

Efficient and Scalable Inverse Kinematics for Continuum Robots

Samuel Wild, Tianyi Zeng, Abdelkhalick Mohammad, John Billingham, Dragos Axinte, Xin Dong

Abstract—With their flexible nature, continuum robots offer hyper-redundancy regarding their workspace; their backbone can take many shapes upon a single tip position and orientation. Deciphering which backbone shape to use under certain conditions is crucial to their operation, especially given the rise in their use in industries such as inspection and repair, and minimally invasive surgery. This complexity increases when additional continuum robot sections are used. This paper presents a novel Piecewise Dual Quaternion (PDQ)-based algorithm for modeling continuum robots, demonstrating improved performance compared to the traditional pseudo-Inverse Jacobian (IJ) method. The proposed algorithm reduces computational complexity and increases convergence speed. We validate the algorithm through simulation studies, comparing its performance in terms of accuracy, frequency, and sequential backbone smoothness, for which a metric is defined. Furthermore, we assess the algorithm’s scalability by extending the analysis to continuum robots with 4, 5, and 6 sections. The results indicate that the PDQ algorithm consistently outperforms the IJ method across various robot sections and paths, increasing the modeling frequency by an order of magnitude while maintaining tip accuracy. These findings have the potential to enable high precision of multi-section continuum robots and facilitate the use of haptic feedback sensors in practical applications.

Index Terms—Continuum Robots, Kinematics, Modeling.

I. INTRODUCTION

CONTINUUM robots (CRs) are a type of robot that have “an actuatable structure whose constitutive material forms curves with continuum tangent vectors” [1]. This enables them to bend, often extend or contract, and twist at any point along their structure. Their small diameter-to-length ratio and high degree of freedom (DoF) design enable flexibility and the ability to avoid obstacles, and perform different forms of locomotion and grasping compared to traditional robots [2]. Their design makes them well suited for applications in confined environments that require dexterity, such as inspection and repair [3] [4], minimally invasive surgery [1] [5] [6], nuclear [3] and search and rescue applications [7].

Unlike traditional robots, their modeling is inherently complex due to the deformable or flexible element composition that change shape and orientation in response to forces. The

This work was supported by the University of Nottingham and Innovate UK under Grant REINSTATE: Repair, Enhanced Inspection, and Novel Sensing Techniques for increased Availability and reduced Through life Expense (TS/V005103/1). (Corresponding author: Xin Dong) S. Wild, T. Zeng, A. Mohammad, D. Axinte and X. Dong are with the Rolls-Royce UTC in Manufacturing and On-Wing Technology, University of Nottingham, NG8 1BB, UK. (e-mail: {samuel.wild; tianyi.zeng; abd.mohammad1; dragos.axinte; xin.dong;}@nottingham.ac.uk)

J. Billingham is with the School of Mathematical Sciences, The University of Nottingham, University Park, Nottingham, NG7 2RD, UK. (e-mail: john.billingham@nottingham.ac.uk)

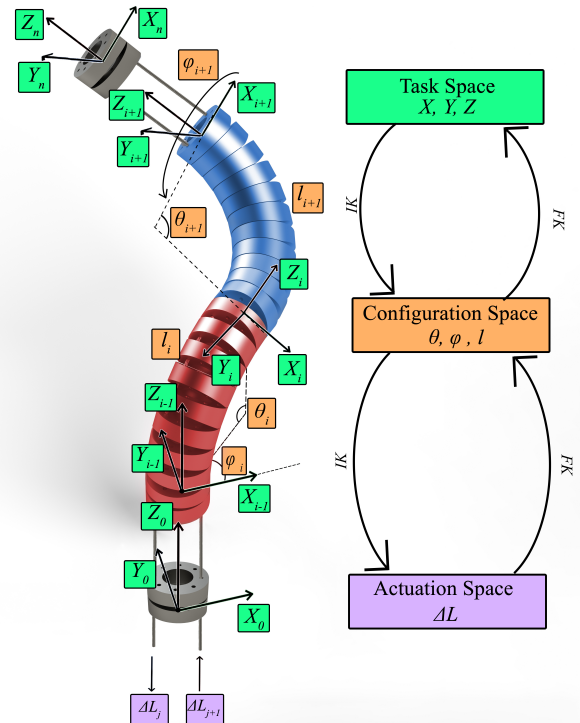


Fig. 1. A multi-section continuum robot, showing the PCC design parameters and modeling spaces. The variable i represents the section number from 0, to n , where n is the tip of the robot, while j represents the cable number. Forward and inverse kinematics are denoted as FK and IK respectively, while l describes the length of the CR backbone.

piecewise constant-curvature (PCC) [8] [9] model, shown in Fig. 1, represents the CR geometry as a finite number of mutually tangent curved segments, each with a constant curvature along its length. This approximation is easily implemented in practice due to the use of simplified mathematical models. This modeling approach maps from configuration space to task space and is independent of the type of CR, but robot specific kinematics are required to map between the configuration and the actuation space, typically cable displacements for tendon-driven CRs. Due to model mismatch, and non-uniform bending in practice, this approach provides an approximate representation of the robot’s motion.

