Results in Engineering 30 (2026) 111434

Contents lists available at ScienceDirect

Results in Engineering

journal homepage: www.sciencedirect.com/journal/results-in-engineering

Research paper

A design framework for long-reach manipulators in confined spaces using task-based kinematic optimization and surface clustering

Emre Uzunoglu [a,*], Dohee Lee [b], Luca Raimondi [a], Kaiqiang Zhang [a], Hongtack Kim [b], Youngmin Park [b], Kwonhee Hong [b], Namil Her [b], Robert Skilton [a]

[a] United Kingdom Atomic Energy Authority, Abingdon, OX14 3DB, UK
[b] Korea Institute of Fusion Energy, Daejeon, 34133, Republic of Korea

ARTICLE INFO

Keywords:
Kinematic design optimization
Long-reach manipulators
Task space clustering
Confined-space inspection
Nuclear fusion maintenance

ABSTRACT

Industrial applications involving hazardous materials often require inspection within confined interiors, posing significant challenges due to restricted access and complex geometries. This paper presents a task-driven design synthesis framework for optimizing manipulators for full-surface inspection inside a nuclear fusion vacuum vessel.

To facilitate the kinematic optimization pipeline, we introduce a methodology using unsupervised surface clustering of CAD-derived geometries to extract a compact set of reachability targets, combined with a 2D plane projection method for fast collision pre-checks. Using this reduced task set, a multi-objective NSGA-II simultaneously optimizes kinematic type and dimensions to minimize total link length and static joint torques while satisfying reachability and collision constraints. These methods are formed as an integrated pipeline that systematically bridges the gap between abstract and evolving task-space definitions and optimized kinematic synthesis for confined environments.

A representative case study of a nuclear fusion vacuum vessel demonstrates that the framework efficiently identifies consistent designs with respect to defined surface inspection metrics. Compared to conventional sampling methods, the proposed approach achieves superior task-space coverage with significantly fewer reachability points. While conventional methods can also yield acceptable results when a sufficiently large number of task points is used, the proposed approach maintains consistent performance even with significantly fewer points, achieving mean runtime reductions of up to 52%. The results demonstrate a systematic and computationally efficient approach to the early-stage design of long-reach manipulators for access-limited confined environments. Although developed for fusion-vessel inspection, the framework can adapt to changes in geometry, constraints, and task requirements.

1. Introduction

Engineering applications for inspection and maintenance of interior surfaces in large, confined spaces with restricted access, such as industrial tanks and vessels, pose significant challenges, especially due to hazardous environments. Mechatronic systems, commonly robotic arms, have been widely utilized and developed in industrial practice, as highlighted in [1], with similar challenges related to confined and hazardous environments. Among various service applications in the manufacturing, transportation, and nuclear sectors (see advancements in [2–4] respectively), fusion reactor vessels are one of the most challenging confined environments due to their limited access ports, complex geometries, and collision-free reachability requirements, all while maintaining the generality of design challenges and solutions.

During the operation of a fusion power plant, plasma is confined within a toroidal vacuum vessel (VV) by strong magnetic fields [5]. Plasma-facing components are exposed to high levels of radiation and high temperatures, resulting in material activation and damage. A planned maintenance schedule is therefore necessary to maximize plant availability [6]. However, human entry is prohibited due to hazards, making robotic systems crucial for inspection and maintenance within the VV. Remote service manipulators must operate in high radiation and magnetic fields, rendering conventional architectures unsuitable. Furthermore, the VV geometry requires long-reach, slender manipulators

* Corresponding author.
E-mail address: emre.uzunoglu@ukaea.uk (E. Uzunoglu).

https://doi.org/10.1016/j.rineng.2026.111434
Received 2 December 2025; Received in revised form 24 May 2026; Accepted 7 June 2026
Available online 9 June 2026

that face challenges including structural compliance, large joint torque, and reduced stiffness. Representative systems such as the Super Dragon [7] and the Articulated Inspection Arm (AIA) [8] exemplify the need for bespoke mechatronics tailored to these constraints.

Despite extensive research on task-based kinematic optimization, including approaches that integrate dimensional synthesis, multi-objective frameworks, and advanced computational heuristics, current literature has not fully addressed the unique requirements of long-reach, slender manipulators operating in the constrained environments of fusion reactors. In [9], a task-oriented dimensional synthesis method is proposed for laser operations in aero engines, demonstrated that optimizing both geometry and joint variables for a discrete set of task points can significantly improve performance in narrow spaces compared to conventional architectures. This paradigm of tailoring modular components for restricted volumes is further reflected in the development of steel arch looping manipulators for tunnel boring [10] and the structural synthesis of parallel robots for cryogenic handling [11], where multi-objective particle swarm optimization (PSO) is leveraged to balance kinematic efficiency against actuator force requirements.

In robotic design, diverse strategies have been explored to enhance adaptability and workspace coverage through modular architectures [12] and the implementation of localized performance indices for workspace optimization [13]. These frameworks have evolved to include unified structural dimensional synthesis using heuristic-guided searches [14,15], as well as black-box and deep reinforcement learning strategies to manage the high-dimensional design spaces inherent in modular systems [16,17]. Another critical aspect of these optimizations is the design of optimization goals; for instance, Chen et al. [18] utilized simulated annealing to evaluate simultaneous kinematic and dynamic performance, while [19] demonstrated how the selection of different torque indices yields distinct trade-offs between joint efficiency and workspace reachability. A multi-objective design optimization approach is proposed in [20] by introducing virtual joints. They transformed a non-redundant manipulator into a kinematically redundant one, enabling redundancy resolution methods to optimize structural parameters.

In the context of robot design optimization, the kinematic design of articulated arms for inspection tasks in power plants has not been extensively studied using task-based approaches. Specialized applications remain focused on decommissioning platforms, such as dual seven-DOF arms [21] and cable-driven hyper-redundant manipulators designed to navigate harsh industrial environments [22]. Recent feasibility assessments for in-vessel inspection systems (IVIS) have further explored the suitability of sensor technologies and structural integrity under vacuum and extreme thermal gradients [23].

From this analysis, it is clear that kinematic optimization has been extensively investigated. However, most existing studies assume a predefined set of task poses and do not address how such task sets should be derived when operating in confined environments with complex surface geometries. In nuclear fusion vacuum vessels and surroundings, inspection feasibility depends not only on reachability but also on surface orientation and accessibility relative to constrained entry ports. Therefore, deriving a semantically meaningful and physically representative subset of task points from dense CAD-derived point clouds becomes a critical preprocessing step.

Deriving geometrical information from the vacuum vessel point cloud requires processing the point cloud data. However, point clouds for intricate 3D geometries often contain a prohibitive number of points, rendering high-level analysis computationally intensive. Consequently, effective downsampling methods are essential to maintain tractability. In point cloud processing, reduction of dense surface data is commonly performed using heuristic downsampling strategies such as Uniform Downsampling (UD) or Farthest Point Sampling (FPS) [24,25]. Uniform sampling enforces spatial regularity but may overlook high-curvature regions or orientation-specific constraints. FPS improves global spatial dispersion by iteratively selecting maximally distant points; however, it remains purely geometric and does not incorporate task-specific constraints such as normal alignment or reference points [26].

