Database-Driven Capability Maps for Pareto-Based Configuration Selection in Tendon-Driven Continuum Robots

Mechanism and Machine Theory 230 (2026) 106606

Contents lists available at ScienceDirect

Mechanism and Machine Theory

journal homepage: www.elsevier.com/locate/mechmt

Research paper

Database-driven capability maps for Pareto-based configuration selection in tendon-driven continuum robots

Mohammad Jabari [a,b,*], Carmen Visconte [b], Giuseppe Quaglia [b], Abdelbadia Chaker [c], Med Amine Laribi [c,*]

[a] *Department of Informatics, Bioengineering, Robotics and Systems Engineering (DIBRIS), Università di Genova, Genova, Italy*
[b] *Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, Turin 10129, Italy*
[c] *Department of GMSC, Pprime Institute, University of Poitiers, CNRS, ISEA-ENSMA, UPR 3346, Poitiers, France*

ARTICLE INFO

Keywords:
Tendon-driven continuum robot
Capability map
Voxel-level configuration selection
Piecewise constant curvature
Particle swarm optimization
Manipulability

ABSTRACT

Tendon-driven continuum robots (TDCRs) have been widely investigated for minimally invasive surgery, inspection, and cluttered-environment operation because of their compliance and dexterity. However, dense workspace characterization and configuration selection remain challenging because inverse kinematics must be solved in a highly redundant, nonlinear space under tendon-actuation constraints. Existing capability map frameworks do not directly address configuration-level tendon feasibility or dexterity-aware selection among feasible alternatives. This paper presents a database-driven CAP framework that combines piecewise constant curvature kinematics with particle swarm optimization to construct an offline library of configurations consistent with the prototype tendon-routing and command-mapping model over 180 workspace voxels and 54,000 voxel-orientation samples. For each feasible sample, pose residuals and Jacobian-based dexterity descriptors are computed offline and stored in the database. At runtime, the method retrieves feasible entries for a queried voxel and ranks them through a Pareto-based trade-off between combined pose residual and inverse manipulability. The output is a dexterity-aware representative configuration for the queried local neighbourhood, rather than a newly optimized exact-pose solution. Motion-capture validation on a custom two-segment TDCR shows millimetre-order position errors in the evaluated ROI.

1. Introduction

Robotic manipulation has traditionally been dominated by rigid-link architectures that provide high precision and payload capacity, yet often lack adaptability when operating in confined or highly constrained environments. Continuum robots have emerged as an alternative class of manipulators capable of smooth bending and twisting along compliant backbones, enabling safe physical interaction and dexterous navigation through narrow, cluttered, and contact-sensitive spaces. These characteristics are particularly valuable in minimally invasive surgery, endoluminal interventions, and inspection tasks, where tool compliance and geometric adaptability are critical requirements [1–3].

Among continuum robot architectures, tendon-driven continuum robots (TDCRs) are particularly advantageous owing to their

* Corresponding authors.
E-mail addresses: mohammad.jabari@polito.it (M. Jabari), carmen.visconte@polito.it (C. Visconte), giuseppe.quaglia@polito.it (G. Quaglia), abdelbadia.chaker@univ-poitiers.fr (A. Chaker), med.amine.laribi@univ-poitiers.fr (M.A. Laribi).

https://doi.org/10.1016/j.mechmachtheory.2026.106606
Received 22 July 2026; Received in revised form 20 August 2026; Accepted 21 August 2026
Available online 9 September 2026

Table 1
Direct comparison between the conference version [21] and the present journal submission.

Aspect	[21]	Present submission
ROI	Distal voxelized ROI	Same ROI concept, extended to database-based selection and validation
Voxel/orientation samples	Voxel trials with downward directions	180 voxels, 300 directions/voxel, 54,000 targets
IK formulation	PCC + PSO for reachability	Disk-wise PCC + PSO with position and SO(3) residuals
Tendon/actuation treatment	Prototype routing described	Tendon-length and antagonistic motor-command mapping per retained entry
Stored CAP fields	Voxel success ratio	Configuration, target/achieved pose, residuals, commands, dexterity descriptors
Dexterity treatment	Not included	Offline Jacobian enrichment and 6D manipulability
Selection rule	Not included	Pareto/ideal-point residual-dexterity selection
Experimental validation	Future work	OptiTrack proof-of-concept validation

remote actuation principle. Tendons routed through spacer disks along a central backbone allow actuation to be placed at the proximal base, resulting in slender distal bodies and compact end-effectors [4,5]. Bending and twisting are generated through differential tendon length variations, avoiding bulky distal joints and facilitating deployment in diameter-constrained settings [6]. Despite these advantages, practical operation of TDCRs remains challenging because the mapping from actuator space to task space is highly nonlinear and strongly constrained by tendon routing, stroke limits, and tension feasibility [7–9]. Friction, hysteresis, tendon slack, and pretension introduce additional nonlinearities that are not captured by purely geometric models. As a result, the already high-dimensional and nonconvex IK problem becomes sensitive to hardware-specific transmission effects that may vary across configurations [3,10].

Existing modelling approaches expose a trade-off between physical fidelity and computational efficiency. Mechanics-based formulations derived from Cosserat-rod theory can accurately capture distributed deformation, including torsion, shear, and load-dependent effects [11,12], but require iterative numerical integration that becomes expensive for dense workspace exploration or repeated IK solving. Geometric models such as piecewise constant curvature (PCC) offer closed-form kinematics and scale well to optimization-based IK and large-scale sampling [1,13]. In this study, a disk-wise PCC model is adopted as a computationally tractable forward model for dense offline CAP construction over 54,000 voxel-orientation targets solved through repeated PSO-based IK queries. This choice is not intended to replace higher-fidelity Cosserat or mechanics-aware models; rather, such models can be integrated into the same CAP framework in future work at increased offline computational cost.

Capability maps (CAPs) provide a complementary framework for organizing robot performance over the workspace by discretizing task space into volumetric cells, called voxels, and linking each cell not only to reachability and manipulability, but also to feasible robot configurations, orientation information, and actuation-related parameters [14,15]. For rigid manipulators, CAPs have been successfully used for base placement, workspace analysis, and motion planning through fast access to precomputed performance information [16,17]. Extending capability maps to TDCRs introduces additional challenges compared with rigid-link manipulators. In TDCRs, geometric reachability does not necessarily imply physical feasibility, because a reachable task-space pose may require tendon-length variations or antagonistic actuation commands that are inconsistent with the actual transmission architecture. Constructing a TDCR capability map therefore requires solving many high-dimensional, nonconvex IK problems over discretized workspace voxels and sampled end-effector orientations, while retaining only configurations that satisfy both pose-accuracy and tendon-feasibility requirements. Moreover, reachability alone is not sufficient for configuration selection, since multiple feasible CAP entries may coexist within the same voxel while exhibiting different residual and dexterity properties. This creates an intra-voxel, or voxel-level, selection problem, in which one representative precomputed configuration must be selected from multiple feasible CAP entries associated with the same voxel; this motivates the residual–dexterity Pareto criterion proposed in this work. Recent studies have further advanced continuum-robot modelling and control through Cosserat-based treatment of large-deflection TDCRs, optimization-based kinematic formulations for multi-segment cable-driven robots, and compliance-oriented modelling of continuum structures [18–20]. These developments mainly address modelling fidelity, control, or mechanical characterization. In contrast, the gap considered in this work concerns the organization of multiple feasible TDCR configurations in a configuration-level capability database and their rapid selection according to both pose residual and local dexterity. The proposed framework therefore complements, rather than replaces, mechanics-based modelling and control approaches.

A preliminary version of this work was reported in [21], where the focus was on task-space sampling, PSO-based IK, and voxel-level capability-map generation using reachability and success-ratio information. The present journal submission extends that preliminary study into a configuration-level CAP database with stored residuals, tendon-command mapping, Jacobian-based dexterity enrichment, Pareto/ideal-point selection, and motion-capture validation. Table 1 summarizes the main differences between [21] and the present work.

Accordingly, this paper proposes a unified, kinematics-centered, database-driven capability-map framework tailored to tendon-driven continuum robots. The main contributions are summarized as follows:

I. A PSO-based disk-wise PCC inverse-kinematics pipeline combined with tendon-length and antagonistic command mapping, linking optimized PCC configurations to prototype-level motor commands during dense offline sampling.
II. A configuration-level CAP database that stores precomputed PCC configurations, voxel-orientation target information, achieved poses, residual errors, and tendon-command information, enabling fast lookup-based retrieval without online inverse-kinematics solving.

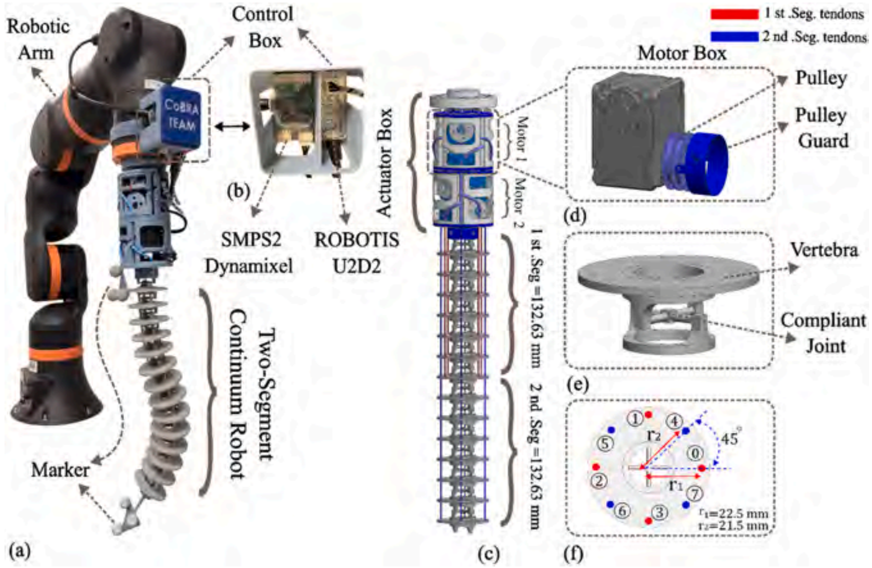

Fig. 1. Experimental platform and mechanical design of the two-segment TDCR: (a) integrated robotic setup, where the robotic arm is used only as a passive support/positioning structure and is not considered in the proposed modelling or CAP-based configuration-selection framework; (b) control and interface hardware; (c) CAD model with segment dimensions; (d) actuation unit and pulley system; (e) compliant joint; (f) tendon routing geometry with radial offsets and 45° spacing.

III. An offline Jacobian-enrichment procedure based on central finite differences and SO(3) logarithmic mapping, providing 6D dexterity descriptors for each stored CAP configuration without requiring closed-form derivatives of the disk-wise PCC model.
IV. A voxel-level Pareto/ideal-point configuration-selection strategy that ranks candidate CAP entries using combined pose residual and inverse manipulability and returns a deterministic dexterity-aware representative configuration for the queried local neighbourhood.
V. A motion-capture-based hardware evaluation on a custom two-segment TDCR, demonstrating the open-loop execution of a representative CAP-selected configuration with millimetre-order position errors in the evaluated case.

2. System description

A laboratory-scale custom two-segment tendon-driven continuum robot (TDCR) was used as a proof-of-concept platform to validate the proposed CAP framework under controlled experimental conditions. The platform integrates (i) a compliant mechanical structure, (ii) a compact base-mounted actuation module, and (iii) an optical motion-capture system used for external measurement of the end-effector pose during validation.

2.1. Mechanical structure

The robot comprises two identical serial continuum segments, indexed *by* $i \in \{1, 2\}$, each 132.63 mm long, yielding a total active backbone length of 265.26 mm. The central compliant backbone provides bending flexibility while maintaining axial stiffness (Fig. 1(e)). Fourteen rigid vertebrae (N=14), uniformly distributed along the backbone (7 per segment), support the piecewise constant curvature (PCC) modelling assumption (Fig. 1(c)). Eight tendons are routed symmetrically around the backbone (four per segment). In the first segment, tendons are arranged along the cardinal directions (90° spacing), while in the second segment the same four-tendon pattern is rotated by 45° relative to the first (Fig. 1(f)). The continuum structure is fabricated monolithically from PA12 Nylon via selective laser sintering (SLS) to ensure lightweight durability during repeated actuation.

2.2. Actuation module

Actuation is provided by two DYNAMIXEL 2XL430-W250-T dual-axis servomotors, delivering four independent output shafts in a compact base unit. Each shaft drives a custom pulley ($r_{pulley} = 10.5\ mm$), with two tendons wound in opposite directions to form an antagonistic pair, enabling bidirectional tendon control. Referring to Fig. 1(f), segment 1 is driven by tendon pairs (0,2) and (1,3) on the two shafts of Motor 1, while the segment 2 uses pairs (4,6) and (5,7) on Motor 2. Pulley guards are incorporated to prevent tendon overlap and to maintain consistent routing during bidirectional motion (Fig. 1(d)). Motor commands are issued from MATLAB through the DYNAMIXEL communication interface.