Cosserat rod theory is another robot-independent modeling approach that describes the backbone’s motion and deformation as a continuous rod subject to both bending and twisting deformations. This approach provides an exact static solution

[10], and can accurately represent the nonlinear behavior of these systems. However, calculation of the Cosserat rod model is computationally burdensome and requires measurements that may not be available from miniaturized sensing solutions. A numerical approach for the forward dynamics simulation of Cosserat rod-based robot models has been solved in real-time at 250 Hz [11].

Rigid link modeling assumes that the robot's flexible structure can be approximated as an infinite serial chain of rigid links connected by joints [6]. This approach allows for the use of established kinematic models to represent the robot's motion, however, it does not consider non-rigid properties of CRs and may not be suitable for modeling the complex deformations. Additionally the number of rigid links chosen balances efficacy and accuracy hence a small number of rigid links may not accurately represent the robot's behavior.

Finite element methods (FEM) are another modeling approach used to simulate CR behavior, however this method analyzes and solves the physical CR structure made up of many small elements, rather than a collection of discrete rigid links. FEM requires a more detailed model than PCC and is effective at modeling the statics and dynamics of CRs by solving a set of equilibrium equations between the position, orientation, internal force, and internal torque of the robot [12]. FEM based models can provide accurate descriptions of the nonlinear behavior of the robot due to its ability to handle large deformation and nonlinearities in materials [12]. However, they lack a closed-form solution and require a numerical method to solve, which increases computational expense making it less suited to real-time applications [13].

Kinematic inversion relies upon analytical or learning-based methods where robot proprioception is not available, a traditional inversion method is the pseudo-Jacobian, however its performance is limited near singularities [14]. The damped least squares method avoids the pseudo-inverse problems but requires an optimized damping parameter based on the robot hardware [14]. Euler angle parameterization also lacks computational efficiency, but quaternions and modal kinematics respond to smoother path planning [15]. Building on these traditional approaches, FABRIK-based algorithms account for continuum robots with variable curvature, and solves the inverse kinematics 1.5 to 2 times faster than the Jacobian-based method [16].

Learning-based approaches have recently gained popularity due to their ability to encompass physical non-linearities, specifically for tendon-driven CRs where cable elongation and friction present difficulties. Model-free approaches offer good performance but require large training sets, for example, 80,000 measured data points were required to learn the inverse kinematics of a 6 DoF CR, which achieved an experimental tip error of less than 1% of the robot length [17]. To increase accuracy, further training data is required; its downfall lies when the robotic hardware changes, the inverse mapping requires retraining. Though promising, there is an element of blindness to physical interactions modeled in learning-based approaches, where unpredictable environmental factors can raise questions of usability and safety.

Regarding CR proprioception, haptic feedback is essential

for CR applications where safety and control are necessary, providing the user with a sense of touch. Furthermore, this is a crucial step toward precise minimally invasive surgery within the medical industry, as grasping tissue without force feedback can cause excessive damage. Therefore, when rendering haptics, the forces applied to the haptic sensor must be updated at 1kHz to maintain realistic haptic simulation [18], additionally for the stable rendering of clean and hard textured surfaces, update rates of greater than 1kHz are preferred [19].

In this paper, we propose an efficient, scalable modeling algorithm that enables configuration angle calculation for desired paths, suited to real-time applications. The proposed algorithm is validated through simulation and compared to the traditional method, the pseudo-inverse Jacobian, in terms of tip accuracy, scalability, modeling frequency, and sequential backbone smoothness. In the following sections, we will describe the algorithm in detail, provide simulation results for a 6 DoF CR, and discuss the scalability of the approach for CR with up to 12 DoF. Finally, we conclude with a discussion of the implications of our work and suggest areas for future research.

II. METHODOLOGY

A. PCC-based kinematics

The PDQ method provides an inverse calculation for the PCC model, its simplicity and ease to calculate mean that it is the most likely method to be used in CR control. PCC-based kinematics [8] [9] are employed to move between the configuration and task space in Figure 1. The shape of a CR backbone can be represented as a series of 2 DoF sections, individually represented along the left of Fig. 1. PCC forward kinematics describes the tip position of the i th section of a CR as

$$t_{i-1}^i = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \frac{l_i}{\theta_i} \begin{bmatrix} c\varphi_i(1 - c\theta_i) \\ s\varphi_i(1 - c\theta_i) \\ s\theta_i \end{bmatrix}, \quad (1)$$

where the substitution of i represents each section, l_i represents the backbone length while, c and s are equivalent to \cos and \sin respectively, i.e. $c\varphi_i$ is equal to $\cos \varphi_i$. The orientation of the tip between the proximal and the distal section of frames $S_{(i-1)}$ and S_i involves a successive rotation of φ_i around the Z_i axis, θ_i around the Y_i axis and $-\varphi_i$ around the Z_i axis. Known as Bishop's frame, this rotation matrix, R_{i-1}^i , can be written as

$$R_{i-1}^i = \begin{bmatrix} c^2\varphi_i(c\theta_i - 1) + 1 & -s\varphi_i c\varphi_i(c\theta_i - 1) & c\varphi_i \\ s\varphi_i c\varphi_i(c\theta_i - 1) & c^2\varphi_i(1 - c\theta_i) + c\theta_i & s\varphi_i \\ -c\varphi_i s\theta_i & s\varphi_i s\theta_i & c\theta_i \end{bmatrix} \quad (2)$$