Recent learning-based sampling approaches, including S-NET [27], SampleNet [28], and AS-PD [29], introduce task-aware point selection mechanisms. Nevertheless, these methods are primarily developed for shape classification, segmentation, or registration tasks and do not explicitly incorporate robotic kinematic feasibility constraints. In particular, they do not account for requirements such as sensor orientation alignment or accessibility constraints in confined environments, and rely on supervised training over large annotated 3D shape datasets.

However, a notable gap remains for a generalized, task-driven optimization pipeline that specifically exploits high degrees of freedom to resolve the inherent trade-off between reachability and collision avoidance within the complex, non-linear geometry of operation spaces like a fusion reactor's. This study addresses this gap by proposing an adaptable synthesis framework designed to optimize the kinematic configuration of long-reach manipulators for the next generation of fusion maintenance. To address this research gap, a new framework is proposed to extract dense task points directly from the vessel geometry. A surface-clustering procedure is applied to condense these into a reduced set of representative poses, bridging task abstraction and manipulator design while reducing computational cost. Based on this reduced task set, a kinematic optimization process determines the manipulator topology and geometry [30], with the objectives of minimizing the number of degrees of freedom, total manipulator length, and static joint torques, while ensuring collision-free operation. The framework is inherently adaptable and can accommodate variations in vessel geometry, operational constraints, or evolving inspection requirements.

The structure of this study is outlined as follows. Section 2 introduces the use case, workspace details, and kinematic constraints. Section 3 presents the details of the surface clustering, point selection, and optimization algorithm. The results are presented in Section 4, in which the best solutions are identified via the Pareto front approach and compared against conventional sampling baselines. Finally, Section 5 presents conclusions and future work.

2. Problem statement and design constraints

The problem can be summarised as the design of an articulated mechatronic system capable of operating within industrial confinements through constrained access ports. Its kinematic chain must enable the end effector to reach designated operational positions distributed across the interior surface of the confinement. This study adopts the representative environment of the VV of a nuclear fusion reactor as a case study to illustrate the design problem. Specifically, the design task is defined as identifying an optimized kinematic chain for an articulated robotic arm performing visual inspection within the VV. The robot is intended to inspect the inner surface of the vessel for potential defects after reactor shutdown. Parameters related to visual inspection, such as viewing distance and angle, are treated as fixed specifications rather than optimization variables, allowing flexibility across different use cases and sensor technologies. This assumption is generally valid, because such parameters are typically dictated by the service instrument's design requirements.

Fig. 1 shows the toroidal geometry of the simplified VV considered. The minor radius, r_1, of the VV is set to 4.60 m , the centerline radius, r_2, is 6.80 m, and the major radius, r_3, is 9 m. The total maximum height of the cross section, h, is 8.47 m.

The deployment port, through which the manipulator is inserted inside the VV, is highlighted with a red circle in Fig. 1 and has a radius of 300 mm. When not in use, the robot is stored in a cask outside the deployment port and can access the reactors by actuating a first prismatic joint. To prevent collision with the environment, the revolute joints can be actuated only when 0.3 m inside the VV. If any joint remains within the cask, its movement is restricted, and the robot will attempt to reach a position with only the DoF inside the VV.

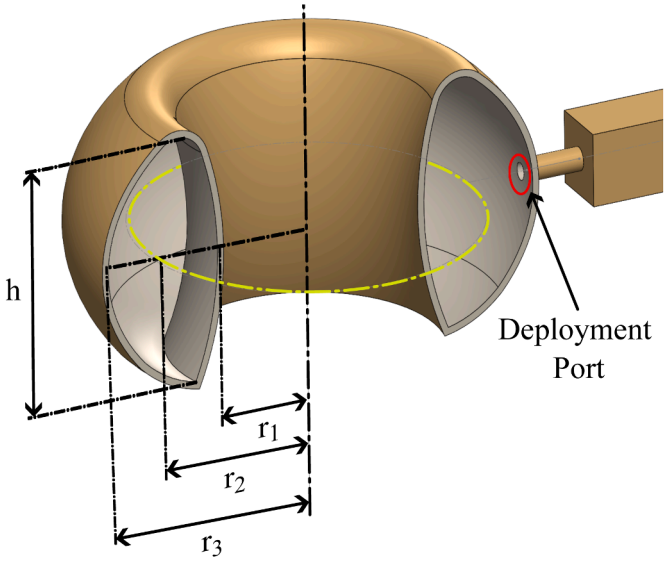

Fig. 1. Simplified geometry of the VV (h: height, r_1 minor radius, r_2 center line, r_3 major radius).

The VV is designed with six dedicated equidistant access ports for diagnostics, a configuration consistent with fusion reactor port architectures as discussed in [31], meaning each port covers 60° of the inner wall's surface, as shown in the top view of Fig. 2. Fig. 2 shows reachability test points (green markers) and their associated surface normals (red arrows) on the VV first wall, obtained by intersecting the geometry with a section plane. A multi-DOF robot carrying out surface visual inspection is displayed as a representative example. The target poses are derived from the surface normal and the Cartesian coordinates of the clustered points. This formulation constrains a 5-DoF task requirement: the target position and the alignment of the primary tool axis (e.g., the longitudinal x-axis), the rotation about that axis is left unconstrained for the solution for this specific case. The first link (black) acts as a prismatic joint that positions the robot within the vessel. The blue links denote the link dimensions to be optimized, while the purple joints represent the revolute joints treated as kinematic optimization variables.

The problem is to find optimized link lengths and robot joint types to align the tip of the robot with the selected reachability test points and, consequently, design a robot capable of inspecting the surface of the vessel. However, such an optimization would be very costly if all the points on the surface are selected as reachability test points in the optimization process. The challenge of this study is to identify a limited number of reachability test points that ensure sufficient coverage without creating a prohibitively expensive algorithm to optimize the robot's kinematics.

For application-specific optimization of the link length and joint type synthesis, the following constraints are considered:

First Prismatic Joint. The first prismatic joint allows the manipulator to be deployed from the storage cask into the VV. The joint can extend from a position 2.7 m inside the cask, i.e. outside the VV, up to the center line of the VV torus (indicated by the yellow dashed line in Fig. 1), reaching 2.2 m inside the vessel from the entrance port, for a total maximum travel of 4.9 m. Such first prismatic (translational) joint mechanisms, sometimes referred to as deployment systems, are commonly adopted to provide sufficient insertion depth of manipulators operating within tokamaks; the CASK-based deployment of the CFETR heavy-duty robot [32] and the AIA system for ITER [33] can be presented as examples.