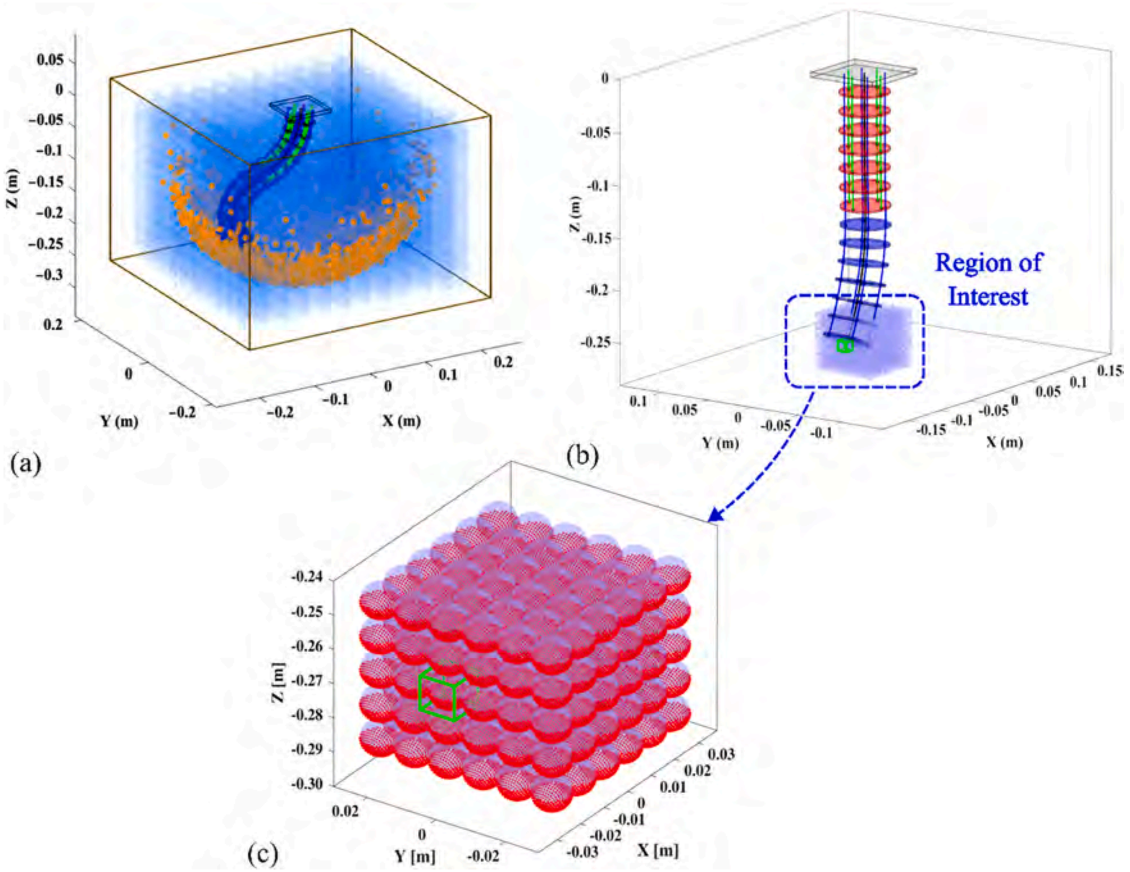

Fig. 2. Workspace sampling and kinematic modelling of the two-segment TDCR: (a) indicative PCC-based reachable-tip envelope, where the blue region visualizes the approximate sampled envelope and the orange markers show a down-sampled subset of tip samples for clarity; (b) distal region of interest (ROI) in the robot workspace; (c) voxelized ROI with spherical neighbourhoods.

2.3. Sensing and calibration

End-effector poses are measured using a multi-camera infrared OptiTrack system (nominal sub-millimetre precision), with a rigid triangular marker cluster mounted at the distal tip and a temporary base marker cluster used for frame registration (Fig. 1(a)). Calibration aligns the optical tracking frame to the robot base by recording the straight, zero-actuation configuration as the reference state. Incremental motor commands are then applied to refine the mapping between motor motion and effective tendon-length variation, accounting for pulley tolerances and initial tendon slack. This procedure improves the consistency of open-loop execution of CAP-derived configurations and supports comparison against PCC predictions.

3. CAP modelling framework

This section describes the kinematics-centered pipeline used to construct the capability map of a two-segment TDCR. As a preliminary step, the robot workspace is explored through PCC-based forward sampling to provide an indicative global view of the distal reachable envelope, as shown in Fig. 2(a). In this panel, the blue region denotes the approximate forward-sampled reachable-tip envelope, while the orange markers show a down-sampled subset of sampled tip positions for visualization clarity rather than the full sampling density. Based on this envelope, a task-relevant region of interest (ROI) is defined in the distal portion of the workspace, where the tool tip is expected to operate during local positioning tasks, as shown in Fig. 2(b,c). The selected ROI is not intended to represent the full reachable workspace of the TDCR. Instead, it provides a controlled local workspace in which dense voxel–orientation sampling, database construction, dexterity analysis, and hardware validation can be performed with sufficient resolution. The same CAP construction and selection principles can be extended to larger workspace regions, at the cost of increased offline computation and database size.

As described in detail in the following subsections, the CAP is then generated offline by solving a dense set of inverse-kinematics (IK) queries over the discretized ROI. For each voxel–orientation sample, a successful query generates a database entry containing the optimized disk-wise PCC configuration, the target and achieved poses, the associated residual errors, and the optimization cost, which is later enriched with Jacobian-derived dexterity metrics.

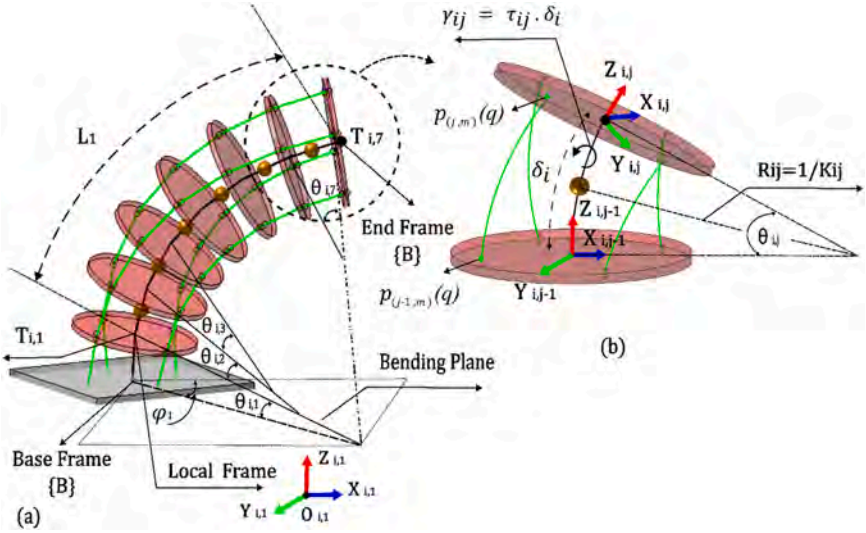

Fig. 3. (a) disk-wise PCC model of one segment with bending and twist; (b) local inter-vertebra geometry showing the element length δ_j, bending angle θ_j, and curvature radius $R_{ij} = 1/\kappa_{ij}$.

3.1. Forward kinematics

The TDCR is modelled using a disk-wise piecewise constant curvature (PCC) representation in which each inter-vertebral element *j* is described by a constant curvature κ_j, a bending-plane angle φ_j, and constant twist rate τ_j over its length δ_j. A base frame{B} is attached to the robot base, and the end-effector frame{E} is attached to the distal tip of the TDCR (Fig.3(a-b)). In the calibrated setup, the straight configuration is aligned with the negative global Z-axis.

The configuration of vertebra *j* is parameterized by curvature κ_j, bending-plane angle φ_j, and twist rate τ_j :

$$\boldsymbol{q_j} = \left[\ \kappa_j\ \varphi_j\ \tau_j\right]^T, \boldsymbol{q} = \left[\ \boldsymbol{q}_1^T, \ldots., \boldsymbol{q}_N^T\right]^T \mathbb{R}^{3N},\ j = 1, ., 14$$

with

$$\kappa_j \in [-25, 25] m^{-1}\ ,\ \varphi_j \in [0, 2\pi]\ rad\ ,\ \tau_j \in [-6, 6] rad/m \tag{1}$$

The corresponding bending angle and twist increment over the vertebra are $\theta_j = \kappa_j \delta_j$ and $\gamma_j = \tau_j \delta_j$, respectively. The homogeneous transform from vertebra *j−1* to vertebra *j* is written as:

$${}^{j-1}\boldsymbol{T_j}\left(\boldsymbol{q_j}\right) = \begin{bmatrix} {}^{j-1}\boldsymbol{R_j} & {}^{j-1}\boldsymbol{P_j} \\ 0_{1\times 3} & 1 \end{bmatrix} \tag{2}$$

The rotation is constructed by first aligning the bending plane via $\boldsymbol{R}_z(\varphi_j)$, applying a planar bend $\boldsymbol{R}_y(\theta_j)$, rotating back to the local frame, and finally applying an axial twist $\boldsymbol{R}_z\left(\gamma_j\right)$.

$${}^{j-1}\boldsymbol{R}_j = \boldsymbol{R}_z(\varphi_j)\boldsymbol{R}_y(\theta_j)\boldsymbol{R}_z(-\varphi_j)\boldsymbol{R}_z\left(\gamma_j\right) \tag{3}$$

For $|\kappa_j|$ sufficiently small, the element is treated as straight with ${}^{j-1}\boldsymbol{P_j} = [0, 0, \delta]^{\mathrm{T}}$.
Otherwise,

$${}^{j-1}\boldsymbol{P_j} = \boldsymbol{R}_z(\varphi_j) \begin{bmatrix} \dfrac{1 - \cos(\theta_j)}{\kappa_j} \\ 0 \\ \dfrac{\sin(\theta_j)}{\kappa_j} \end{bmatrix} \tag{4}$$

The overall end-effector transformation is obtained via serial chaining of the vertebra-level homogeneous transforms along each segment, as formalized by Eq. (5).

$${}^{\mathbf{B}}\mathbf{T}_{\mathbf{E}}(\mathbf{q}) = \prod_{j=1}^{N} {}^{j-1}\mathbf{T}_j\left(\mathbf{q}_j\right) \tag{5}$$

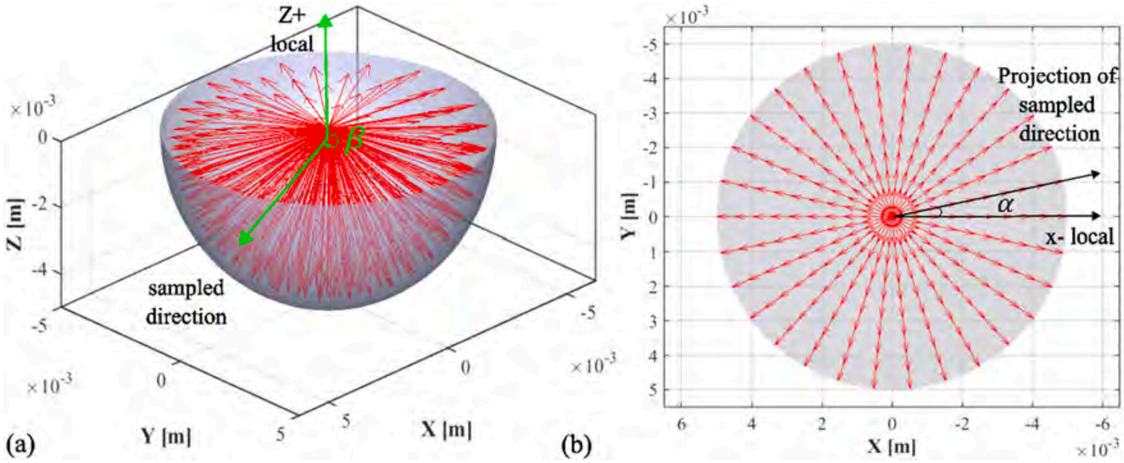

Fig. 4. (a) regular lower-hemisphere sampling of candidate tool-axis directions for one voxel; (b) top view of the 300 candidate tool-axis directions.

During CAP construction, a consistent tool-direction convention is enforced by sampling target orientations in the downward hemisphere. Accepted solutions are required to satisfy $\mathbf{R} \in SO(3)$, but the sampling is restricted to downward tool-axis directions. For each sampled direction, a target orientation matrix is generated offline using a Frenet-like frame construction, where the end-effector z-axis aligns with the sampled direction, the x-axis is obtained by projecting a fixed reference axis onto the plane orthogonal to z, and the y-axis is defined by the cross product. The resulting frame is orthonormalized to ensure membership in SO(3) and a consistent roll convention across samples.