Therefore, the homogeneous transformation matrix can be written as

$$T_{i-1}^i = \begin{bmatrix} R_{i-1}^i & t_{i-1}^i \\ 0_{3 \times 1} & 1_{1 \times 1} \end{bmatrix} \quad (3)$$

The forward kinematics for the robot backbone of a multi-section CR can be obtained by adding further sections and calculated by equating each section's proximal frame to the

distal frame of the previous one. Further, the PCC model parameters can be reduced using dual quaternions instead of Euler angles [20]. The unique mapping from transformation matrix to dual quaternion, \hat{q}_{ih} , can be written as the dual quaternion product of \hat{q}_{tr1} , \hat{q}_{tr2} and \hat{q}_{tr3} ,

$$\hat{q}_{tr1} = [c(\frac{\varphi_i}{2}) \quad 0 \quad 0 \quad s(\frac{\varphi_i}{2}) \quad 0 \quad 0 \quad 0 \quad 0], \quad (4)$$

$$\hat{q}_{tr2} = [c(\frac{\theta_i}{2}) \quad 0 \quad s(\frac{\theta_i}{2}) \quad 0 \quad 0 \quad 0 \quad 0 \quad \frac{l_i}{\theta_i} s(\frac{\theta_i}{2})], \quad (5)$$

$$\hat{q}_{tr3} = [c(-\frac{\varphi_i}{2}) \quad 0 \quad 0 \quad s(-\frac{\varphi_i}{2}) \quad 0 \quad 0 \quad 0 \quad 0], \quad (6)$$

$$\hat{q}_{ih} = \hat{q}_{tr1} \hat{\otimes} \hat{q}_{tr2} \hat{\otimes} \hat{q}_{tr3}, \quad (7)$$

where $\hat{\otimes}$ represents the dual quaternion product. The dual quaternion representation requires fewer parameters, 8, than the Euler angle representation, 16. When modeling a multi-section CR, the calculated kinematics of a 3-section robot, denoted as Q_{calc} are obtained through the successive dual quaternion product of Eq. 7.

$$Q_{calc} = \hat{q}_{(i-1)h} \hat{\otimes} \hat{q}_{ih} \hat{\otimes} \hat{q}_{(i+1)h} \quad (8)$$

Whereby the components of the dual quaternion, Q_{calc} , are indexed as $(Q_{calc_1}, Q_{calc_2}, \dots, Q_{calc_8})$.

B. PDQ Algorithm

Hyper-redundancy of multi-section CRs offers multiple inverse kinematic solutions for a single desired tip pose and orientation, which makes determining the best solutions for a desired path difficult. Our solution performs iterative nonlinear optimization, converging to a local minimum while satisfying constraints and minimizing a cost function. The evaluated solution uses a gradient-based method, utilizing MATLAB's 'fmincon' and Sequential Quadratic Programming (SQP) algorithm. The algorithm's computational speed is enhanced by using the previous solution of the algorithm as the initial guess for the current solution. This strategy enables a small numerical difference between successive solutions when following a desired path. As a result, the algorithm starts the search close to the optimal solution, reducing iterations and overall computation time, while providing a viable path in the solution space. To further minimize this difference, the cost function, $F(k)$, minimizes the difference between previous and current solutions such that a smooth backbone and path motion is followed, the equation for a single section being

$$\Delta\theta_i(k) = |\theta_i(k) - \theta_i(k-1)|, \quad (9)$$

$$\Delta\varphi_i(k) = |\varphi_i(k) - \varphi_i(k-1)| \quad \text{mod } 2\pi, \quad (10)$$

$$F(k) = \sum_{i=1}^n 0.5(\Delta\theta_i(k))^2 + (\Delta\varphi_i(k))^2, \quad (11)$$

where k is the iteration number along the path, and n is the total number of sections. The partial derivatives of the

cost function are given to indicate the direction to move in the search space and further improve efficiency. While the algorithm minimizes the cost function, it also adheres to constraints,

$$ceq = [Q_{calc_1} - Q_{des_1}, Q_{calc_2} - Q_{des_2}, \dots, Q_{calc_8} - Q_{des_8}], \quad (12)$$

the difference between every index of Q_{calc} and the desired dual quaternion, Q_{des} , are constrained to zero as it changes through the desired path and orientation. Equation 12 does not carry a physical meaning, rather it serves a computational tool that guides the algorithm to converge to a solution that matches the desired path and orientation in Cartesian space. The subtraction constraint was chosen to enhance the computational efficiency of the algorithm rather than dual quaternion division, which involves the more computationally intensive task of calculating the conjugate, subtraction achieves the desired outcomes while reducing the computational load. Prior to constraint applications, unit dual quaternions are ensured, in doing so, both their magnitude constraint and dual part constraint are met. The maximum bending angle of the CR was also constrained such that it is bound to its workspace. In line with these constraints, the solution robustly converges to a result that aligns with the physical characteristics of the robot.