Degrees of Freedom and Joint Types. To ensure reachability in both position and orientation, a minimum of 6 DoF is considered with additional redundant DoFs potentially added during the optimization of the kinematic enumeration (n_{DoF}) to enable avoidance of self-collision and collision with the environment. To simplify the actuation, the robot is designed to have only revolute joints configured with yaw and pitch, following the first prismatic joint. Throughout the paper, the configurations of the revolute joints will be identified using binary codes; yaw joints will be labeled *1* and pitch joints *0*. For example, a configuration code of *1010* for a 5-DoF robot indicates a sequence of yaw, pitch, yaw, pitch joints following a prismatic joint.

Link Lengths and Joint Limits. The link length l_i $(i = 1, 2, ., n_{DoF})$ is constrained in the range $0.3\ \text{m} \leq l_i \leq 1.5\ \text{m}$ for ease of deployment and maneuverability inside the VV. Likewise, the rotation range for the revolute joints θ_i is constrained in the range $-90° \leq \theta_i \leq 90°$. In addition, to prevent collision with the entrance port, the revolute joints can initiate motion only after being 0.3 m inside the vessel.

Static Torque Calculation. To calculate the required joint torques under static conditions for a given configuration of the manipulator, a payload of 10 kg at the tip of the last link is considered. This value, as shown in other studies [7,8], is an estimate in excess of the majority of the end-effector tools, e.g., cameras, used to inspect and maintain fusion reactors. Each intermediate link is modelled as a uniform body with its mass concentrated at the link centre, whose mass scales linearly with its length, yielding approximately 5 kg/m, consistent with the 6 kg per 1.2 m section reported by [34] for a comparable system. During optimization, link lengths are updated dynamically, and the corresponding link masses are recalculated accordingly. The static torque at each joint is determined by the gravitational load of all downstream links and the end-effector payload, together with their respective moment arms relative to that joint. This creates an inherent trade-off within the optimization. The current mass model serves as a preliminary estimate for the kinematic synthesis stage; refined structural parameters will be incorporated once the kinematic topology is finalised and detailed structural design is undertaken.

3. Methodology

The initial part of the methodology presents a new pipeline that reduces the number of reachability test points in kinematic optimization, while preserving the geometric and directional variety of the surface to be examined. This approach aims to accelerate optimization without sacrificing coverage quality, especially in restricted geometries such as

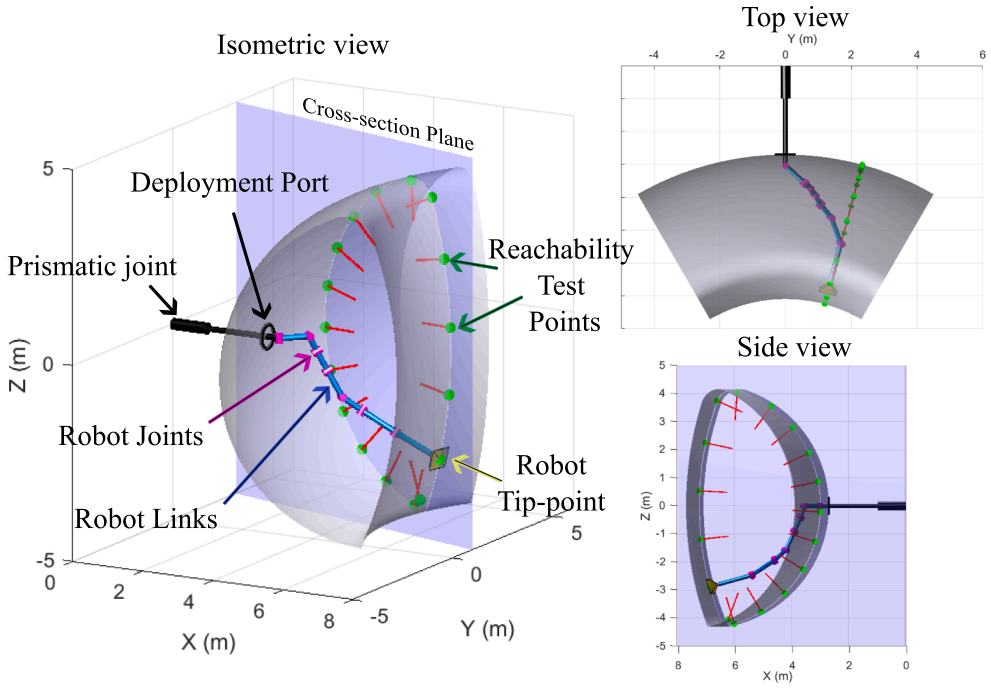

Fig. 2. A cross-section of the 3D geometry defined by a blue plane, showing reachability test points (green markers) with normals (red arrows) on the surface. The multi-DOF robot within the VV features a black prismatic joint for positioning, blue links for optimized dimensions, and purple revolute joints for kinematic parameter optimization, illustrated in 3D top and side views. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the interior of a nuclear fusion vessel. Then, a kinematic optimisation with two objectives, the total link length and the sum of joint torque requirements in static conditions, is introduced to ensure the robot can reach all the target points in position and orientation. For this, the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [35] is employed, which is a widely used multi-objective optimization algorithm that utilizes a crowding distance mechanism to maintain a diverse set of Pareto-optimal solutions. This method efficiently balances exploration by preserving solution diversity while converging toward the Pareto front.

3.1. Surface clustering and reachability point selection

Evaluating reachability against every discretized surface point is computationally prohibitive for kinematic optimization in confined geometries. The approach consists of three integrated stages: (i) clustering the surface using a hybrid metric combining position, surface orientation, and distance-to-port; (ii) selecting a globally aware but compact set of representative reachability points per cluster via a diversity-optimized scoring; and (iii) building the PCA-based 2D boundary hulls for fast collision screening. This reduces the number of test points while maintaining coverage of geometric and directional variation relevant to optimization.

The procedure begins with the extraction of points from the mesh file generated from the CAD model. Each extracted point is represented by the following data:

$$\{(\mathbf{p}_i, \mathbf{n}_i, d_i)\}_{i=1}^{N} \tag{1}$$

where $\mathbf{p}_i \in \mathbb{R}^3$ are Cartesian coordinates, $\mathbf{n}_i \in \mathbb{R}^3$ are unit normals, and $d_i \in \mathbb{R}$ is the signed distance to the manipulator entrance port. For a case-specific demonstration, the export mesh file and the surface normals directed towards the operation space of the vessel are extracted from the model represented in Fig. 1.

3.1.1. Hybrid k-means surface clustering

Clustering is a fundamental unsupervised learning technique used to partition data into groups based on feature similarity. A popular approach is k-means clustering, which aims to minimize the within-cluster sum of squared distances (variance) to cluster centroids. This method is widely employed because of its relative computational efficiency and conceptual simplicity, especially in large-scale applications like segmentation and vector quantization [36]. Standard k-means uses a unimodal distance metric (typically Euclidean), which limits its applicability when data exhibit multiple features that distinguish them or when similarity cannot be captured by spatial distances alone. In such cases, a hybrid distance metric that combines multiple features can enable more semantically meaningful clusters.

