perceptive soft robots: Progress and challenges," *Advanced Science*, vol. 5, no. 9, p. 1800541, 2018.

[2] Y.-L. Park, B.-R. Chen, and R. J. Wood, "Design and fabrication of soft artificial skin using embedded microchannels and liquid conductors," *IEEE Sensors journal*, vol. 12, no. 8, pp. 2711–2718, 2012.

[3] J. T. Muth, D. M. Vogt, R. L. Truby, Y. Mengüç, D. B. Kolesky, R. J. Wood, and J. A. Lewis, "Embedded 3d printing of strain sensors within highly stretchable elastomers," *Advanced materials*, vol. 26, no. 36, pp. 6307–6312, 2014.

[4] T. Helps and J. Rossiter, "Proprioceptive flexible fluidic actuators using conductive working fluids," *Soft robotics*, vol. 5, no. 2, pp. 175–189, 2018.

[5] M. C. Yuen, R. Kramer-Bottiglio, and J. Paik, "Strain sensor-embedded soft pneumatic actuators for extension and bending feedback," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 202–207, 2018.

[6] K. C. Galloway, Y. Chen, E. Templeton, B. Rife, I. S. Godage, and E. J. Barth, "Fiber optic shape sensing for soft robotics," *Soft Robotics*, vol. 6, no. 5, pp. 671–684, 2019. PMID: 31241408.

[7] W. Zhuang, G. Sun, H. Li, X. Lou, M. Dong, and L. Zhu, "Fbg based shape sensing of a silicone octopus tentacle model for soft robotics," *Optik*, vol. 165, pp. 7–15, 2018.

[8] H. Zhao, J. Jalving, R. Huang, R. Knepper, A. Ruina, and R. Shepherd, "A helping hand: Soft orthosis with integrated optical strain sensors and emg control," *IEEE Robotics & Automation Magazine*, vol. 23, no. 3, pp. 55–64, 2016.

[9] I. Van Meerbeek, C. De Sa, and R. Shepherd, "Soft optoelectronic sensory foams with proprioception," *Science Robotics*, vol. 3, no. 24, p. eaau2489, 2018.

[10] M. Luo, E. H. Skorina, W. Tao, F. Chen, S. Ozel, Y. Sun, and C. D. Onal, "Toward modular soft robotics: Proprioceptive curvature sensing and sliding-mode control of soft bidirectional bending modules," *Soft Robotics*, vol. 4, no. 2, pp. 117–125, 2017. PMID: 29182091.

[11] S. Ozel, N. A. Keskin, D. Khea, and C. D. Onal, "A precise embedded curvature sensor module for soft-bodied robots," *Sensors and Actuators A: Physical*, vol. 236, pp. 349–356, 2015.

[12] W. Felt, K. Y. Chin, and C. D. Remy, "Contraction sensing with smart braid mckibben muscles," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 3, pp. 1201–1209, 2015.

[13] N. Lazarus and S. S. Bedair, "Bubble inductors: Pneumatic tuning of a stretchable inductor," *AIP Advances*, vol. 8, no. 5, p. 056601, 2018.

[14] W. Felt, M. J. Telleria, T. F. Allen, G. Hein, J. B. Pompa, K. Albert, and C. D. Remy, "An inductance-based sensing system for bellows-driven continuum joints in soft robots," *Autonomous robots*, vol. 43, no. 2, pp. 435–448, 2019.

[15] J. C. Case, J. Booth, D. S. Shah, M. C. Yuen, and R. Kramer-Bottiglio, "State and stiffness estimation using robotic fabrics," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 522–527, April 2018.

[16] M. C. Yuen, H. Tonoyan, E. L. White, M. Telleria, and R. K. Kramer, "Fabric sensory sleeves for soft robot state estimation," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 5511–5518, 7 2017.

[17] J. Tapia, E. Knoop, M. Mutný, M. A. Otaduy, and M. Bächer, "Makesense: Automated sensor design for proprioceptive soft robots," *Soft Robotics*, vol. 7, no. 3, pp. 332–345, 2020. PMID: 31891526.

[18] R. L. Truby, C. D. Santina, and D. Rus, "Distributed proprioception of 3d configuration in soft, sensorized robots via deep learning," *IEEE Robotics and Automation Letters*, vol. 5, pp. 3299–3306, April 2020.

[19] T. G. Thuruthel, B. Shih, C. Laschi, and M. T. Tolley, "Soft robot perception using embedded soft sensors and recurrent neural networks," *Science Robotics*, vol. 4, no. 26, p. eaav1488, 2019.

[20] N. Ahmad, R. A. R. Ghazilla, N. M. Khairi, and V. Kasi, "Reviews on various inertial measurement unit (imu) sensor applications," *International Journal of Signal Processing Systems*, vol. 1, no. 2, pp. 256–262, 2013.

[21] J. Hughes, F. Stella, C. D. Santina, and D. Rus, "Sensing soft robot shape using imus: An experimental investigation," in *International Symposium on Experimental Robotics*, pp. 543–552, Springer, 2020.

[22] O. D. Yirmibesoglu and Y. Menguc, "Hybrid soft sensor with embedded imus to measure motion," in *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 798–804, IEEE, 2016.

[23] I. Robert J. Webster and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.

[24] E. Mitchell and A. Rogers, "Quaternion parameters in the simulation of a spinning rigid body," *SIMULATION*, vol. 4, no. 6, pp. 390–396, 1965.

[25] T. Foote, "tf: The transform library," in *2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA)*, pp. 1–6, 2013.

[26] M. Leeney, "Fast quaternion slerp," *International Journal of Computer Mathematics*, vol. 86, no. 1, pp. 79–84, 2009.

[27] B. Sensortec, "Bno055 intelligent 9-axis absolute orientation sensor," *Bosch Sensortec, Baden-Württemberg, Germany*, pp. 1–106, 2016.

[28] S. M. Toolbox *et al.*, "Matlab," *Mathworks Inc*, 1993.