C. Selection of Solution

Due to the SQP algorithm, a local minimum is found that satisfies the constraints and minimizes the cost function. Using the robot's straight initial position, the initial guess to the algorithm are the configuration variables that are close to the start position, specifically $0.01rad$ for all angles, to reduce computational complexity. This approach contributes to the algorithm's effectiveness in finding a viable path in the solution space. The stopping criteria for the algorithm are defined as follows,

Optimality Tolerance: 5e-5

Step Tolerance: 5e-4

Constraint Tolerance: 1e-6,

These tolerances were found to give the best performance of the simulation results regarding tip and orientation accuracy for a 3-section CR.

III. PDQ ALGORITHM VALIDATION

In this section, we present the results of our simulation study of the proposed algorithm run on MATLAB r2022b on MacOS Ventura, M1 Pro (P-core, 3220 MHz) with 16GB RAM. The simulated CR was of three sections with an equal section length of 55mm and a maximum bending angle, θ , of 100° . The PDQ method was compared against the IJ method and, four different paths were followed 20 times each at a height of 90% of the robot length, the Z axis in Fig. 1. The evaluated paths consist of a circle, a square, a spiral, and a dual loop, which all lie in the X-Y plane. The paths for a 3-section CR are shown in Fig. 2, and their criteria are defined for multi-section CRs. All paths were uniformly distributed with a step resolution of 0.01mm and, the orientation of the tip

was maintained along the Z_0 axis, Fig. 1, as this is common for industrial inspection and repair techniques. An additional complex path was evaluated, shown in Fig. 3, that combines both a spiral and square to reach toward the robot's bending limits, the orientation was also varied such that the tip always points toward a fixed point at coordinates (0,0,165).

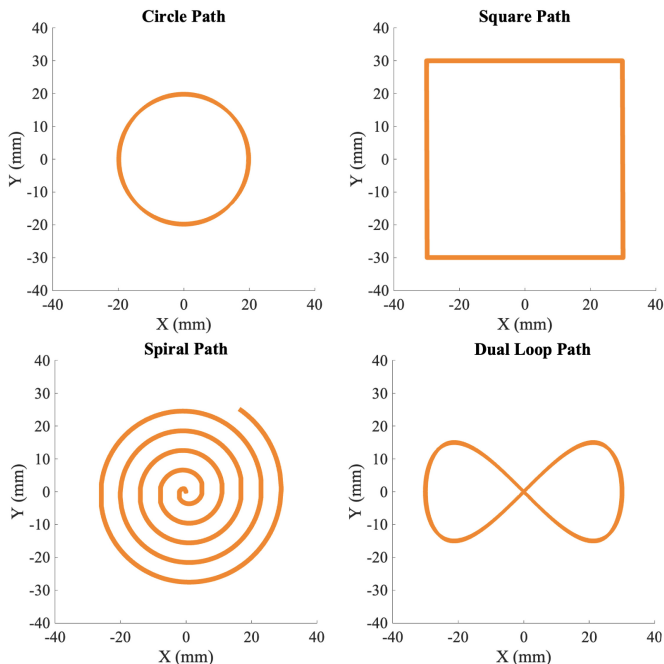


Fig. 2. Graphical representation of the simulated paths used to compare the PDQ method against the traditional IJ method. The paths are at a height equivalent to 90% of the robot length, the Z -axis, and variations include; the Circle Path – with a radius 12% of the total length of the robot; the Square Path – consists of four straight segments, each with a length of 60mm; the Spiral Path – starts at the origin of the X - Y plane and spirals outwards for 5 coils. There is a distance equal to 5% of the total length of the robot between consecutive loops; and the Dual Loop Path – with a total width of 60mm and height 30mm.

We present the performance of our algorithm in terms of accuracy, frequency, sequential backbone smoothness and scalability for increased degrees of freedom. To evaluate the accuracy of our proposed method, we compare the robot tip position and orientation between the desired path and the calculated path; the Euclidean distance between the actual and desired tip positions calculates the tip position error; the orientation error is calculated as the difference between actual and desired orientation angles. Frequency, or update rate, was calculated as the total number of steps in the path divided by the time it took to compute the path,

$$\text{Frequency} = \frac{P}{\Delta t}, \quad (13)$$

where P is the total number of path steps, and Δt is the computation time. We introduce sequential backbone smoothness (SBS) that quantifies the uniformity of the orientation distribution of a multi-section continuum robot. The SBS is computed as the average of the sum of absolute differences of the configuration angles, θ and φ , for each section and step along the path,

$$SBS = \frac{1}{nP} \sum_{k=1}^P \sum_{i=1}^n (\Delta\theta_i(k) + \Delta\varphi_i(k)) \quad (14)$$

A lower value of the SBS indicates a smoother backbone in the θ and φ space, while a higher value suggests a more significant variation between the orientations of the sections.