In this methodology, the hybrid distance $\Delta_{\text{hyb}}(i,j)$ between two points i and j is defined as the sum of three normalized components:

$$\Delta_{\text{hyb}}(i,j) = \bar{\Delta}_{\text{pos}}(i,j) + \bar{\Delta}_{\text{ang}}(i,j) + \bar{\Delta}_{\text{dist}}(i,j),$$

where $\bar{\Delta}_{\text{pos}}$ is the Euclidean position distance that groups spatially adjacent regions (curvature/extension), $\bar{\Delta}_{\text{ang}}$ accounts for the normal vector angular difference which groups similarly oriented patches (tooling/view relevance), and $\bar{\Delta}_{\text{dist}}$ refers to differences in signed distance to a reference point which emphasizes accessibility from the entry port with a hybrid distance that balances spatial proximity, local orientation, and accessibility. These terms are defined as follows:

$$\bar{\Delta}_{\text{pos}}(i,j) = \frac{\|\mathbf{p}_i - \mathbf{p}_j\|_2}{\max(\Delta_{\text{pos}})}, \tag{2}$$

$$\bar{\Delta}_{\text{ang}}(i,j) = \frac{\arccos(\hat{\mathbf{n}}_i^{\top}\hat{\mathbf{n}}_j)}{\max(\Delta_{\text{ang}})}, \tag{3}$$

$$\bar{\Delta}_{\text{dist}}(i,j) = \frac{|d_i - d_j|}{\max(\Delta_{\text{dist}})}, \tag{4}$$

where $\max(\Delta_{\text{pos}})$, $\max(\Delta_{\text{ang}})$, and $\max(\Delta_{\text{dist}})$ represent the maximum pairwise distances for position, angle, and entry-port distance found in the specific surface point cloud.

Clustering is initialised via $k-means++$ [37] and updated iteratively using the hybrid distance metric.During the iterative update process, each point is assigned to the cluster minimizing Δ_{hyb}, and the cen-

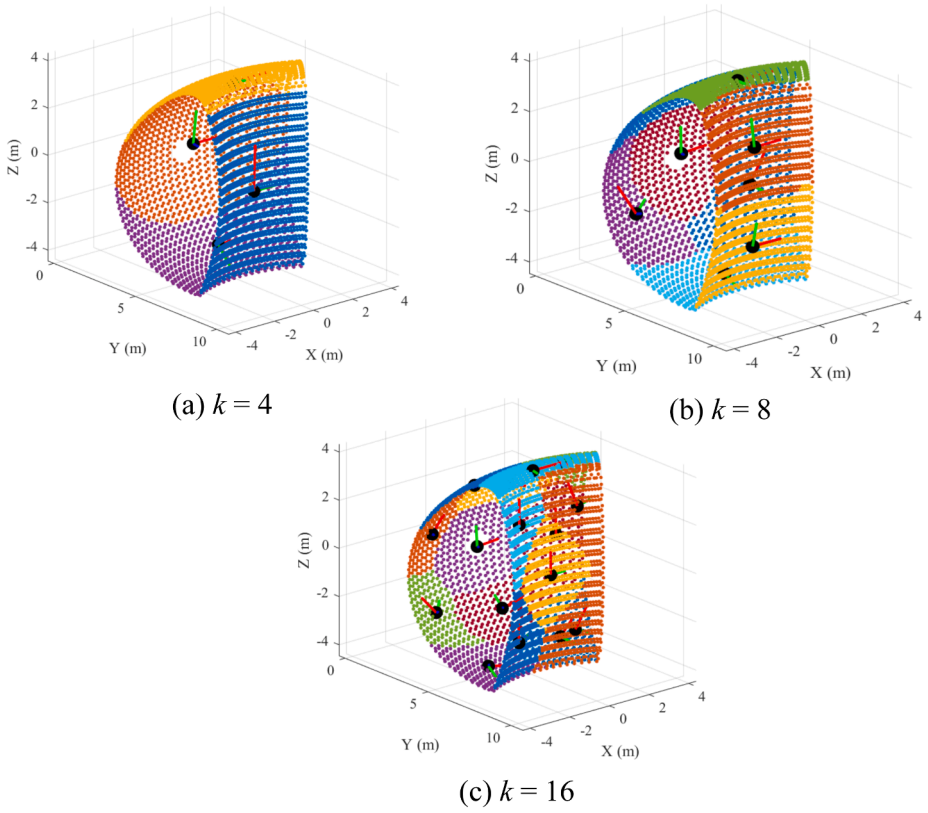

Fig. 3. Hybrid clustering results for (a) $k = 4$, (b) $k = 8$, and (c) $k = 16$. Colors denote clusters. Local PCA axes per cluster are overlaid to indicate dominant tangent directions used later for boundary-hull construction.

troid for each cluster C_m is updated based on the mean of the combined feature vector $[\mathbf{p}_m, \mathbf{n}_m, d_m]$. This approach leverages rich geometric information to align clusters more closely with the kinematic task context.

Fig. 3 shows the outcome of the proposed hybrid k-means surface clustering for three different numbers of clusters. Each color represents a distinct cluster derived from the CAD-based task surface. For each cluster, a local Principal Component Analysis (PCA) is performed on the point coordinates to identify its dominant tangent directions. Specifically, PCA decomposes the local point distribution into orthogonal components: the first and second principal components, span the best-fit tangent plane, while the third component corresponds to the surface normal direction ($\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$). The local PCA axes are overlaid as line segments in the plots to visualise these dominant directions, which are later used to construct the boundary hulls for collision and reachability checks.

3.1.2. Globally aware iterative selection from each cluster

To ensure that the kinematic optimization is driven by a workspace representation that is both locally accurate and globally comprehensive, a globally aware selection strategy is applied to the clustered task-surface points. While hybrid k-means surface clustering ensures that points within each cluster are spatially coherent, purely local sampling can overlook coverage gaps in the global workspace. To address this, a selection process that evaluates candidate points not only by their proximity to the cluster centroid but also by their global diversity in position is used.

After the surface clustering is completed, a fixed number of representative points is selected from each cluster for subsequent reachability test point selection. For each cluster, the aim is to have one center point (medoid) and a limited selection of test points, ensuring radial extent, global coverage, and directional spread. To ensure cluster diversity, a sequential adjacency-aware selection algorithm is proposed. This approach maintains local diversity by favouring points with distinct radial and angular properties with respect to the cluster center, while introducing a lightweight awareness term that penalises redundancy with recently selected test points in neighbouring clusters.

Let $C_m = \{\mathbf{p}_{m,i}\}_{i=1}^{N_m}$ denote the set of N_m points in cluster m, and $\bar{\mathbf{p}}_m$ its centroid:

$$\bar{\mathbf{p}}_m = \frac{1}{N_m} \sum_{i=1}^{N_m} \mathbf{p}_{m,i} \tag{5}$$

For each point $\mathbf{p}_{m,i}$ its radial distance is defined from the centroid:

$$r_{m,i} = \|\mathbf{p}_{m,i} - \bar{\mathbf{p}}_m\|_2 \tag{6}$$

and its unit direction vector:

$$\hat{\mathbf{u}}_{m,i} = \frac{\mathbf{p}_{m,i} - \bar{\mathbf{p}}_m}{r_{m,i}}, \quad r_{m,i} > 0 \tag{7}$$

If unit surface normals $\hat{\mathbf{n}}_{m,i}$ are available from the mesh geometry, they are normalized and associated with each point.