3.2. Sampling-based CAP construction

The CAP is constructed over a distal region of interest (ROI) discretized into a Cartesian voxel grid (Fig. 2(b)). To obtain a compact, overlap-free neighbourhood representation for each voxel, we associate every voxel center with an inscribed spherical neighbourhood of fixed radius $r_{sphere} = 5\ mm$. Accordingly, voxel centers are sampled on a uniform grid with spacing $2 \cdot r_{sphere}$ over $x, y \in [-25, 25]$ mm and $z \in [\mathrm{L} - 20, \mathrm{L} + 20]$ mm, where L denotes the nominal tip position in the straight configuration under the adopted z-down convention. The additional axial margin around L is introduced to cover a small distal neighbourhood around the nominal straight-tip location and to avoid excluding nearby feasible samples due to local bending-induced shifts in the reachable tip position. The adopted ROI discretization yields N_v=180 voxels in the considered region (Fig. 2(c)). Let v denote the voxel index and s the sampled direction index. Each CAP query is therefore represented by the target pose $(\mathbf{p}_v, \mathbf{R}_{v,s})$, and each retained database entry is indexed by its voxel and sampled-orientation identifiers. The voxel-level candidate set and runtime ranking procedure are defined in Section 5.

For each voxel center, candidate tool-axis directions are generated by regular lower-hemisphere sampling, with $N_\alpha = 30$ uniformly spaced azimuth values in $[0,2\pi]$ and $N_\beta = 10$ uniformly spaced inclination values in $[\pi/2,\pi]$, yielding $N_\alpha N_\beta = 300$ directions per voxel (Fig. 4). These directions are converted offline into full target poses using the frame-construction rule described in Section 3.1. Overall, the ROI contains 180 voxels and 54,000 voxel-orientation targets. A voxel-level reachability pre-check is applied before PSO-based IK, allowing targets associated with unreachable voxels to be skipped.

3.3. PSO-based inverse kinematics

For each target pose $(\mathbf{p}_v, \mathbf{R}_{v,s})$, the IK problem is formulated as a bound-constrained optimization in $\mathbb{R}^{3N}$, $q^* = \arg \min_{q \in \mathbb{R}^{3N}} \mathscr{J}_{IK}(\mathbf{q})$. The cost function $\mathscr{J}_{IK}$ combines position error and the geodesic orientation distance on SO(3):

$$\mathscr{J}_{IK}(\mathbf{q}) = \| \mathbf{p}(\mathbf{q}) - \mathbf{p}_{v,s} \|_2 + \lambda_{ang} dist_{SO(3)}(\mathbf{R}(\mathbf{q}), \mathbf{R}_{v,s})$$

$$\mathrm{dist}_{SO(3)}(\mathbf{R}_1, \mathbf{R}_2) = \arccos\left(\frac{\mathrm{trace}(\mathbf{R}_1^{\mathrm{T}} \mathbf{R}_2) - 1}{2}\right), \quad \lambda_{ang} = 0.25\, L, \quad L = \sum_{j=1}^{N} \delta_j \tag{6}$$

where $\boldsymbol{p(q)}$ and $\mathbf{R}(\mathbf{q})$ are extracted from ${}^{\mathbf{B}}\mathbf{T}_{\mathrm{E}}(\mathbf{q})$, and L denotes the total backbone length. The coefficient λ_{ang} has dimensions of length and converts the SO(3) geodesic orientation error, evaluated in radians, into an equivalent translational contribution. In this study, $\lambda_{ang} = 0.25\, L$, where L is the active backbone length, so that the position-orientation weighting scales with robot size; for other robots or task requirements, the dimensionless multiplier of L can be adjusted according to the desired position-orientation tolerance ratio. The same λ_{ang} is used consistently in the IK cost, the residual measure, and the Jacobian scaling; angular errors are reported in degrees only for readability.

A Particle Swarm Optimization (PSO) solver is employed to handle the high dimensionality and nonconvexity typical of TDCR IK [16,17]. For all voxel–orientation targets, the solver uses the configuration bounds defined in Eq. (1), a swarm size of 50 particles, a maximum of 150 iterations, and a function tolerance of 10^{-6}. The same solver settings and acceptance thresholds are used for all IK queries to ensure consistent CAP database generation.

For acceptance checking, the translational and rotational residuals are defined as

$$e_p = \| \mathbf{p}(\mathbf{q}^*) - \mathbf{p}_{v,s} \|_2, \quad e_R = \mathrm{dist}_{SO(3)}\left(\mathbf{R}(\mathbf{q}^*), \mathbf{R}_{v,s} \right) \tag{7}$$

A solution is accepted and stored in the CAP database only if $e_p \le 5$ *mm*, and $e_R \le 10^{\circ}$, where the latter is applied internally as $e_R \le \pi/18$ rad.

3.4. Actuation mapping

For each feasible sample, the CAP database stores the target pose, the optimized configuration vector $\mathbf{q}^*$, the achieved pose, and the associated residual errors. An additional actuation-level consistency step is then performed by evaluating tendon-length variations with respect to the antagonistic pulley-driven architecture of the prototype. Accordingly, the retained CAP samples should be interpreted as command-compatible under the adopted tendon-routing and command-mapping model.

Specifically, the deformed tendon length $\ell_m(\boldsymbol{q}^*)$ is computed for each tendon $m \in \{0,\ldots,7\}$, from the vertebra-level tendon guide points and their consecutive distances along the backbone. The tendon length and the corresponding variation with respect to the rest configuration are defined as:

$$\ell_m(\mathbf{q}) = \sum_{j=1}^{N} \| \mathbf{p}_{\{j,m\}}(\mathbf{q}) - \mathbf{p}_{\{j-1,m\}}(\mathbf{q}) \|_2, \ \Delta\ell_m(\mathbf{q}) = \ell_m(\mathbf{q}) - \ell_m^{rest} \tag{8}$$

where ℓ_m^{rest} denotes the tendon rest length used as the actuation reference (Fig. 3(b)). In the present prototype, this reference length depends on the segment to which the tendon belongs, for segment 1, $\ell_m^{rest} = 135$ *mm* and for segment 2, $\ell_m^{rest} = 275$ *mm*. In the present implementation, actuator-level effects such as tendon slack, tension feasibility, hysteresis, and tendon elasticity are not imposed as hard constraints during the PSO-based IK search; instead, tendon-length variations are computed after the PCC solution is obtained as an actuation-mapping step.

These tendon-length variations define the actuation inputs associated with the corresponding robot configuration. For experimental execution on the motor-driven prototype, the linear tendon displacements are converted into motor encoder counts. Given the pulley radius r_{pulley} and the motor resolution $N_{\mathrm{enc}} = 4096$ counts/rev, the command u_m associated with tendon m is computed as:

$$u_m = \mathrm{round}\left(\frac{\Delta\ell_m(\mathbf{q})}{\mathrm{r_{pulley}}} . \frac{\mathrm{N_{enc}}}{2\pi} \right) \tag{9}$$

Since each motor shaft drives an antagonistic tendon pair, the final shaft command is obtained by applying equal and opposite tendon displacements within the corresponding pair. In this way, each retained CAP sample is linked to prototype-level tendon and motor commands, providing command-level compatibility with the implemented actuation model rather than a guarantee of exact disk-wise shape reproduction on the underactuated physical robot.

4. Differential kinematics and dexterity evaluation

To complement the CAP database of feasible configurations, each accepted sample is enriched offline with differential kinematic information so that local dexterity can be incorporated into the subsequent configuration-selection stage. For a given configuration $\mathbf{q} \in \mathbb{R}^{3N}$, the forward kinematics provides the end-effector pose ${}^{\mathbf{B}}\mathbf{T}_{\mathbf{E}}(\mathbf{q})$, from which the position $\mathbf{p}(\mathbf{q}) \in \mathbb{R}^3$ and the orientation $\mathbf{R}(\mathbf{q}) \in SO(3)$ are extracted. The differential relationship between generalized velocity $\dot{q}$ and end-effector twist is expressed as:

$$\mathbf{J}(\mathbf{q}) = \begin{bmatrix} \mathbf{J_p}(\mathbf{q}) \\ \mathbf{J}_w(\mathbf{q}) \end{bmatrix} \in \mathbb{R}^{6\times 3N}, \mathbf{J}(\mathbf{q})\mathbf{q} = \begin{bmatrix} \mathbf{v}(\mathbf{q}) \\ \mathbf{w}(\mathbf{q}) \end{bmatrix} \tag{10}$$

where $\boldsymbol{v} \in \mathbb{R}^3$ and $\boldsymbol{w} \in \mathbb{R}^3$denote translational and angular velocity, respectively.

4.1. Finite-difference jacobian computation

Since closed-form PCC derivatives become complex for the adopted disk-wise parameterization, the Jacobian is computed numerically from the forward kinematic map. Finite-difference perturbations are applied directly to the PCC configuration vector, without re-applying the target-frame orientation convention during differentiation, to avoid artificial discontinuities in the local linearization. The component-wise perturbation steps were set to $\delta_\kappa = 5 \times 10^{-5} m^{-1}$, $\delta_\varphi = 10^{-4}$ rad; $\delta_\tau = 10^{-5}$ rad/m; for near-zero curvature values, i.e., $|\kappa| < 10^{-6}$, a slightly larger perturbation of $\delta_\kappa = 2.5 \times 10^{-4}\ m^{-1}$was used to avoid numerical degeneracy around the straight configuration. For the n_{th} variable, with basis vector $\boldsymbol{e}_n$ and perturbation step δ_n, the translational Jacobian column is approximated as:

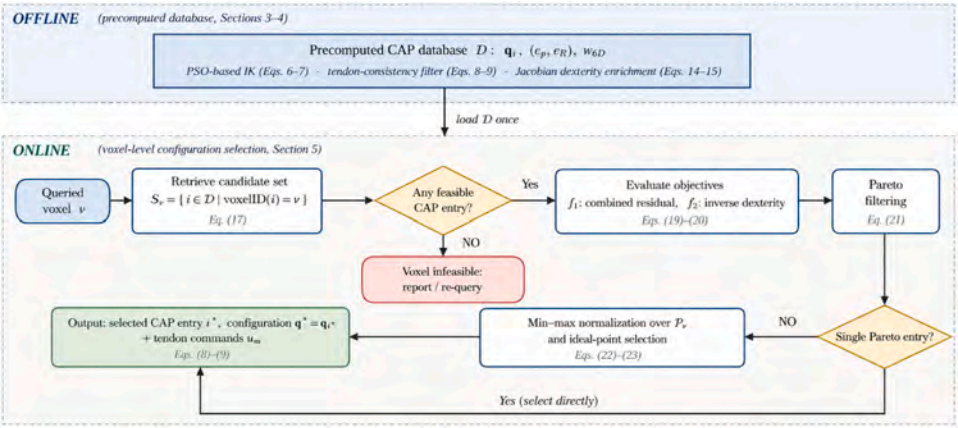

Fig. 5. Flowchart of the proposed database-driven bi-objective CAP selector, showing the transition from offline CAP construction to online voxel-level configuration selection.

$$J_{p(:,n)}(\boldsymbol{q}) \approx \frac{\mathbf{p}(\boldsymbol{q} + \delta_n \boldsymbol{e}_n) - \mathbf{p}(\boldsymbol{q} - \delta_n \boldsymbol{e}_n)}{2\delta_n} \tag{11}$$

For the rotational component, orientation differences are mapped on to the tangent space of SO(3) through the logarithm map. Defining:

$$\boldsymbol{\xi}_n^{\pm} = vee\left(\log\left(\boldsymbol{R}(\boldsymbol{q})^T \boldsymbol{R}(\boldsymbol{q} \pm \delta_n \boldsymbol{e}_n)\right)\right) \tag{12}$$

where vee $(\cdot)$ maps a skew-symmetric matrix in SO(3) to its vector representation. The corresponding rotational Jacobian column is approximated by

$$\mathbf{J}_{w(:,n)}(\boldsymbol{q}) \approx \frac{\boldsymbol{\xi}_n^{+} - \boldsymbol{\xi}_n^{-}}{2\delta_n} \tag{13}$$

This numerical procedure provides a consistent local linear approximation of the PCC kinematics for each accepted CAP sample and enables offline dexterity evaluation directly from the stored database configurations.

4.2. Dexterity metric

Once the Jacobian has been computed, each feasible CAP sample is assigned a scalar dexterity value. In the present framework, dexterity is quantified through a 6D manipulability measure, which is also the descriptor adopted in the subsequent bi-objective configuration selection. To make the translational and rotational contributions commensurate, the same scaling factor λ_{ang} introduced in Eq. (6) and discussed in Section 3.3 is retained in the scaled Jacobian. The scaled Jacobian is therefore defined as

$$\mathbf{J}_6(\mathbf{q}) = \begin{bmatrix} \mathbf{J}_p(\mathbf{q}) \\ \lambda_{ang}\mathbf{J}_w(\mathbf{q}) \end{bmatrix} \tag{14}$$

and the corresponding dexterity descriptor is computed as

$$w_{6D}(\boldsymbol{q}) = \sqrt{det\left(\mathbf{J}_6(\mathbf{q})\mathbf{J}_6(\mathbf{q})^{\mathrm{T}}\right)} \tag{15}$$

This quantity is used as the dexterity term in the bi-objective CAP-based selection. In addition, a positional manipulability descriptor is computed offline for descriptive analysis of the database:

$$w_p(\boldsymbol{q}) = \sqrt{det\left(\mathbf{J}_p(\mathbf{q})\mathbf{J}_p(\mathbf{q})^{\mathrm{T}}\right)} \tag{16}$$

The scalar descriptors w_{6D} and w_p provide compact task-independent summaries of local dexterity, but they do not distinguish the capability along individual task-space directions. In the implemented CAP database, the Jacobian-enrichment stage also stores direction-specific descriptors, including the singular values of the positional Jacobian $\mathbf{J}_p$, the positional condition number, and the directional translational and rotational gains along the tool-approach axis. For a prescribed unit direction $\mathbf{u}$, these descriptors can be expressed as $d_p(\boldsymbol{q},\boldsymbol{u}) = \| \mathbf{J}_p(\mathbf{q})^{\mathrm{T}}\boldsymbol{u} \|$ for translational capability and $d_r(\boldsymbol{q},\boldsymbol{u}) = \| \mathbf{J}_w(\mathbf{q})^{\mathrm{T}}\boldsymbol{u} \|$ for rotational capability, where $\mathbf{J}_w$ denotes the angular part of the Jacobian. These quantities can be used as additional CAP fields or as task-specific alternatives to the scalar dexterity objective in the Pareto selector. In the proposed CAP framework, w_{6D} is retained as the default metric to keep the selection task-independent.