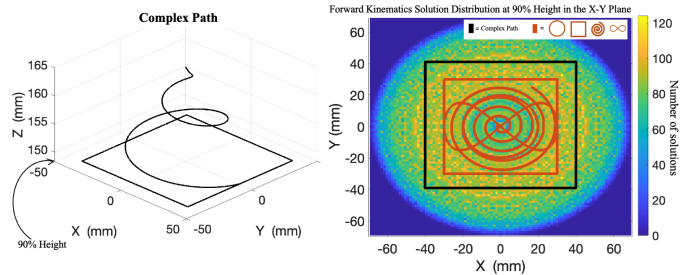


Fig. 3. A comparison between the Complex Path and primitive paths. On the left, a 3D plot illustrates the Complex Path, which begins in the straight configuration, spirals down and outward to 90% of the robot's maximum height, and transitions into a square path. On the right, a density plot depicts the number of feasible solutions at 90% height within the workspace. Both the primitive paths and complex path are overlaid in orange and black respectively, highlighting their coverage of the workspace. While the primitive paths lay comfortably in a solution-dense area of the workspace, the complex path reaches up to 81% of the workspace boundary where there are fewer solutions.

A. PDQ vs. IJ method: Three-Section Robot

The proposed PDQ algorithm accurately followed the desired paths while adhering to constraints, results for a three-section CR are shown in Table I. The mean frequency across all paths was 1220 Hz, yielding a mean time to solution of 0.81 ms, demonstrating fast convergence of configuration variables. While the mean tip error remained small, being $7.2 \times 10^{-8}\%$ of the robot length, indicating that the end effector followed the desired path with high accuracy. Similarly, the mean pitch, yaw, and roll errors were beneath a practical threshold of 10^{-5} rad which is deemed acceptable.

The PDQ method was compared to the IJ method, presented in Table II, to enable a fair comparison, the tip error was kept constant across the two algorithms such that frequency could be compared. The PDQ method demonstrates frequency of an order of magnitude larger than the IJ method, highlighting its performance benefits. The PDQ method produced a smaller mean SBS than the IJ method indicating that the PDQ method took smaller steps in configuration angles along the paths. A smaller SBS was observed for the square and dual loop paths, while the SBS remained comparable for circular and spiral paths, indicating that the backbones followed similar motions. The complex path results provide an insight into the PDQ method's accuracy under challenging orientation tracking while changing height and tracking a path that lies close to the workspace boundary. The results, presented in Table III, underline the method's robustness in maintaining high accuracy when the path involves more complexity, especially as bending angles get closer to their limits.

B. PDQ Subtraction vs. Division Constraints

The proposed algorithm adheres to index subtraction constraints, shown in Equation 12, diverging from the tradi-

TABLE I
THREE-SECTION RESULTS FOR PDQ METHOD





Path X-Y Plane	Frequency / Hz	Tip Error / mm	Orientation Error / rad			SBS
			Pitch	Yaw	Roll	
	1230	1.77×10^{-5}	4.42×10^{-8}	8.34×10^{-10}	4.42×10^{-8}	7.53×10^{-7}
	1185	9.56×10^{-6}	5.64×10^{-8}	8.97×10^{-9}	5.64×10^{-8}	2.97×10^{-7}
	1160	7.30×10^{-6}	3.49×10^{-9}	3.40×10^{-7}	3.49×10^{-9}	4.07×10^{-6}
	1305	1.00×10^{-5}	7.80×10^{-8}	5.04×10^{-9}	7.17×10^{-8}	2.85×10^{-7}
Mean	1220	1.11×10^{-5}	4.55×10^{-8}	8.87×10^{-8}	4.40×10^{-8}	1.35×10^{-6}

TABLE II
THREE-SECTION RESULTS FOR PSEUDO INVERSE JACOBIAN METHOD





Path X-Y Plane	Frequency / Hz	Tip Error / mm	Orientation Error / rad			SBS
			Pitch	Yaw	Roll	
	127	1.62×10^{-5}	1.68×10^{-16}	4.89×10^{-17}	1.58×10^{-16}	7.53×10^{-7}
	125	9.56×10^{-6}	1.62×10^{-16}	5.46×10^{-17}	1.62×10^{-16}	3.60×10^{-6}
	112	7.75×10^{-6}	1.53×10^{-13}	1.13×10^{-9}	7.68×10^{-10}	4.37×10^{-6}
	127	9.56×10^{-6}	8.12×10^{-10}	7.56×10^{-7}	8.74×10^{-7}	8.80×10^{-6}
Mean	123	1.08×10^{-5}	2.03×10^{-10}	1.89×10^{-7}	8.75×10^{-7}	4.37×10^{-6}