The first representative point for each cluster is the medoid, selected by:

$$\arg\min_i \|\mathbf{p}_{m,i} - \bar{\mathbf{p}}_m\|_2 \tag{8}$$

This ensures that the local cluster center is always represented in the reachability test points.

For the choice of the additional points within each cluster, a scoring function is created:

$$S_{m,i} = \lambda_r R_{m,i} + \lambda_a A_{m,i} - w_p P_{\text{prox}}(m,i) \tag{9}$$

where λ_r and λ_a are the radial and angular weight terms, and w_p adjacency weight penalty.

$$R_{m,i} = \frac{r_{m,i}}{r_m^{90}}, \quad r_m^{90} = \text{percentile}_{90}(\{r_{m,i}\}) \tag{10}$$

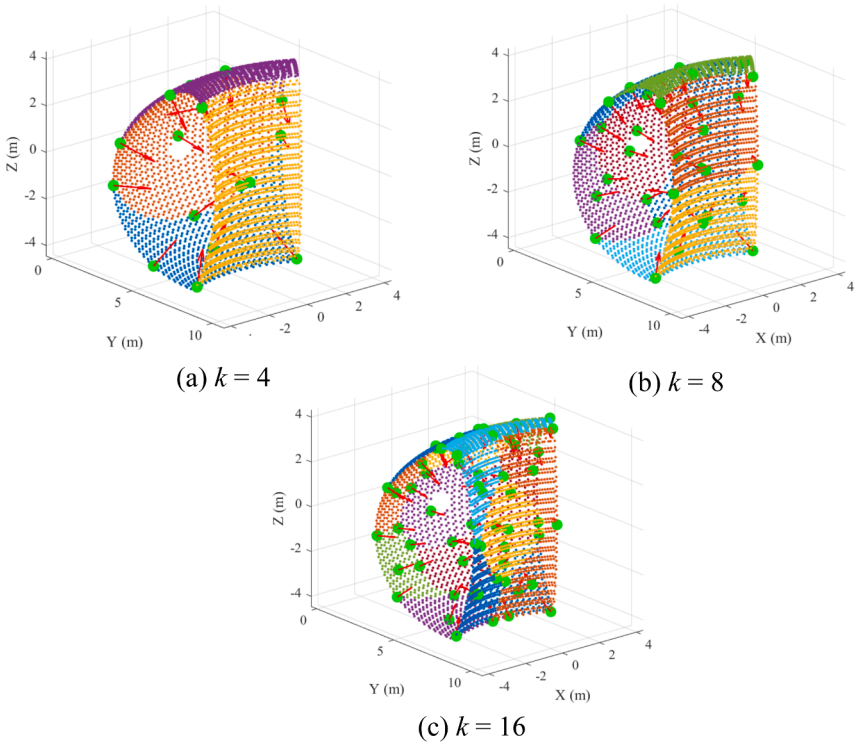

Fig. 4. Representative reachability points after globally aware selection. For each cluster: one center (near $\bar{\mathbf{p}}_m$) and 3 points selected with the algorithm for boundary representation. The set is radially extended, globally spread, and angularly diverse, providing a compact but informative test set for optimization.

Here, r_m^{90} denotes the 90th percentile of radial distances within a cluster, which reflects the typical outer extent of each clusters.

For a candidate with direction $\hat{\mathbf{u}}_{m,i}$ and already-selected local directions $\{\hat{\mathbf{u}}_{m,j}\}$:

$$A_{m,i} = 1 - \max_j \left|\hat{\mathbf{u}}_{m,i}^{\top}\hat{\mathbf{u}}_{m,j}\right|, \tag{11}$$

This term encourages wide angular separation within the same cluster. In this iterative adjacency penalty addition, only previously processed clusters within an adjacency window L are considered. Let P_{prox} denote the set of representative points selected in the last L clusters, with positions $\mathbf{q}_\ell$ denoting the set of representative points already selected across the clusters up to iteration k, ensuring global awareness during selection.

The penalties are computed as:

$$P_{\text{prox}}(m,i) = \max_k \exp\left[-\left(\frac{\|\mathbf{p}_{m,i} - \mathbf{q}_k\|_2}{\sigma_d\, r_m^{90}}\right)^2\right]. \tag{12}$$

where σ_d can be used to control the decay with distance and angular difference.

Fig. 4 illustrates the reachability points obtained using the procedure proposed for $k = 4$, $k = 8$, and $k = 16$ clusters. In each case, the algorithm selects one central point (located near the cluster centroid $\bar{\mathbf{p}}_m$) and selects 3 points with iterative selection from each cluster that are radially extended and angularly diverse. As k increases to 8 and 16, the point distribution becomes denser and more locally refined, with boundary points capturing finer geometric detail while maintaining global spread through adjacency-aware penalisation.

3.1.3. Collision plane generation and collision checking in the VV

During the next stage, where the kinematic optimization algorithm is run, collision checking is necessary to ensure the robot configuration can reach the chosen points safely. To perform this efficiently, the clusters previously created are used, and the 3D points are projected onto a 2D planar representation, which enables a fast, approximate check for potential collisions.

For each C_m, PCA on any $\mathbf{p}_i$ yields an orthonormal basis $(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$ where $(\mathbf{v}_1, \mathbf{v}_2)$ span the dominant tangent plane and $\mathbf{v}_3$ approximates the local normal. The projected coordinates are:

$$u_i = (\mathbf{p}_i - \bar{\mathbf{p}}_m)^{\top}\mathbf{v}_1, \quad v_i = (\mathbf{p}_i - \bar{\mathbf{p}}_m)^{\top}\mathbf{v}_2, \quad w_i = (\mathbf{p}_i - \bar{\mathbf{p}}_m)^{\top}\mathbf{v}_3. \tag{13}$$

A 2D boundary hull $\mathcal{H}_m$ is computed from (u_i, v_i) to form a conservative guard band $\mathcal{H}_m^+$. A joint position $\mathbf{x} \in \mathbb{R}^3$ is mapped to:

$$u_x = (\mathbf{x} - \bar{\mathbf{p}}_m)^{\top}\mathbf{v}_1, \quad v_x = (\mathbf{x} - \bar{\mathbf{p}}_m)^{\top}\mathbf{v}_2, \quad w_x = (\mathbf{x} - \bar{\mathbf{p}}_m)^{\top}\mathbf{v}_3, \tag{14}$$

In Fig. 5, a representative cluster taken from the processed VV with a hybrid k-means surface clustering algorithm is shown alongside the plane and the 2D boundary projected onto the plane, which is used for reducing the dimensionality of 3D scattered points.