5. Database-driven bi-objective configuration selection

To enable fast runtime configuration retrieval, all computationally expensive steps are performed offline during CAP construction. The proposed selector retrieves feasible precomputed CAP entries associated with a queried voxel and selects one dexterity-aware representative configuration from that local neighbourhood, rather than solving a new pose-conditioned IK problem online. The complete offline-online workflow is summarized in Fig. 5.

5.1. Candidate set and objective functions

Let D denote the offline CAP database. For a queried voxel v, the admissible candidate set is defined as:

$$\mathscr{S}_v = \{\mathrm{i} \in D \mid \mathrm{voxelID(i)} = \mathrm{v}\} \tag{17}$$

Each candidate $i \in \mathscr{S}_v$ is a tendon-feasible PCC configuration generated for the queried voxel and one sampled tool-axis direction. Thus, $\mathscr{S}_v$ contains orientation-diverse feasible entries within the same local neighbourhood, and the selector chooses one representative configuration $i^* \in \mathscr{S}_v$ by balancing stored pose quality and local dexterity. The bi-objective problem can be formulated as:

$$i^* = \arg \min_{\mathrm{i} \in \mathscr{S}_v} [f_1(i), f_2(i)]^T \tag{18}$$

The first objective evaluates the stored pose quality of each CAP entry through a combined residual measure, while the second objective promotes configurations with higher local dexterity. The same scaling factor λ_{ang} used in the offline IK stage is retained here so that the stored residual measure remains consistent with the objective according to which the feasible CAP entries were originally generated.

$$f_1(i) = e_P(i) + \lambda_{ang} e_R(i) \tag{19}$$

where $e_P(i)$ and $e_R(i)$ are the stored position and orientation residuals of candidate i. The angular residual $e_R(i)$ is evaluated in radians before multiplication by λ_{ang}, consistently with the IK cost in Eq. (6).

The second objective promotes configurations with higher local motion capability by minimizing the inverse of the precomputed 6D manipulability measure:

$$f_2(i) = \frac{1}{w_{6D}(i) + \varepsilon} \tag{20}$$

where $w_{6D}(i)$ is the offline dexterity metric associated with the same CAP entry, and $\varepsilon > 0$ is a small constant introduced to avoid numerical issues near singular configurations.

5.2. Pareto set extraction

Since f_1 and f_2 may conflict, the selection is limited to the non-dominated subset of $\mathscr{S}_v$ using Pareto dominance. A candidate "A" dominates "B" if

$$(f_1(A) \le f_1(B) \wedge f_2(A) \le f_2(B)) \wedge (f_1(A) < f_1(B) \vee f_2(A) < f_2(B)) \tag{21}$$

The resulting Pareto set $P_v \subseteq \mathscr{S}_v$ contains all admissible candidates that are not dominated by any other element. Since the entries in $\mathscr{S}_v$ may correspond to different sampled orientations within the same voxel, the Pareto set represents the attainable trade-offs between stored pose quality and dexterity at the voxel level.

5.3. Normalization and ideal-point selection

To obtain a single execution-ready configuration, a best-compromise solution is selected from P_v using an ideal-point criterion in normalized objective space. If the Pareto set contains only one candidate, that configuration is selected directly. Otherwise, each objective is min-max normalized over P_v as

$$\widetilde{f}_b(i) = \frac{f_b(i) - f_b^{min}}{f_b^{max} - f_b^{min}}, \ \mathrm{b} \in \{1, 2\} \tag{22}$$

where f_b^{min} and f_b^{max} denote the minimum and maximum values of f_b over the Pareto set. The final selection is then obtained by minimizing the Euclidean distance to the ideal point (0,0):

$$i^* = \arg \min_{\mathrm{i} \in P_v} \sqrt{\widetilde{f}_1(i)^2 + \widetilde{f}_2(i)^2} \tag{23}$$

This procedure yields a deterministic trade-off between stored pose quality and dexterity without introducing additional manual weights beyond the scaling already used in the IK formulation. Since all required quantities are precomputed, the online stage reduces

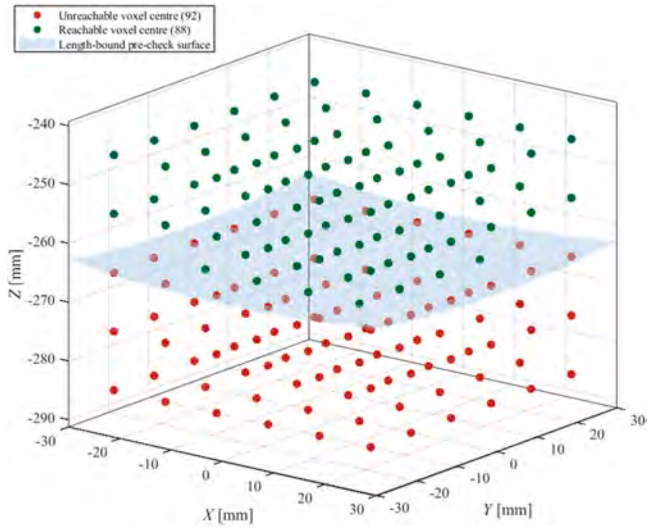

Fig. 6. Reachability classification of the ROI. Green and red markers denote reachable and unreachable voxel centres, respectively, while the semi-transparent surface indicates the PCC-based reachability boundary used for preliminary filtering.

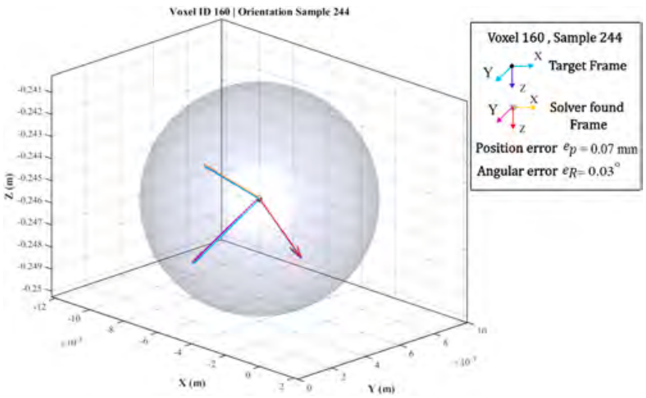

Fig. 7. Representative voxel-level IK result showing the target frame and the solver-found pose inside the spherical neighbourhood associated with the selected voxel.

to database filtering, Pareto extraction, normalization, and ideal-point evaluation.

6. Offline CAP analysis and configuration selection results

This section analyses the constructed CAP database, including ROI coverage, representative PSO-based IK outcomes, and the distribution of the dexterity measure, and then examines the resulting voxel-level Pareto selection behaviour prior to hardware validation.

6.1. ROI reachability and coverage

The offline CAP generation classifies the discretized ROI according to kinematic feasibility under the adopted backbone-length and curvature constraints. Among the 180 discretized voxels, 88 voxels (48.9%) satisfied this preliminary reachability condition, while the remaining 92 voxels (51.1%) were classified as unreachable. Fig. 6 shows the resulting reachability map, where green voxel centres denote the reachable subset and red voxel centres denote the unreachable subset. The semi-transparent surface in the figure indicates the PCC-based reachability boundary used for this preliminary classification. By excluding the 92 unreachable voxels before PSO-based IK, the pipeline skips 27,600 voxel-orientation queries. The reachable voxels are concentrated in the central and distal part of the ROI, while boundary regions show reduced accessibility, consistent with the ROI position near the reachable-workspace boundary.

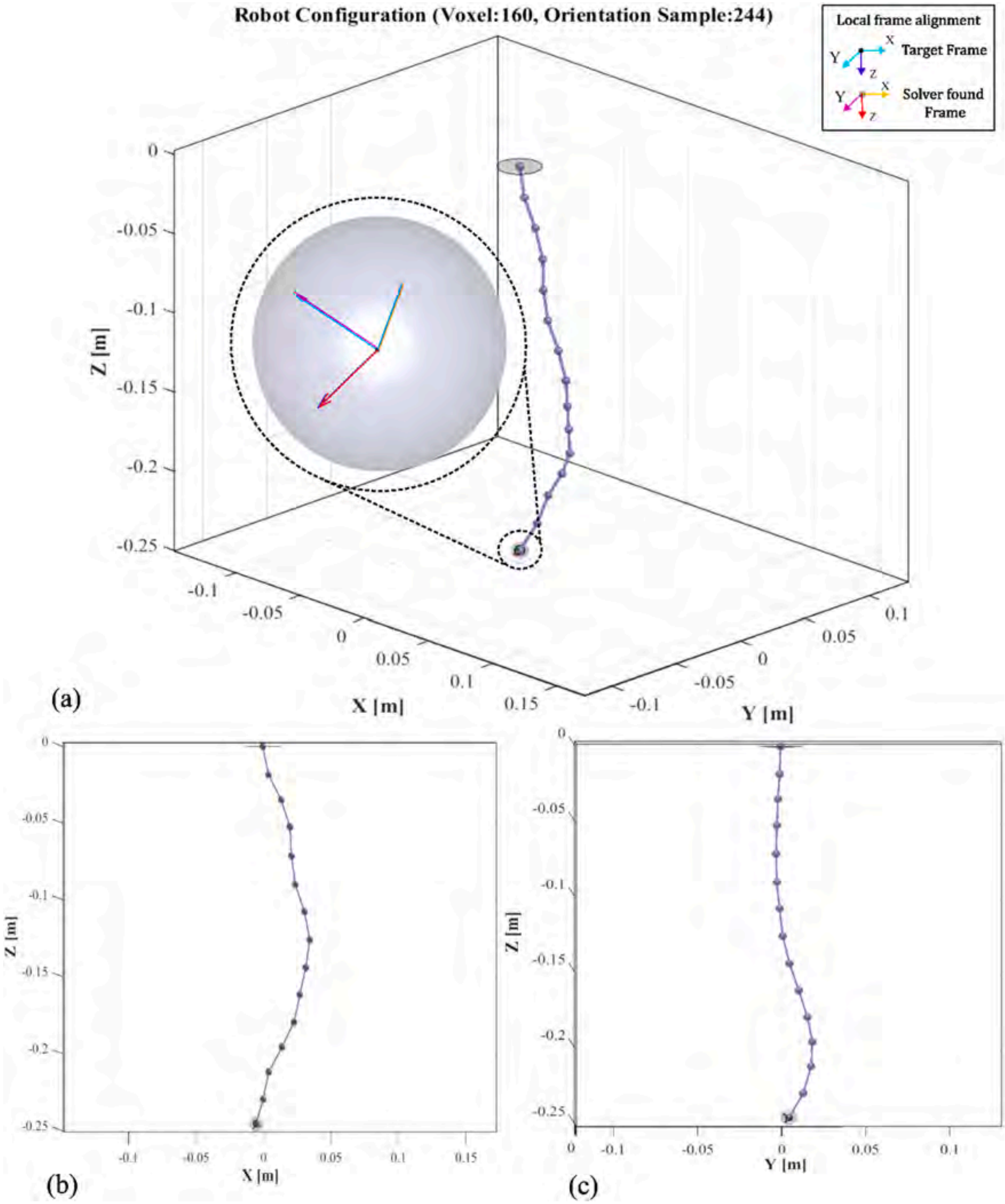

Fig. 8. Representative solver-found TDCR configuration (a) 3D reconstructed backbone with the selected voxel and local frame; (b) $x-z$ projection; (c) $y-z$ projection.

6.2. Representative PSO-based IK outcomes

The numerical behaviour of the PSO-based IK solver is illustrated through representative target samples distributed across the ROI. Figures 7 and 8 illustrate one representative example: Fig. 7 compares the target frame T_{target} with the solver-found frame T_{found}, while Fig. 8 shows the corresponding reconstructed backbone shape. The reconstructed shape is consistent with the adopted disk-wise PCC model, where each inter-vertebra element has its own local curvature, bending-plane angle, and twist parameter. Therefore, apparent differences in smoothness between the (x)-(z) and (y)-(z) projections can occur because the same three-dimensional piecewise-constant-curvature backbone is projected onto different coordinate planes.

Table 2
Representative raw IK outcomes for five sampled voxel–orientation targets before the CAP acceptance filtering.