TABLE III
THREE-SECTION RESULTS FOR PDQ METHOD TRACKING THE COMPLEX PATH

Complex Path	Frequency / Hz	Tip Error / mm	Orientation Error / rad			SBS
			Pitch	Yaw	Roll	
Mean	1240	6.52×10^{-5}	1.48×10^{-5}	9.59×10^{-7}	1.43×10^{-5}	3.29×10^{-7}

tional approach of dual quaternion division. This choice was motivated by a desire to enhance computational efficiency, dual quaternion division is computationally expensive, while subtraction achieves comparable accuracy at a larger computational frequency. To validate the efficacy of the subtraction constraints, additional simulations were conducted for two primitive paths using the traditional division approach. The constraints were modified such that the divided dual quaternion, Equation 15, aimed to reach [1,0,0,0,0,0,0].





$$Q_{div} = \frac{Q_{calc}}{Q_{desired}}, \quad (15)$$

Whereby $Q_{div} = (Q_{div_1}, Q_{div_2}, \dots, Q_{div_8})$, thus, the division constraints were formulated as,

$$ceq = [Q_{div_1} - 1, Q_{div_2}, \dots, Q_{div_8}], \quad (16)$$

as the constraint function, $ceq(x) = 0$. Circular and square paths were re-simulated 20 times using the division-based method, the tip and orientation error remained comparable between that of the proposed method, while the computational

TABLE IV
THREE-SECTION RESULTS FOR PDQ SUBTRACTION VS. DIVISION CONSTRAINTS FOR PRIMITIVE PATHS.

Path X-Y Plane	Method	Frequency / Hz	Frequency Difference to Subtraction	Tip Error / mm	Orientation Error / rad			SBS
					Pitch	Yaw	Roll	
	Subtraction (Equation 12)	1230		1.77×10^{-5}	4.42×10^{-8}	8.34×10^{-10}	4.42×10^{-5}	7.53×10^{-7}
	Division (Equation 16)	693	-44%	7.35×10^{-6}	8.93×10^{-7}	9.07×10^{-7}	8.40×10^{-7}	7.53×10^{-7}
	Subtraction (Equation 12)	1185		9.56×10^{-6}	5.64×10^{-8}	8.97×10^{-9}	5.64×10^{-8}	2.97×10^{-7}
	Division (Equation 16)	635	-46%	6.49×10^{-5}	2.10×10^{-6}	7.45×10^{-7}	7.75×10^{-7}	2.97×10^{-7}

frequency suffered a reduction of 44% and 46% respectively, the full results are shown in Table IV. This finding validates our choice of the subtraction constraints outlined in Equation 12, confirming the subtraction method yields higher frequency.

C. Evaluation of Algorithm Scalability

The algorithms were tested against their scalability, with more complex calculations per each additional section. The same paths as Fig. 2 were conducted for CRs with 4, 5, and 6 sections following the path criteria. The results, shown in Fig. 4, indicate that the PDQ method maintains superior performance as the number of robot sections increases. For a 6-section CR, the IJ algorithm (label D in Fig. 4) were left for 48 hours and did not converge.

Across all paths and robot configurations, the PDQ algorithm consistently demonstrated a higher frequency than the IJ method, indicating that the proposed method can efficiently model continuum robots with increased degrees of freedom without compromising convergence speed alike the IJ method. Moreover, the mean orientation error for the PDQ method remained beneath the threshold, and mean tip error was consistently low as the number of sections increased, with a maximum of 0.08mm for the square path of both 5 and 6 sections. The tip error can be modified upon changing the optimization tolerances, however efficacy is implicated. The results demonstrate the PDQ's ability to maintain high accuracy and performance even for more complex robot configurations.

Analysis upon the SBS values across multiple paths, shows that for a three-section CR the sequential backbone smoothness of the PDQ and IJ algorithm are comparable for the circular and spiral path, each having smooth backbone motion, while the PDQ method shows smoother backbone motion than the IJ for the square and dual loop paths.

IV. DISCUSSION

In this paper, we presented a novel PDQ algorithm for modeling continuum robots, demonstrating its superior frequency, accuracy, and scalability performance compared to the traditional IJ method.

A. Key Findings

Our simulation results show that the proposed PDQ algorithm outperforms the IJ method, with a mean frequency