Fig. 6 shows the resulting collision surfaces generated for the VV geometry partitioned into 16 clusters. Each cluster and its boundaries are represented with red lines, which show the result of the algorithm to generate the reduced surface representation of each cluster for the VV to be used for collision detection.

An additional rule-based gate is used to check if any joint of the robot inside the vessel is in contact or close to any 2D collision planes. For each joint position in Euclidean space, represented with $\mathbf{q} = [q_x, q_y, q_z]$, the collision of each joint with the surface is checked with Algorithm 1.

3.2. Optimization method

Two objectives, f_1 and f_2, are defined as the cost functions for each candidate solution $\boldsymbol{\xi} \in P$ in the population:

$$f_1 = \sum_{i=1}^{n_{DoF}} l_i, \tag{15}$$

$$f_2 = \frac{1}{m}\sum_{j=1}^{m} \|\boldsymbol{\tau}(j)\| \tag{16}$$

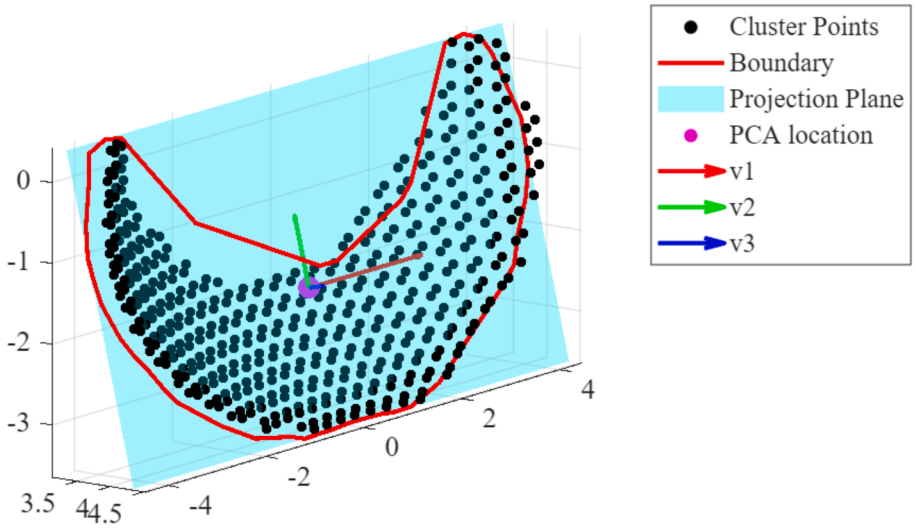

Fig. 5. PCA-plane and boundary generated for a representative cluster. Left: projection of cluster points onto $(\mathbf{v}_1, \mathbf{v}_2)$. Right: boundary $\mathcal{H}_m$ and outward expansion $\mathcal{H}_m^+$ used for fast collision screening of joint positions.

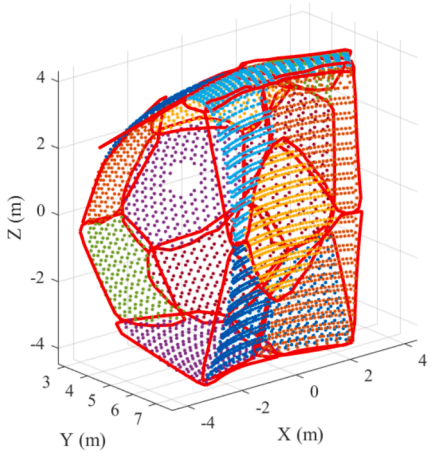

Fig. 6. 2D boundaries generated for the k = 16 cluster case for preliminary fast collision screening.

Algorithm 1 PCA-plane hull collision check for a joint position.

Require: Joint position $\mathbf{q} \in \mathbb{R}^3$; cluster j data: centroid $\boldsymbol{\mu}'_j$, PCA basis $(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$, boundary hull $\mathbf{H}'_j$ of each cluster
Ensure: COLLISION $\in$ {true, false}
Plane distance: $d_\perp \leftarrow \mathbf{v}_3^\top(\mathbf{q} - \boldsymbol{\mu}'_j)$
Projection to plane: $\mathbf{q}_\Pi \leftarrow \mathbf{q} - d_\perp \mathbf{v}_3$
Local 2D coords:
  $u \leftarrow (\mathbf{q}_\Pi - \boldsymbol{\mu}'_j)^\top \mathbf{v}_1, \quad v \leftarrow (\mathbf{q}_\Pi - \boldsymbol{\mu}'_j)^\top \mathbf{v}_2$
Hull inclusion: $h \leftarrow \text{INPOLYGON}\big((u, v), \mathbf{H}'_j\big)$
Decision:
if $|d_\perp| < d_0 \ \wedge \ h = \text{true}$ **then**
  return TRUE {collision}
else
  return FALSE
end if

where f_1 represents the total link length, f_2 the average norm of the static joint torque vector evaluated over all configurations $j = 1, \ldots, m$, and m the number of reachability test points.

The two objectives are evaluated together as the objective vector:

$$(f_1, f_2) = FF(\mathcal{T}, \boldsymbol{\xi}, \boldsymbol{\nu}), \tag{17}$$

where $\mathcal{T} = \{\boldsymbol{T}_1, \boldsymbol{T}_2, \ldots, \boldsymbol{T}_m\}$ is the set of desired tasks representing the required end-effector poses for the m reachability task points, and $\boldsymbol{\nu}$ denotes the design variables, including link length lower and upper bounds that define the constraints.

The values of candidate solutions $\boldsymbol{\xi}$ generated by the genetic algorithm (GA) represent the kinematic tree and are characterized by $n_{DoF} + 1$ parameters:

$$\boldsymbol{\xi} = \left[\xi_1 \ldots \xi_{n_{DoF}+1}\right] \tag{18}$$

For each candidate solution, the first n_{DoF} parameters are subjected to the constraint defined in Eq. (19), with lb representing the lower bound and ub representing the upper bound for the link lengths, as detailed in Section 2:

$$lb < \xi_{(1,\ldots,n_{DoF})} < ub \tag{19}$$

The last parameter, $\xi_{n_{DoF}+1}$, uses binary coding. The decimal values ranging from 0 to $2^{n_{DoF}} - 1$ each represent a unique configuration of the kinematic tree, incorporating variations in yaw and pitch joints, as discussed in Section 2. This approach enables the representation of all joint types within the kinematic tree through a single variable.

3.3. Fitness function algorithm

Algorithm 2 provides a schematic overview of the optimization workflow to calculate the cost of both objectives for each candidate solution generated by the GA. This workflow focuses on finding the solution for each reachability point and computing the fitness objectives f_1 and f_2. An inverse kinematics solver (IKS) is applied to each generated kinematic structure to test the reachability points and to determine its feasibility for optimization. The optimization of kinematic enumeration is represented for $n_{DoF} + 1$ variables in vector form by (18). Given a desired set of task poses $\mathbf{T} \in \mathcal{T}$ for an end-effector, the IKS computes the joint configuration vectors θ_{sol} that realize the desired end-effector poses while ensuring compliance with design constraints.