Sample ID	Voxel ID	Orientation Sample (s)	Position Error $\|\| \boldsymbol{e_p} \|\|$ (mm)	Orientation Error $\|\| \boldsymbol{e_R} \|\|$ (deg)	Cost ($\mathscr{J}_{IK}(\boldsymbol{q})$)	Retained in CAP
7317	25	117	46.00	4.875017767	0.051647335	No
47944	160	244	0.071	0.030570594	0.000106306	Yes
50359	168	259	0.050	1.082488597	0.00130243	Yes
24060	81	60	23.30	19.1366543	0.045446632	No
20964	70	264	17.67	0.005494452	0.017675196	No

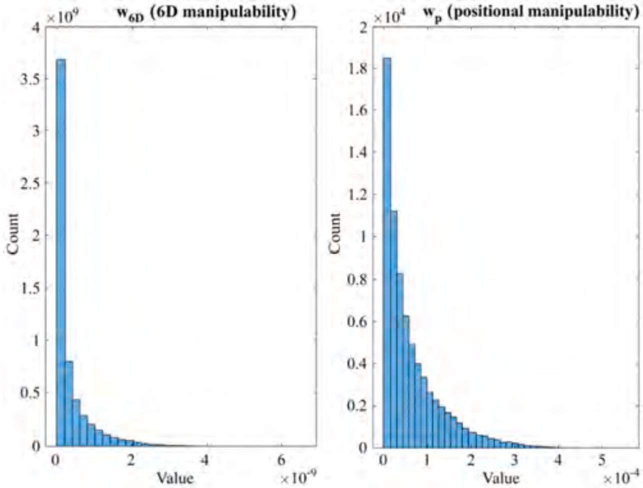

Fig. 9. Global distributions of the 6D manipulability measure w_{6D} and the positional manipulability measure w_p over all feasible CAP entries.

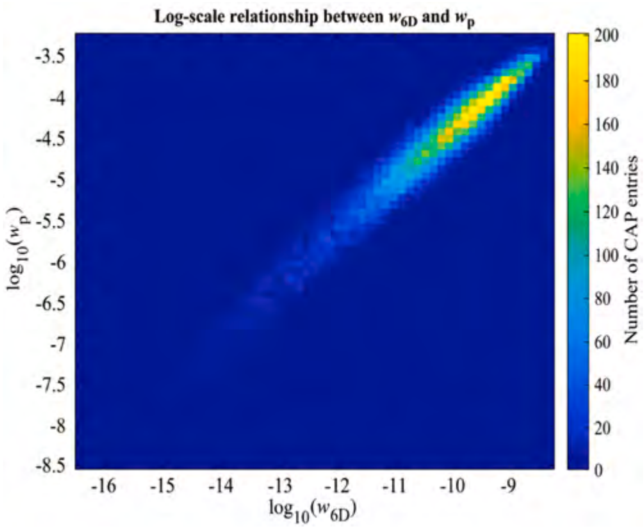

Fig. 10. Log-scale relationship between the 6D manipulability measure $\boldsymbol{w}_{6D}$ and the positional manipulability measure $\boldsymbol{w}_p$ over all feasible CAP entries.

Table 2 reports five representative raw PSO-based IK outcomes before CAP acceptance filtering. These examples illustrate the range of solver responses across the ROI; only solutions satisfying the position and orientation residual thresholds defined in Section 3.3 are retained in the final CAP database and used in subsequent selection analysis.

6.3. Distribution of the adopted dexterity metric

The offline Jacobian analysis was used to evaluate the dexterity distribution over the feasible CAP entries. The adopted 6D

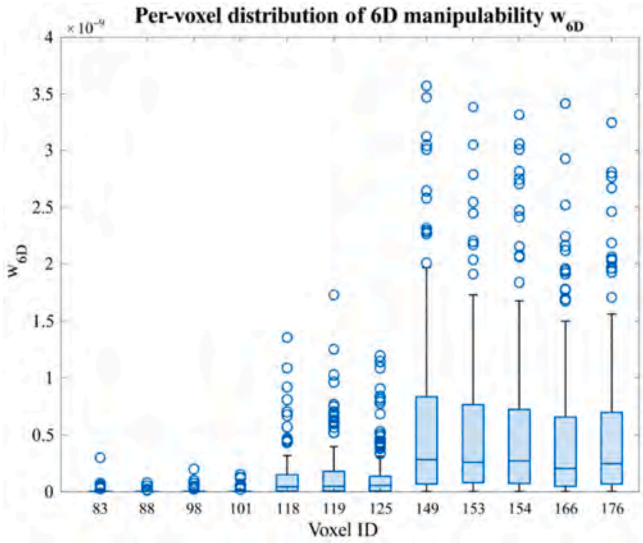

Fig.11. Distribution of the 6D manipulability measure w_{6D} over feasible CAP entries within selected workspace voxels.

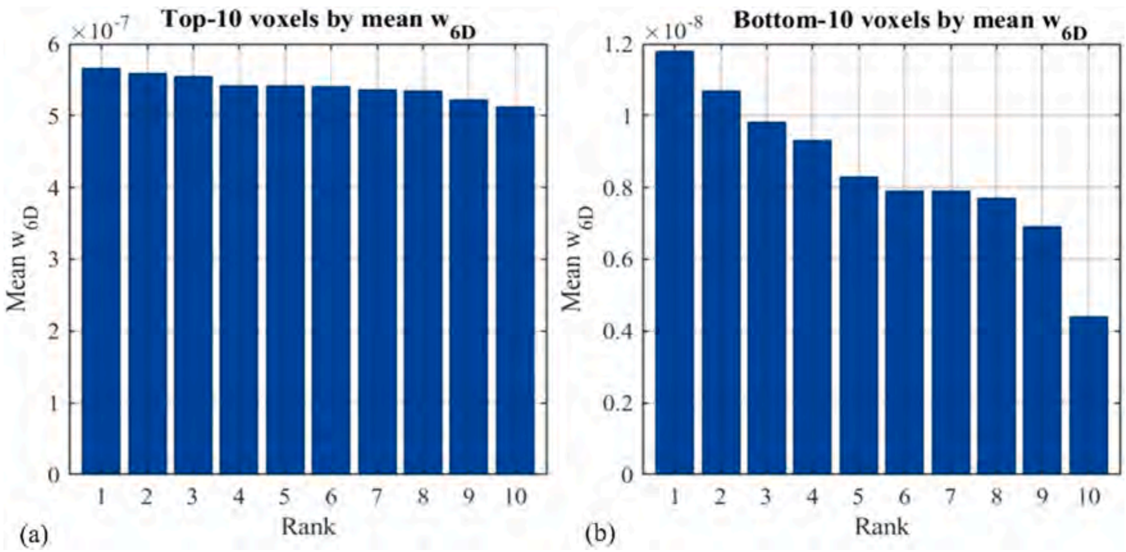

Fig.12. Ranking of workspace voxels based on the mean 6D manipulability measure w_{6D} (a) ten voxels with the highest mean w_{6D}; (b) ten voxels with the lowest mean.

manipulability measure, w_{6D}, is used as the main dexterity descriptor in the residual–dexterity selection criterion of Section 5, while the auxiliary positional manipulability, w_p , is reported only to illustrate the translational contribution. As shown in Fig. 9, both descriptors are highly non-uniform across the database, indicating that reachability alone does not imply comparable local motion capability. Fig. 10 further shows the feasible CAP entries in the (w_p, w_{6D}) plane. Although the two descriptors are positively correlated, their scatter confirms that w_p cannot replace w_{6D}, because the latter also includes rotational capability.

The stored singular values of the positional Jacobian also confirm that the retained configurations can be strongly anisotropic. Over the retained CAP entries, the median positional condition number, defined as σ_1/σ_3, is approximately 17.7, and about 79% of the entries have $\sigma_1/\sigma_3 > 10$. Therefore, w_{6D} should be interpreted as a compact task-independent descriptor of overall local dexterity, not as a complete representation of direction-specific capability; tasks with prescribed motion or approach directions can instead use direction-specific descriptors.

Dexterity varies not only across the workspace, but also among the feasible configurations stored within each voxel. As shown in Fig. 11, each boxplot represents the distribution of the 6D manipulability measure w_{6D} over all successful CAP entries associated with a selected voxel. Since these entries correspond to different internal PCC configurations reaching the same local task-space neighbourhood, the observed spreads and outliers indicate that pose feasibility alone does not guarantee comparable dexterity. This makes

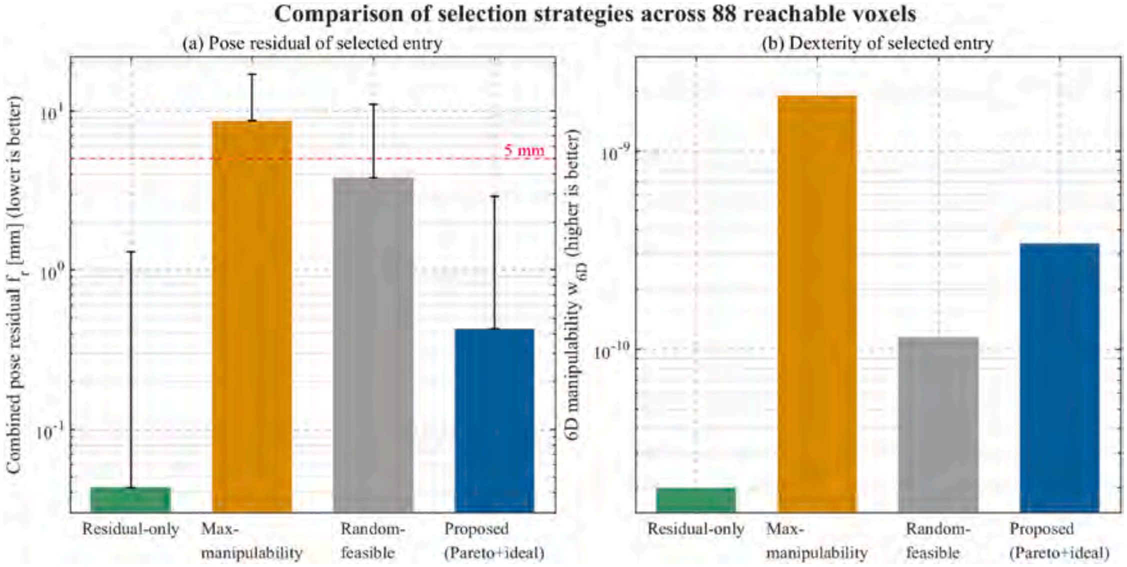

Fig. 13. Baseline comparison of selection strategies across 88 reachable voxels. Panel (a) shows the selected pose residual, and panel (b) shows the corresponding 6D manipulability.

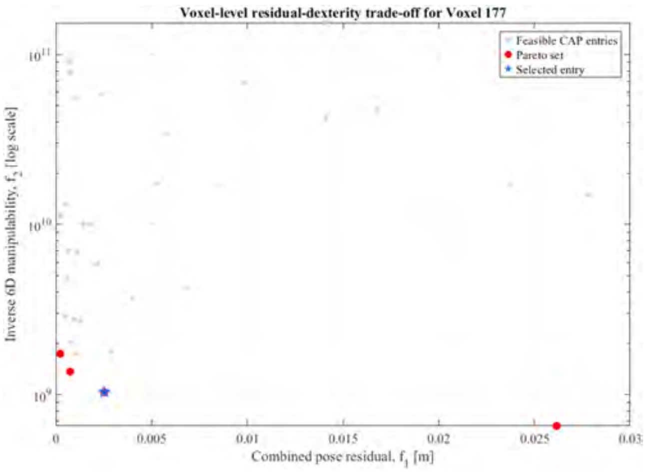

Fig. 14. Voxel-level residual-dexterity trade-off for Voxel 177. The axes show the combined pose residual $\boldsymbol{f}_1$ and inverse 6D manipulability $\boldsymbol{f}_2$. Gray points denote feasible CAP entries, red markers indicate the Pareto set, and the blue star marks the representative entry selected by the normalized ideal-point criterion.

configuration selection within a voxel nontrivial and motivates the use of the residual–dexterity Pareto criterion adopted in this work. Finally, Fig. 12 ranks workspace voxels according to their mean w_{6D}, highlighting the large difference between the most and least dexterous local regions of the CAP database.

6.4. Bi-objective selection analysis

The bi-objective analysis is performed at the voxel level using the candidate set defined in Section 5. Each point in the objective plots represents one feasible CAP entry within the queried voxel, evaluated by the combined residual f_1 and inverse manipulability f_2. The resulting Pareto set captures the non-dominated residual-dexterity trade-off, from which the normalized ideal-point criterion selects the representative configuration.

The proposed Pareto/ideal-point strategy was compared with three database-level baselines over the 88 reachable voxels: residual-only, maximum-manipulability, and random-feasible selection. As shown in Fig. 13, residual-only selection favours pose accuracy, maximum-manipulability selection favours dexterity, and random-feasible selection provides a non-deterministic reference. The proposed selector provides a deterministic compromise between pose quality and local dexterity.