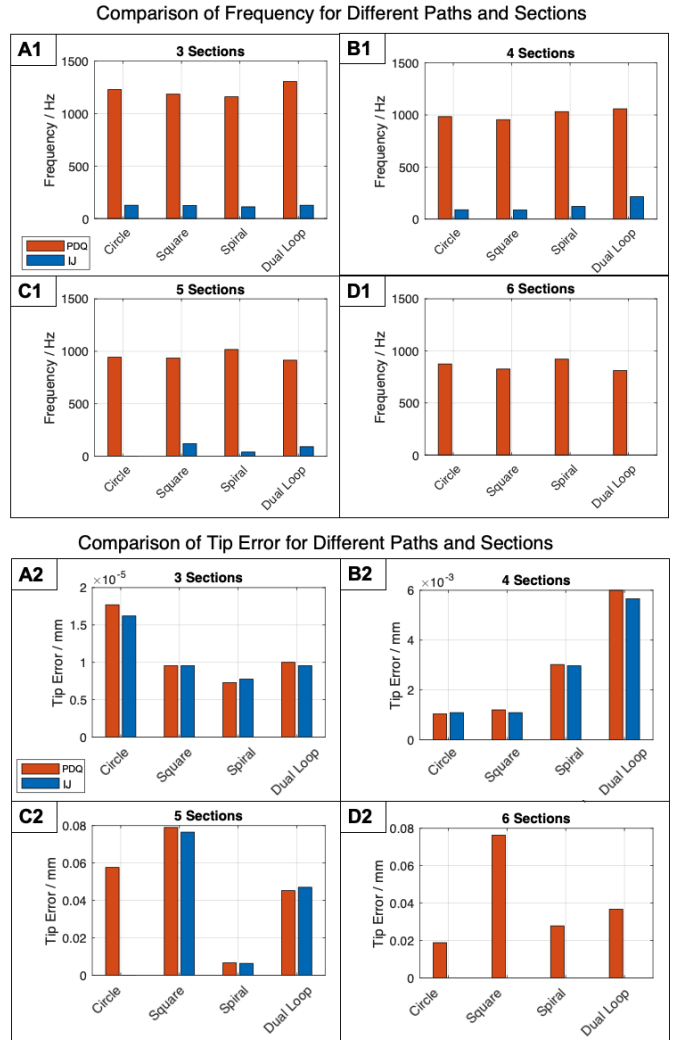


Fig. 4. A comparison between the PDQ method and IJ method, in terms of mean frequency and tip error. A1 and A2 compare Table I and II, while additional sub-figures alphabetically increment in number of CR sections. The IJ circular path in C1 and C2 did not converge to a feasible value, while D1 and D2 only show the PDQ method as the IJ did not converge.

of 1220 Hz, which is above the required 1 kHz for haptic feedback applications. Additionally, PDQ maintains high accuracy regarding tip and orientation error across various robot configurations and paths and scalability for robots with increased degrees of freedom. A crucial factor when considering the application of the algorithm in real-world scenarios, where precise control of the robot's position and orientation is essential.

The use of dual quaternions allows for a unified representation of both position and orientation, reducing the complexity while enhancing accuracy. In addition, the compact nature of dual quaternions ensures PDQ remains robust across the full range of tip positions and orientations.

The PDQ algorithm presents an optimization-based approach that leverages the robot's kinematic model to find the optimal joint configurations subject to constraints. This approach minimizes the differences between joint angles, such that the algorithm converges to the closest set of angles, enabling a backbone with the smallest displacement in the θ and φ plane. This approach allows for better handling of joint angles across a desired path, such that when practiced this minimized change allows smooth actuation and a reduction of tip error due to reduced backbone smoothness compared to the IJ method. Considering backbone motion for the circular path requires a rotation around φ for a single section, and bending angles, θ , were kept constant, it can be inferred that the PDQ method surpasses the IJ method in terms of sequential backbone smoothness for more complex paths, such as the square and dual loop, shown in Table I and II.

Additionally, the PDQ algorithm considers the CR's physical constraints, notably bending limits. Each CR section has a maximum allowable bending angle, for example 100 degrees, which is integrated into the algorithm, ensuring that the generated joint configurations are feasible and practical, leading to safer and reliable control in real-world applications.

It's important to clarify that the choice to focus on a 3-section CR was deliberate, because of its common use in industrial applications. While the inverse kinematics for additional sections can be further simplified, the computational cost required was not deemed justifiable within the context of our primary analysis. Additionally the PCC model provides just simplification of the robot's kinematics regarding accuracy and efficiency. The PDQ algorithm is capable of handling a larger number of sections than simulated, but were not computed as our analysis covers a broad range of CR compositions commonly used industries.

B. Practical Applications

One significant application of our PDQ algorithm is in the realm of haptic feedback for continuum robots, where a minimum frequency of 1 kHz is required to maintain realistic haptic simulation. PDQ's high-frequency performance enables real-time modeling of continuum robots in various industries, such as medical procedures, manufacturing, and inspection tasks. Our method's precise modeling can be applied to various continuum robot designs e.g. central backbone and pivot joint designs, to improve their efficiency, accuracy and safety in the applications.

C. Future Work

Future work could aim to further improve the PDQ algorithms frequency to enable stable rendering of textured surfaces while maintaining smooth motion, extending haptics applicability to more complex in-situ sensing. Furthermore, this method handles the inverse kinematics for a given path and orientation, parallel optimization-based approaches for path planning and robot modeling would allow for path planning within the reachable workspace and collision avoidance. Integrating the PDQ method within a control framework that accounts for system noise and external disturbances could also be explored. Furthermore, the improvement in computational time is attributed to the use of dual quaternions rather than any modification to the PCC model. While the PDQ algorithm presented is tied to the PCC model for validation purposes, its underlying framework is adaptable to other continuum robot models.