The fitness evaluation function FF assesses the performance of a kinematic structure by determining its ability to reach a set of target poses $\mathcal{T}$ while avoiding collisions within a predefined workspace. The algorithm takes the kinematic parameter vector $\boldsymbol{\xi}$, the set of target pose $\mathbf{T}_j$, additional design variables $\mathbf{v}$ related to the kinematic tree configuration, and the geometric model **Geo** representing the environment as

input. The process begins by enumerating an n-degree-of-freedom (DoF) kinematic tree and setting an initial joint configuration vector $\boldsymbol{\theta}_{\text{init}}$ with a tolerance threshold to assess accuracy. The algorithm then iterates through each reachability test point $\mathbf{T}_j$, solving the inverse kinematics (IK). Once a solution $\boldsymbol{\theta}_{\text{sol},j}$ is obtained, the algorithm evaluates reachability via an error function e_{err}, which measures the deviation between the achieved end-effector pose and the target $\mathbf{T}_j$. In this study, e_{err} is specifically configured to monitor positional accuracy and axial alignment with the surface normal, while intentionally omitting the axial rotation (roll) constraint to match the 5-DOF nature of the inspection task. If e_{err} falls within the predefined tolerance, the collision detection step CollCheck is performed to ensure the joint configuration does not violate environmental constraints defined by **Geo**. In cases where no valid solution is found, a randomization step explores alternative initial joint configurations by sampling multiple starting points in the configuration space, reducing the risk of convergence to local minima inherent in non-convex IK problems. When a valid, collision-free solution is identified for all points in $\mathcal{T}$, the fitness objectives f_1 and f_2 are calculated, which are utilized to find the Pareto-optimal solutions.

Algorithm 2 Evaluation of the fitness function.

Require: $\boldsymbol{\xi}$, $\mathcal{T}$, $\boldsymbol{\nu}$, **Geo**
Ensure: f_1, f_2
Initialization:
Generate n_{DoF} kinematic tree using $\boldsymbol{\xi}, \boldsymbol{\nu}$
Set initial joint configuration vector $\boldsymbol{\theta}_{\text{init}}$ and reachability error tolerance
for $j = 1$ to m **do**
  Solve inverse kinematics: $\boldsymbol{\theta}_{\text{sol},j} \leftarrow \text{IKS}(\mathbf{T}_j, \boldsymbol{\theta}_{\text{init}})$
  if $err(\mathbf{T}_j, \text{FK}(\boldsymbol{\theta}_{\text{sol},j})) \leq$ tolerance **then**
    while CollCheck$(\boldsymbol{\theta}_{\text{sol},j}, \textbf{Geo}) ==$ true **do**
      Update $\boldsymbol{\theta}_{\text{init}}$ via null-space repulsive gradient
      Re-solve: $\boldsymbol{\theta}_{\text{sol},j} \leftarrow \text{IKS}(\mathbf{T}_j, \boldsymbol{\theta}_{\text{init}})$
      if no feasible solution within limited iterations **then**
        break
      end if
    end while
    if CollCheck$(\boldsymbol{\theta}_{\text{sol},j}, \textbf{Geo}) ==$ false **then**
      Calculate static torque norm $\|\boldsymbol{\tau}_j\|$ for jth test point
      $\boldsymbol{\theta}_{\text{init}} \leftarrow \boldsymbol{\theta}_{\text{sol},j}$ {warm-start next point}
    end if
  else
    Attempt randomization for a limited number of steps
    if no feasible solution within operational space **then**
      break
    end if
  end if
end for
Calculate fitness objectives f_1, f_2
Return f_1, f_2

The IK is solved using the non-linear constrained optimization approach, formulation following the approach introduced by [38], which evaluates the forward kinematics at each iteration to compute the task-space error and provides numerically robust convergence without requiring explicit singularity handling. This formulation includes redundant DoF through the damped pseudoinverse of the Jacobian [30,39]. To achieve the solution consistency across spatially adjacent task poses, a well-known challenge in redundant IK where infinitely many valid configurations exist for each target [30], the solver uses a continuation strategy while checking for reachability solutions. This method comprises the converged joint vector $\boldsymbol{\theta}_{\text{sol},j-1}$ from the previous task point serves as the initial seed for the next, $\boldsymbol{\theta}_{\text{init},j} = \boldsymbol{\theta}_{\text{sol},j-1}$. Because the task points are ordered by cluster adjacency, this warm-starting naturally favours smooth configuration transitions and suppresses discontinuous jumps between IK solution across different reachability test points. When the warm-started solution results in a collision, a null-space repulsive gradient is used to drive the initial joint configuration $\boldsymbol{\theta}_{\text{init}}$ away from the obstacle, following the redundancy resolution framework of [40]. This allows the robot to find a collision-free posture that satisfies the positional requirement without violating hard joint limits. If no feasible solution is obtained within a bounded number of attempts, the point is recorded as unreachable and penalised in the fitness evaluation. Furthermore, a tube constraint is applied as a logical mask: any joint i positioned outside of the vacuum vessel volume before the deployment port is treated as unactuated ($\dot{\theta}_i = 0$), effectively reducing the active degrees of freedom until the link enters the main vessel workspace.

4. Results and discussion

The algorithm is implemented and tested in MATLAB R2024a using the multi-objective NSGA-II function [41], executed for a maximum of 50 generations with a population size of 70, function tolerance is set to 0.005, and maximum stall generations is set to 5 for stopping conditions. These values are selected based on a balance between computational efficiency and solution quality, ensuring sufficient exploration of the search space without sacrificing too much computational cost. The Pareto fraction is set to 0.35, providing a balanced selection of non-dominated solutions. A crossover fraction of 0.8 is applied to promote genetic diversity while maintaining a reasonable convergence speed. As previously described, for each candidate solution, a kinematic tree is generated, and the cost function is evaluated as detailed in Section 3 and with constraints mentioned in Section 2.

A 6-DoF case is initially evaluated to meet the design objectives; however, no feasible solution is obtained because reachability points near the entrance port resulted in collisions with the environment. It is concluded that the manipulator must necessarily accommodate redundant degree of freedoms to comply with the shape of the operational space and design constraints. The redundancy also contributes to finding a solution for any reachability point without self-collision and collision with the environment.

To evaluate the effectiveness of the proposed framework, independent optimization runs are conducted for each cluster resolution ($k = 4, 8, 16$).The optimization results for the 7 DoF case are presented in Fig. 7. In this figure, the x-axis represents the first objective value, which is the sum of the link lengths, while the y-axis represents the second objective, which is the total torque requirement at all reachable points under static conditions. The blue points indicate all candidate solutions with a valid inverse kinematic solution under the specified constraints, while the green point represents the Pareto front solution obtained with the GA.