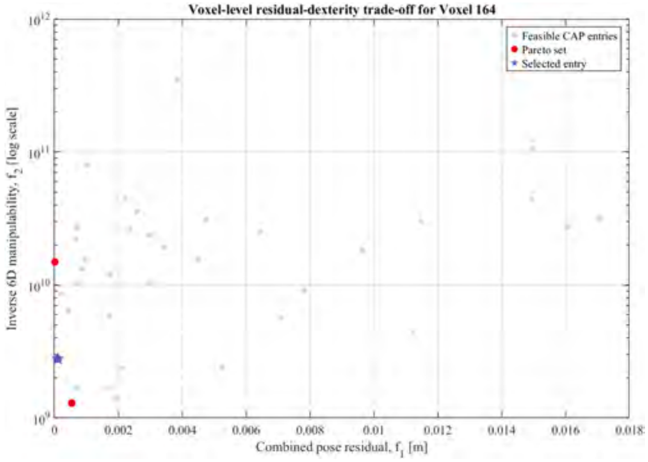

Fig. 15. Voxel-level residual–dexterity trade-off for Voxel 164, showing a sparser feasible CAP distribution relative to Voxel 177. Axes, markers, and the normalized ideal-point selection criterion follow Fig. 14.

Figs. 14 and 15 show two representative voxel-level selection cases for Voxels 177 and 164, respectively. Voxel 177 contains a richer candidate distribution with several competitive trade-offs, whereas Voxel 164 is sparser, reflecting more restrictive local kinematic conditions. In both cases, the selected entry corresponds to a best-compromise configuration within the voxel-level Pareto set.

The selected entries were further examined through their internal PCC parameters. Figs. 16 and 17 show the curvature, bending-plane angle, and twist profiles for Voxels 177 and 164, respectively, while Fig. 18 provides the corresponding reconstructed backbone configurations. Together, these results show that the CAP framework goes beyond reachability mapping by supporting dexterity-aware selection among feasible alternatives within a queried local neighbourhood.

6.5. Computational cost and online query time

Table 3 summarizes the computational quantities of the proposed offline–online framework. The offline stage includes reachability pre-filtering, PSO-based IK, and Jacobian-based dexterity enrichment, whereas the online stage performs only the voxel-level retrieval and ranking procedure described in Section 5.Because PSO is stochastic, the CAP database was generated using fixed random seeds and identical solver settings; each voxel–orientation query was solved as an independent single PSO run, without warm-starting, multiple restarts, or best-of-restarts selection. Accordingly, the reported retention rate denotes the database acceptance rate under the prescribed residual thresholds, not a pure PSO convergence-failure rate. The timings were obtained in MATLAB R2024b on an AMD Ryzen 7 5800H laptop with 16 GB RAM.

7. Experimental validation

This section provides a proof-of-concept hardware evaluation of the proposed CAP-based selection framework on the two-segment TDCR prototype. The validation focuses on one representative reachable voxel and compares the selected CAP entry, the solver-found pose, and the experimentally measured end-effector pose under open-loop tendon actuation.

7.1. Accuracy verification

Hardware validation was performed on Voxel 151, a representative reachable mid-workspace region with 184 accepted voxel-orientation entries in the CAP database. Although this voxel is located near the nominal straight-tip neighbourhood, the stored configurations are not purely straight-backbone solutions; for the executed CAP-selected sample, the accumulated disk-wise bending measure ($\sum_j |k_j|\delta_j$) is approximately 198 deg, indicating distributed internal bending rather than net axial rotation. The database-driven selector returned Sample 45141 as the best-compromise entry for this voxel according to the Pareto-based ideal-point ranking; consistent with the voxel-level formulation, this should be interpreted as the preferred executable entry for the queried neighbourhood rather than an exact-pose optimum. The corresponding tendon commands were executed on the prototype, and the resulting end-effector pose was measured using the OptiTrack system. Fig. 19 illustrates the experimental setup, the executed CAP-selected configuration, and the local comparison between the target, solver-found, and experimentally measured poses for the selected voxel.

Fig. 20 shows the initial and final experimentally measured poses together with the CAP target and solver-found poses in the common z-down frame. The start pose is included only as the reference configuration before actuation, while the main comparison concerns the end pose after execution. As seen in the figure, the final experimental pose moves from the remote start state toward the local neighbourhood of the CAP-predicted poses, providing a qualitative visualization of execution consistency.

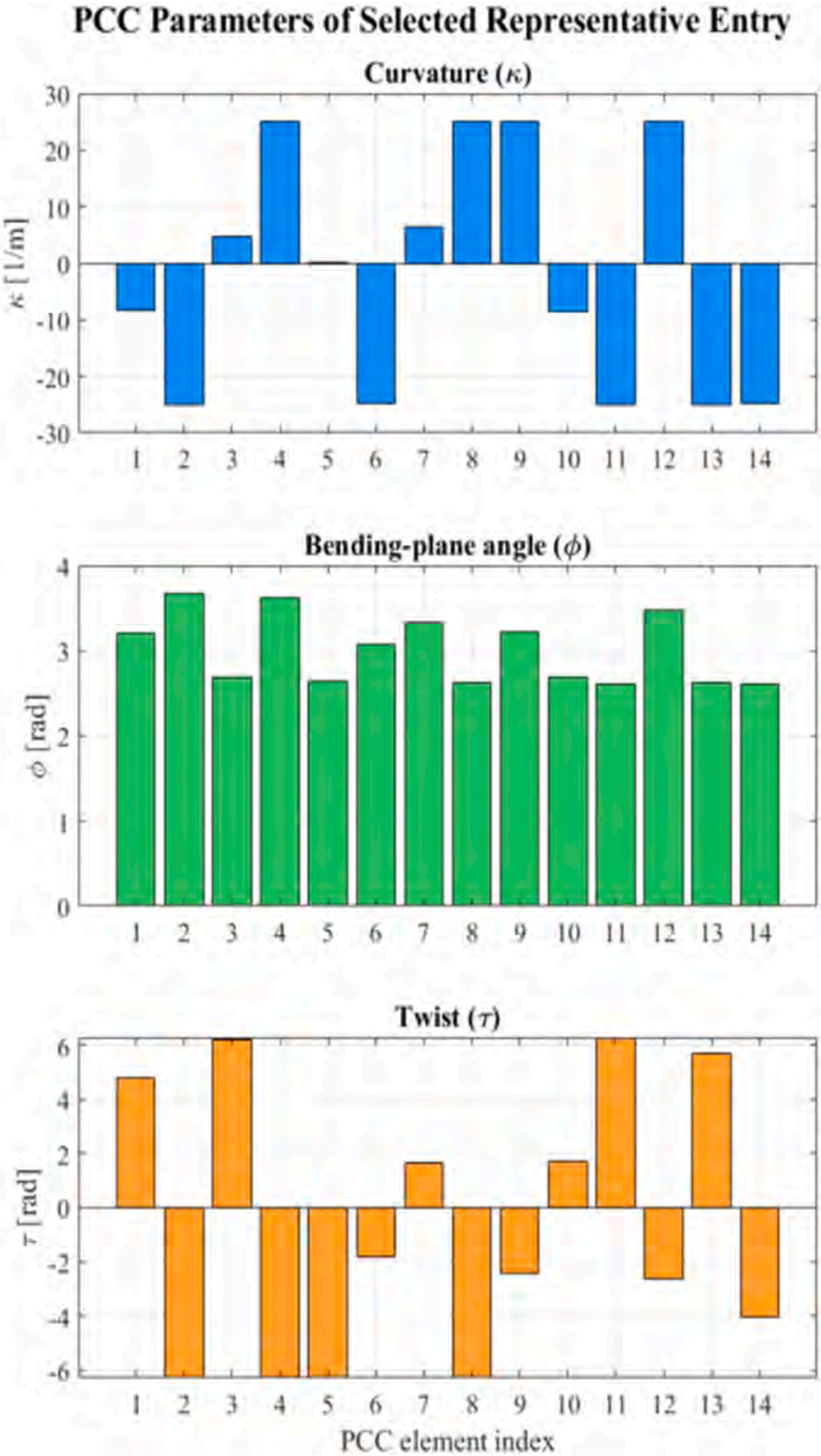

Fig. 16. PCC parameters of the representative CAP entry selected for Voxel 177.

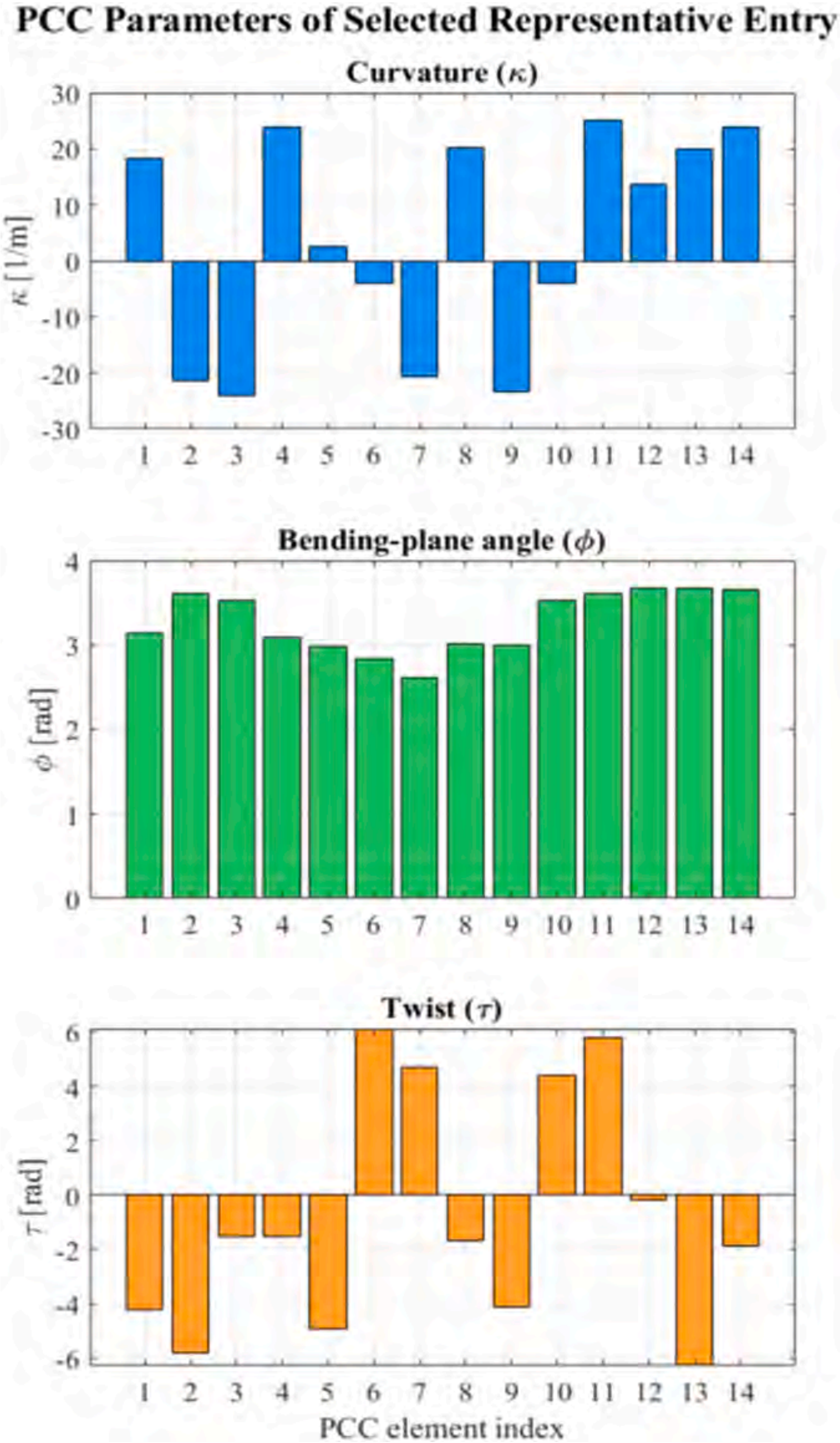

Fig. 17. PCC parameters of the representative CAP entry selected for Voxel 164.

For the quantitative comparison, the experimentally measured final pose is denoted by T_{end}, and its position and orientation errors are computed with respect to the nominal CAP target pose T_{target} and the stored solver-found pose T_{found}. The former corresponds to the ideal voxel–orientation target defined during CAP generation, while the latter corresponds to the pose actually returned by the offline IK solver and stored in the CAP database for the same entry.