V. CONCLUSION

In conclusion, our proposed PDQ algorithm presents efficient and accurate modeling for multi-section continuum robots, it not only 10 times faster than the traditional inverse Jacobian method for a 3-section continuum robot but also demonstrates excellent scalability for robots with more sections, with a mean tip error of 0.04mm for a six-section CR at a mean frequency of 850 Hz. This performance across various robot configurations and paths highlights the PDQ method's potential for adoption in continuum robotics, enabling additional sensors that require high frequencies, such as haptic feedback, allowing more precise and efficient control of these robots in various applications.

REFERENCES

- [1] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, "Continuum robots for medical applications: A survey," *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1261–1280, 2015.
- [2] G. Chirikjian and J. Burdick, "A hyper-redundant manipulator," *IEEE Robotics Automation Magazine*, vol. 1, no. 4, pp. 22–29, 1994.
- [3] A. Ahmad-Mohammad, M. Russo, Y. Fang, X. Dong, D. Axinte, and J. Kell, "An efficient follow-the-leader strategy for continuum robot navigation and coiling," *IEEE Robotics and Automation Letters*, vol. 6, pp. 7493–7500, 10 2021.
- [4] X. Dong, D. Axinte, D. Palmer, S. Cobos, M. Raffles, A. Rabani, and J. Kell, "Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines," *Robotics and Computer-Integrated Manufacturing*, vol. 44, pp. 218–229, 2017.
- [5] E. Amanov, T.-D. Nguyen, and J. Burgner-Kahrs, "Tendon-driven continuum robots with extensible sections—a model-based evaluation of path-following motions," *The International Journal of Robotics Research*, vol. 40, no. 1, pp. 7–23, 2021.
- [6] T. Veiga, J. Chandler, P. Lloyd, G. Pittiglio, N. Wilkinson, A. kafash hoshiar, R. Harris, and P. Valdastrì, "Challenges of continuum robots in clinical context: A review," *Progress in Biomedical Engineering*, vol. 2, 06 2020.
- [7] M. M. Coad, L. H. Blumenschein, S. Cutler, J. A. R. Zepeda, N. D. Naclerio, H. El-Hussieny, U. Mehmood, J.-H. Ryu, E. W. Hawkes, and A. M. Okamura, "Vine robots," *IEEE Robotics Automation Magazine*, vol. 27, no. 3, pp. 120–132, 2020.
- [8] B. A. Jones and I. D. Walker, "Practical kinematics for real-time implementation of continuum robots," *IEEE Transactions on Robotics*, vol. 22, no. 6, pp. 1087–1099, 2006.
- [9] 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.

- [10] D. Rucker and R. Webster, "Statics and dynamics of continuum robots with general tendon routing and external loading," *Robotics, IEEE Transactions on*, vol. 27, pp. 1033 – 1044, 01 2012.
- [11] J. Till, V. Aloï, and C. Rucker, "Real-time dynamics of soft and continuum robots based on cosserat rod models," *The International Journal of Robotics Research*, vol. 38, no. 6, pp. 723–746, 2019.
- [12] S. Grazioso, G. Di Gironimo, L. Rosati, and B. Siciliano, *Modeling and Simulation of Hybrid Soft Robots Using Finite Element Methods: Brief Overview and Benefits*, 01 2021, pp. 335–340.
- [13] M. Chikhaoui and J. Burgner-Kahrs, "Control of continuum robots for medical applications: State of the art," in *International Conference and Exhibition on New Actuators and Drive Systems*. VDE VERLAG GMBH, 2018, pp. 154–164.
- [14] S. Buss, "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods," *IEEE Transactions in Robotics and Automation*, vol. 17, 05 2004.
- [15] I. Godage and I. Walker, "Dual quaternion based modal kinematics for multisection continuum arms," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015, pp. 1416–1422, 06 2015.
- [16] D. Kolpashchikov, O. Gerget, and V. Danilov, "FABRIKx: Tackling the Inverse Kinematics Problem of Continuum Robots with Variable Curvature," *Robotics*, vol. 11, no. 6, 2022.
- [17] R. Grassmann, V. Modes, and J. Burgner-Kahrs, "Learning the forward and inverse kinematics of a 6-dof concentric tube continuum robot in $se(3)$," 10 2018, pp. 5125–5132.
- [18] X. Hou and O. Sourina, "Real-time adaptive prediction method for smooth haptic rendering," *CoRR*, vol. abs/1603.06674, 2016.
- [19] S. Choi and H. Tan, "Effect of update rate on perceived instability of virtual haptic texture," vol. 4, 01 2004, pp. 3577 – 3582.
- [20] T. Lorenzo, G. Cazzulani, and F. Braghin, "Forward kinematic modeling of constant curvature continuum robots using dual quaternions," *7th ECCOMAS Thematic Conference on Smart Structures and Materials*, p. 16, 01 2015.