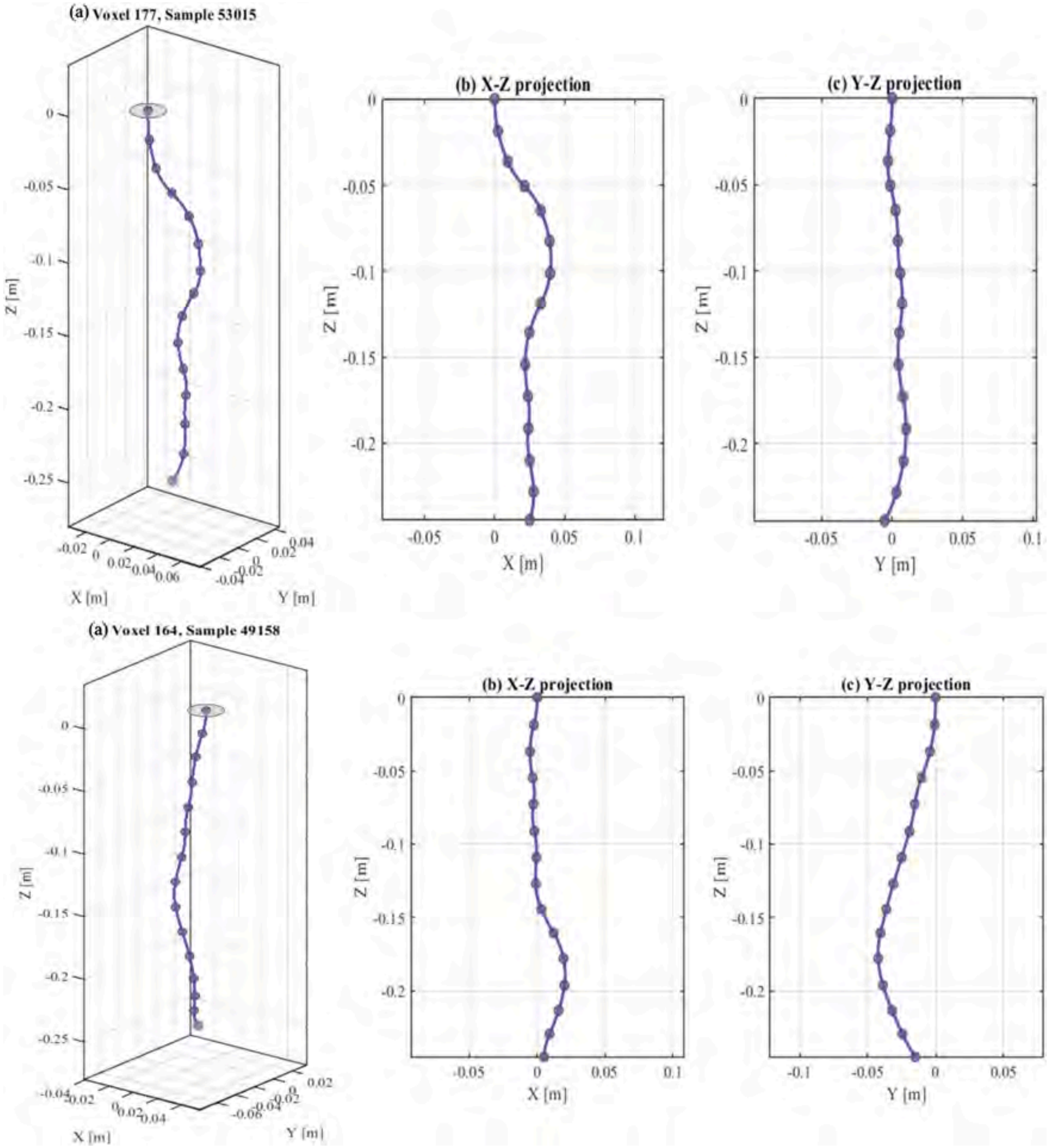

Fig. 18. Reconstructed backbone configurations corresponding to the selected representative CAP entries in Voxel 177 and Voxel 164. The 3D views and planar projections complement the PCC parameter profiles reported in Figs. 16 and 17.

Table 3
Computational-cost summary of the CAP-based selection framework.

Quantity	Value	Quantity	Value
Voxel-orientation targets	54,000	**Retained CAP entries**	13,585
Reachable / unreachable voxels	88 / 92	**Mean online CAP query time**	1.6 ms
Queries skipped by reachability pre-check	27,600	**Mean reference PSO-IK solve time**	2.2 s
Executed PSO-based IK queries	26,400	**Speed-up over PSO-IK**	1.4×10^3
Equivalent serial offline PSO-IK time	approximately 16.1 h	**CAP retention rate after executed IK**	13,585 /26,400 = 51.5%
CAP database memory footprint	66.1 MB	**Offline Jacobian-enrichment time**	approximately 12 min

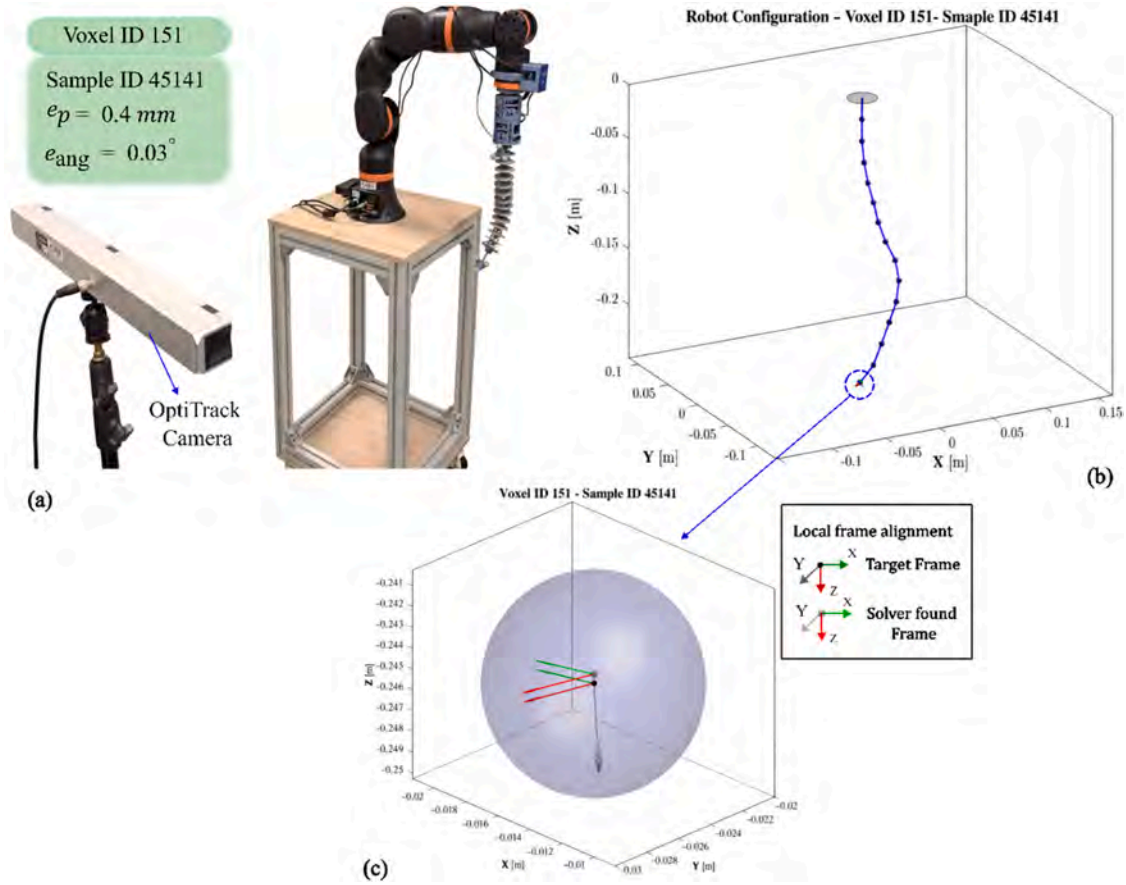

Fig. 19. Hardware validation of the selected CAP entry (a) TDCR–OptiTrack experimental setup; (b) Executed configuration for Voxel ID 151 and Sample ID 45141; (c) Target, solver-found, and experimentally measured poses within the selected voxel.

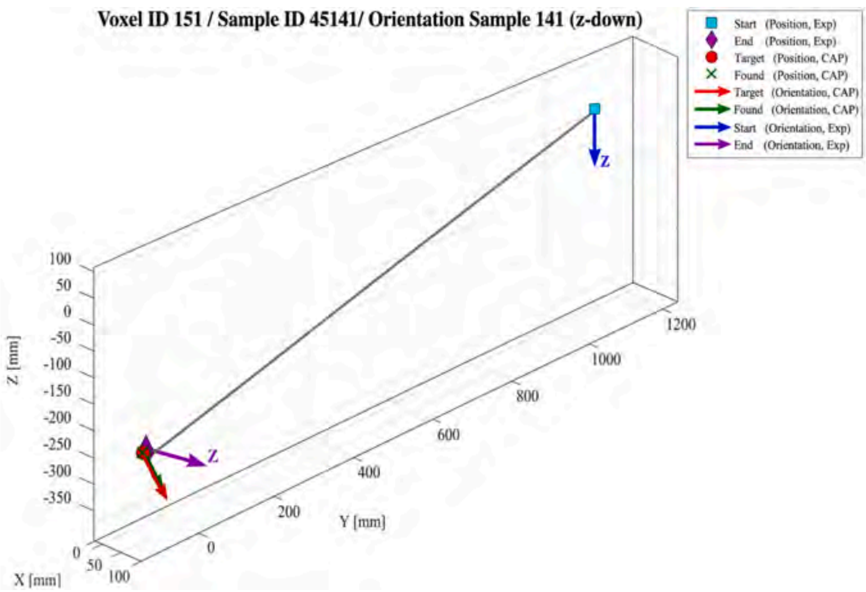

Fig. 20. Spatial comparison of the CAP target, solver-found CAP pose, and experimentally measured start/end poses in Voxel 151.

Table 4
Numerical summary of experimental error distributions for Voxel 151.

Statistic	Position vs. T_{target} [mm]	Position vs. T_{found} [mm]	Orientation vs. T_{target} [deg]	Orientation vs. T_{found} [deg]
Minimum	0.19	0.22	4.8	4.8
25th percentile (Q1)	0.57	0.54	28.5	28.5
Median	0.64	0.63	39.1	39.1
75th percentile (Q3)	0.86	1.19	73.5	73.5
Maximum	1.46	1.87	110.1	110.1

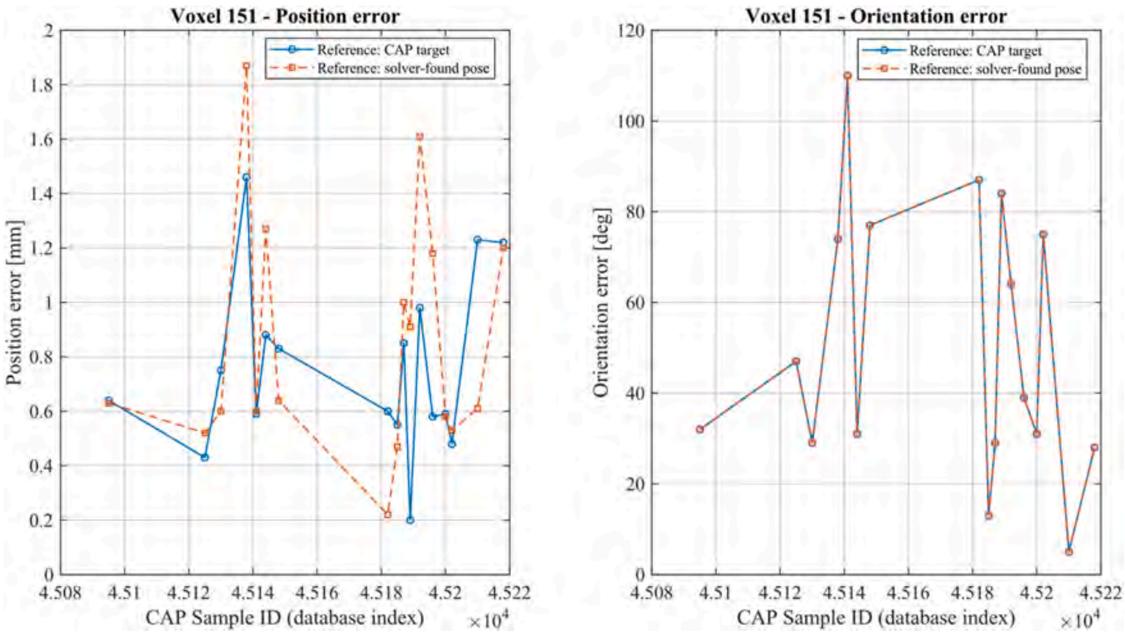

Fig. 21. Position and orientation errors of the experimentally executed CAP samples associated with Voxel 151, reported with respect to both the nominal CAP target and the solver-found pose.

For the position error with respect to T_{target}, the maximum value corresponds to the outlier shown in the box plot of Fig. 22.

The corresponding numerical error statistics are reported in Table 4, while Figs. 21 and 22 visualize the same experimental errors across the matched CAP entries and their distributions. In Fig. 21, the horizontal axis reports the CAP Sample ID, i.e., the database index of each matched voxel-orientation entry associated with Voxel 151. This identifier is used only to trace the corresponding stored CAP entry and should not be interpreted as time, distance, orientation angle, or a spatial coordinate. The results show that position errors remain in the millimetre range, whereas orientation errors can exceed 100° in the reported trials and should therefore be regarded as a limitation of the current open-loop execution.

These orientation errors should not be interpreted as residuals of the offline optimizer, since the retained CAP entries satisfy the database acceptance criteria before hardware execution. Rather, they arise after the selected configuration is converted into tendon/motor commands and executed on the physical prototype, where tendon slack, friction, pulley asymmetry, pretension variations, and inter-segment coupling can affect the measured orientation.

An important observation is that the measured final pose is often closer to the solver-found CAP pose than to the nominal target pose. This follows from the discretized CAP formulation: the robot is actuated from the stored solver-found configuration, not from the idealized voxel-level target. The remaining discrepancy is attributed to model-hardware mismatch and tendon-transmission effects during open-loop execution.

The same representative CAP-selected configuration was executed multiple times from the straight reference state to qualitatively assess trial-to-trial consistency. The measured end-effector poses remained in the neighbourhood of the stored solver-found configuration, supporting the physical executability of the selected CAP entry, although quantitative repeatability analysis over multiple voxels and more trials remains future work.

8. Discussion and conclusion

Offline CAP generation combined with Pareto-based selection provides a practical offline-online trade-off for TDCR configuration retrieval when the task is formulated at the voxel level rather than as exact pose-conditioned planning. By storing tendon-feasible PCC

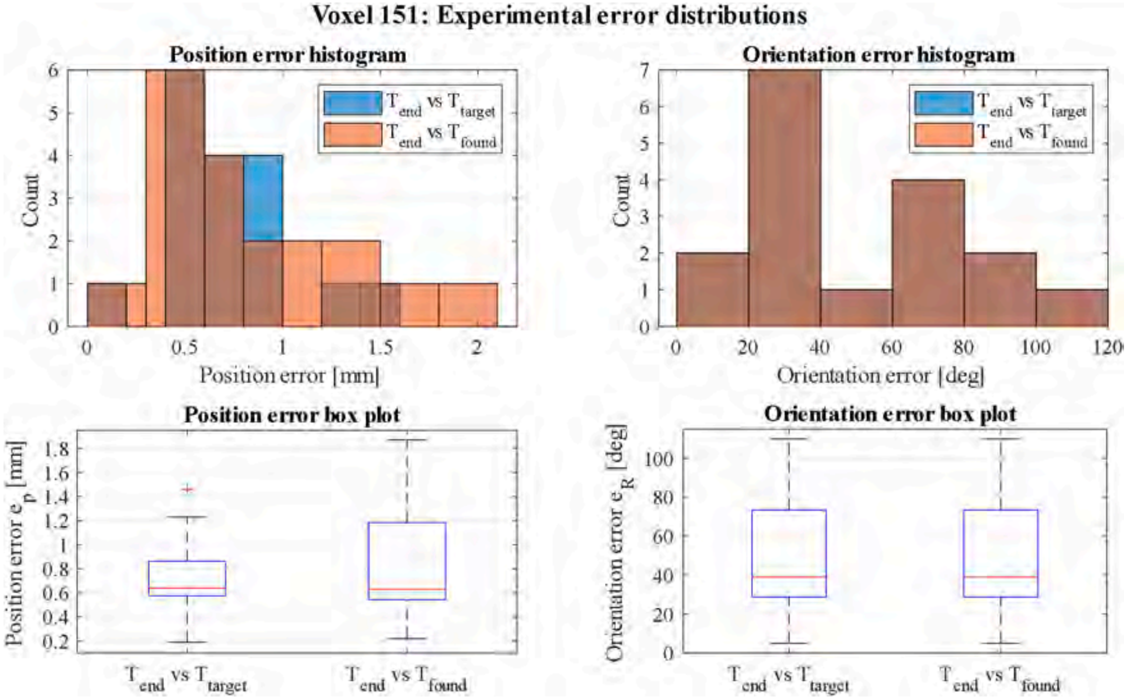

Fig. 22. Experimental error distributions for Voxel 151. Position and orientation errors are summarized with respect to the nominal CAP target and the stored solver-found pose.

configurations together with pose residuals and Jacobian-based dexterity descriptors, the proposed database enables millisecond-level retrieval of a dexterity-aware representative configuration from a precomputed local neighbourhood. The results show that reachability alone is insufficient for configuration selection, since feasible entries within the same voxel can exhibit different residual and manipulability properties. The Pareto/ideal-point selector therefore provides a deterministic compromise between stored pose quality and local dexterity. From a design perspective, these results indicate that reachability alone is insufficient when multiple feasible TDCR configurations exist within the same local neighbourhood; configuration-level dexterity should also be considered during configuration selection. The experimental validation complements the computational analysis by assessing the physical executability of a representative CAP-selected configuration, while the remaining orientation errors highlight the current model-to-hardware limitations of the open-loop implementation.

The present study should be interpreted as a proof-of-concept validation of the database-driven CAP framework on a laboratory-scale two-segment TDCR. The selector returns a neighbourhood representative rather than an exact-pose optimum; if a specific tool direction is required, the candidate set can first be filtered by orientation before Pareto ranking. The CAP database is generated using a kinematic disk-wise PCC model, which enables tractable offline sampling but does not explicitly capture gravity-dependent deformation, tendon friction, elasticity, hysteresis, slack redistribution, obstacle contact, or load-dependent effects. Therefore, the reported accuracy, obtained in the vertically suspended calibrated ROI, should not be assumed to transfer unchanged to horizontal mounting, larger loads, or contact-rich operation without recalibrating or regenerating the database under the corresponding conditions. Hardware validation showed millimetre-range position errors in the evaluated ROI, but the orientation errors remain significant and limit the current open-loop implementation for orientation-critical tasks. The fact that measured poses were often closer to the solver-found CAP entries than to the nominal voxel-level targets further confirms that the stored IK solutions are more representative of attainable robot states than the idealized voxel centres alone. Future work will extend the framework toward mechanics-aware and experimentally calibrated models, adaptive and parallel CAP construction, lightweight online correction, waypoint-based execution, horizontal-mounting and payload-variation tests, and broader multi-voxel hardware validation.

Funding

Mohammad Jabari acknowledges financial support from the PNRR-NGEU programme. This publication is part of a project funded by the Italian Ministry of University and Research (MUR) under Ministerial Decree No. 351/2022. This work was also financially supported by the Pprime Institute, a CNRS research unit established in partnership with the University of Poitiers and ISAE-ENSMA, and was carried out within the CoBRA research team.

CRediT authorship contribution statement

Mohammad Jabari: Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Carmen Visconte:** Writing – review & editing, Supervision, Methodology, Funding acquisition, Conceptualization. **Giuseppe Quaglia:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **Abdelbadia Chaker:** Writing – review & editing, Supervision, Software, Investigation, Conceptualization. **Med Amine Laribi:** Writing – review & editing, Validation, Supervision, Resources, Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] F. Iqbal, M. Esfandiari, G. Amirkhani, H. Hoshyarmanesh, S. Lama, M. Tavakoli, G.R. Sutherland, Continuum and soft robots in minimally invasive surgery: a systematic review, IEEE Access 13 (2025) 24053–24079, https://doi.org/10.1109/ACCESS.2025.3535677.
[2] T. da Veiga, J.H. Chandler, P. Lloyd, G. Pittiglio, N.J. Wilkinson, A.K. Hoshiar, R.A. Harris, P. Valdastri, Challenges of continuum robots in clinical context: a review, Prog. Biomed. Eng. 2 (3) (2020) 032003, https://doi.org/10.1088/2516-1091/ab9f41.
[3] P. Rao, Q. Peyron, S. Lilge, J. Burgner-Kahrs, How to model tendon-driven continuum robots and benchmark modelling performance, Front. Robot. AI 7 (2021) 630245, https://doi.org/10.3389/frobt.2020.630245.
[4] P.E. Dupont, N. Simaan, H. Choset, C. Rucker, Continuum robots for medical interventions, Proc. IEEE 110 (7) (2022) 847–870, https://doi.org/10.1109/JPROC.2022.3141338.
[5] M. Russo, S.M.H. Sadati, X. Dong, A. Mohammad, I.D. Walker, C. Bergeles, K. Xu, D.A. Axinte, Continuum robots: an overview, Adv. Intell. Syst 5 (5) (2023) 2200367, https://doi.org/10.1002/aisy.202200367.
[6] M. Jabari, C. Visconte, G. Quaglia, M.A. Laribi, Hybrid supervised learning and constrained optimization for task-space trajectory generation in tendon-driven continuum robots, ASME Open J. Eng. 5 (2026) 051003, https://doi.org/10.1115/1.4070630.
[7] L. Raimondi, M. Russo, X. Dong, D. Axinte, Understanding friction and superelasticity in tendon-driven continuum robots, Mechatronics 104 (2024) 103241, https://doi.org/10.1016/j.mechatronics.2024.103241.
[8] R. Thakker, et al., EELS: towards autonomous mobility in extreme terrain with a versatile snake robot with resilience to exteroception failures, in: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Detroit, MI, USA, 2023, pp. 9886–9893, https://doi.org/10.1109/IROS55552.2023.10341448.
[9] M. Jabari, C. Visconte, G. Quaglia, M.A. Laribi, Multi-objective optimization for dimensional synthesis of tendon placement and structural design for energy-efficient and feasible static workspace in continuum robots, Robot. Auton. Syst. 197 (2026) 105300, https://doi.org/10.1016/j.robot.2025.105300.
[10] B.Y. Cho, D.S. Esser, J. Thompson, B. Thach, R.J. Webster, A. Kuntz, Accounting for hysteresis in the forward kinematics of nonlinearly-routed tendon-driven continuum robots via a learned deep decoder network, IEEE Robot. Autom. Lett. 9 (11) (2024) 9263–9270, https://doi.org/10.1109/LRA.2024.3455797.
[11] M. Tummers, V. Lebastard, F. Boyer, J. Troccaz, B. Rosa, M.T. Chikhaoui, Cosserat rod modeling of continuum robots from Newtonian and Lagrangian perspectives, IEEE Trans. Robot. 39 (3) (2023) 2360–2378, https://doi.org/10.1109/TRO.2023.3238171.
[12] A. Doroudchi, Z. Qiao, W. Zhang, S. Berman, Implementation of a Cosserat rod-based configuration tracking controller on a multi-segment soft robotic arm, in: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Detroit, MI, USA, 2023, pp. 620–626, https://doi.org/10.1109/IROS55552.2023.10342160.
[13] Y. Wang, Z. Wu, L. Wang, B. Feng, K. Xu, Inverse kinematics and dexterous workspace formulation for 2-segment continuum robots with inextensible segments, IEEE Robot. Autom. Lett. 7 (1) (2022) 510–517, https://doi.org/10.1109/LRA.2021.3128689.
[14] A. Trabelsi, J. Sandoval, A. Mlika, S. Lahouar, S. Zeghloul, M.A. Laribi, Robot base placement and tool mounting optimization based on capability map for robot-assistant camera holder, Robotica 42 (8) (2024) 2489–2510, https://doi.org/10.1017/S0263574724000870.
[15] A. Trabelsi, J. Sandoval, A. Mlika, S. Lahouar, S. Zeghloul, J. Cau, M.A. Laribi, Optimal multi-robot placement based on capability map for medical applications, in: A. Müller, M. Brandstötter (Eds.), Advances in Service and Industrial Robotics, RAAD 2022, Mechanisms and Machine Science, Advances in Service and Industrial Robotics, RAAD 2022, Mechanisms and Machine Science, 120, Springer, Cham, 2022, pp. 333–342, https://doi.org/10.1007/978-3-031-04870-8_39.
[16] O. Porges, T. Stouraitis, C. Borst, M.A. Roa, Reachability and capability analysis for manipulation tasks, in: M. Armada, A. Sanfeliu, M. Ferre (Eds.), ROBOT2013: First Iberian Robotics Conference, Advances in Intelligent Systems and Computing, ROBOT2013: First Iberian Robotics Conference, Advances in Intelligent Systems and Computing, 253, Springer, Cham, 2014, pp. 703–718, https://doi.org/10.1007/978-3-319-03653-3_50.
[17] S. Cao, J. Yu, J. Pan, X. Pei, Design and moving capability analysis of a flexible continuum robot with rolling contact, J. Mech. Eng. 57 (19) (2021) 21–29, https://doi.org/10.3901/jme.2021.19.002.
[18] R. Danesh, F. Janabi-Sharifi, Backstepping control of tendon-driven continuum robots in large deflections using the Cosserat rod model, Mech. Mach. Theory 208 (2025) 105953, https://doi.org/10.1016/j.mechmachtheory.2025.105953.
[19] A. Bhalkikar, S. Lokesh, K.P. Ashwin, Kinematic models for cable-driven continuum robots with multiple segments and varying cable offsets, Mech. Mach. Theory 200 (2024) 105701, https://doi.org/10.1016/j.mechmachtheory.2024.105701.
[20] G. Lods, B. Rosa, B. Bayle, F. Nageotte, Exact derivative propagation method to compute the generalized compliance matrix for continuum robots: application to concentric tubes continuum robots, Mech. Mach. Theory 200 (2024) 105696, https://doi.org/10.1016/j.mechmachtheory.2024.105696.
[21] M. Jabari, C. Visconte, G. Quaglia, A. Chaker, M.A. Laribi, Capability mapping of a tendon-driven continuum robot using task-space sampling and PSO-based IK, in: A. Chaker, C. Villegas Ulloa, H. Saafi (Eds.), Emerging Fields in Mechanism and Machine Science, EFIMMS 2026, Mechanisms and Machine Science, Emerging Fields in Mechanism and Machine Science, EFIMMS 2026, Mechanisms and Machine Science, 205, Springer, Cham, 2026, pp. 207–214, https://doi.org/10.1007/978-3-032-22270-1_20.