In each case, the reachability points are obtained by clustering the CAD-derived surface points and the associated normals as described in Section 2. In each plot of Fig. 7, the elbow point (trade-off solution) is highlighted, representing a balanced compromise between minimizing

Table 1
Elbow-point (trade-off) solutions across k.

k	f_1 (m)	f_2 (Nm)	Joint Type (binary)	Link Lengths (m)
4	5.4980	570.6858	1011101	1.0298 0.4860 0.7904 0.6256 0.8553 0.9104 0.7982
8	5.4831	675.98	1011010	0.7492 0.7807 0.5226 0.7294 0.5986 1.0031 1.0977
16	5.5238	642.8205	1011011	0.9272 0.6104 0.5879 0.7014 0.7170 0.8999 1.0786

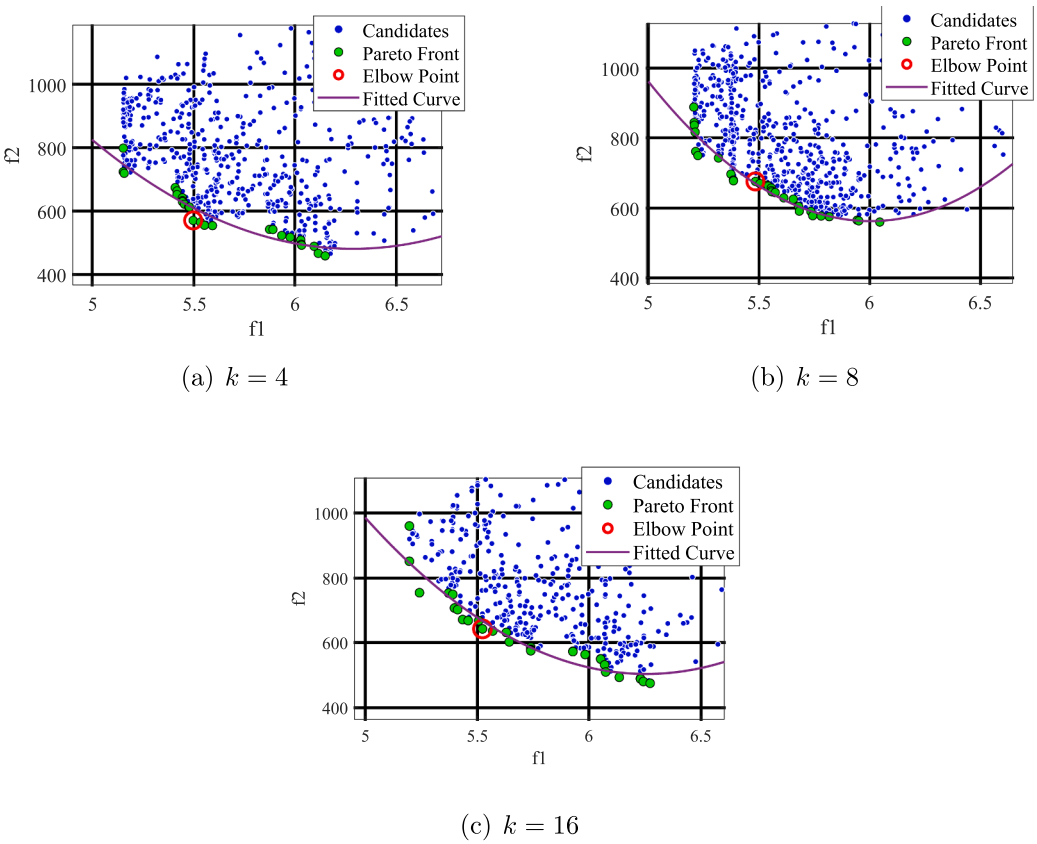

Fig. 7. Pareto front solutions for k = 4, k = 8, and k = 16, showing the distribution of feasible solutions in the (f_1, f_2) objective space. Red markers indicate the elbow point solution in each case.

f_1 and f_2. This is selected using Chebyshev scalarization method [42], in which each Pareto solution is normalized, and the one minimizing the maximum of the two normalized objectives is chosen [43].

To provide a consistent benchmark across cluster cases, the elbow-point (trade-off) solution from each Pareto front is presented. Table 1 summarises the elbow-point (trade-off) solutions obtained for $k = 4$, $k = 8$, and $k = 16$ clusters.

All three elbow-point solutions converge to similar joint-types, reflecting a strong bias towards proximal yaw joints interleaved with distal pitch joints. This structure supports the earlier observation that yaw joints placed proximally reduce actuator torque demands by leveraging gravitational support. Across all cases, the total manipulator length stabilises around 5.48-5.53 m, while the static torque requirement remains consistently high (~570-675 Nm). The joint type indicates that yaw joints (1) dominate the proximal links, with pitch joints (0) used sparingly in the middle of the chain. In all three cases, the optimal solutions exhibit proximal placement of yaw joints, which reduces actuator torque requirements by transferring gravitational loads to the manipulator structure instead of the actuators. This observation is consistent with the trends identified for the 7 DoF configuration results, reinforcing the design principle that proximal yaw joints are beneficial in long-reach manipulators.

Beyond identifying a specific optimal configuration, the Pareto fronts generated across the varying cluster resolutions ($k = 4, 8, 16$) demonstrate the framework's capacity to extract fundamental engineering heuristics for the system with a case study. Instead of merely converging on a single result, the approach illuminates the non-linear trade-offs between total link length, joint sequencing, and actuator torque requirements.

- **Low-torque, long-length solutions** Typically $f_1 \approx 5.80$–6.30 m with $f_2 \approx 450$–550 Nm. These designs utilize proximal yaw joints to leverage the structure for gravitational support, significantly reducing actuator demand.
- **High-torque, short-length solutions** Characterized by $f_1 \approx 5.20$–5.35 m and $f_2 \gtrsim 700$ Nm. These compact configurations often employ a base pitch joint, which increases joint torque due to reduced structural leverage against gravity.
- **Elbow-point (trade-off) designs** for each k value fall consistently around $f_1 \approx 5.40$–5.70 m and $f_2 \approx 570$–700 Nm. These configurations are characterised by a prismatic joint followed by yaw and pitch joints, where yaw remains dominant in the proximal portion of the chain. This structure balances reachability and torque efficiency, yielding compact yet mechanically feasible manipulators.

Fig. 8 illustrates the solution at the elbow point for the $k = 8$ cluster. In the visualisation, yaw joints are shown in red and pitch joints in green, while the manipulator tip is represented by a cone indicating the field of view. On the CAD-extracted vessel surface, the designated entrance port is marked by a circle, through which the robot is deployed using the base prismatic actuator.

The optimized elbow point kinematic structure can be compared against existing manipulators designed for similar confined environments. The Super Dragon [7], the AIA [8] that employs serial chains with identical consecutive revolute link sections. However, prismatic translational deployers accompanied with initial yaw joints have been adopted for larger heavy-duty fusion robots, such as the CFETR multi-purpose overload robot (CMOR) [32], a 9-DOF serial manipulator with a total length of up to 10.8 m, deployed via a prismatic rail system

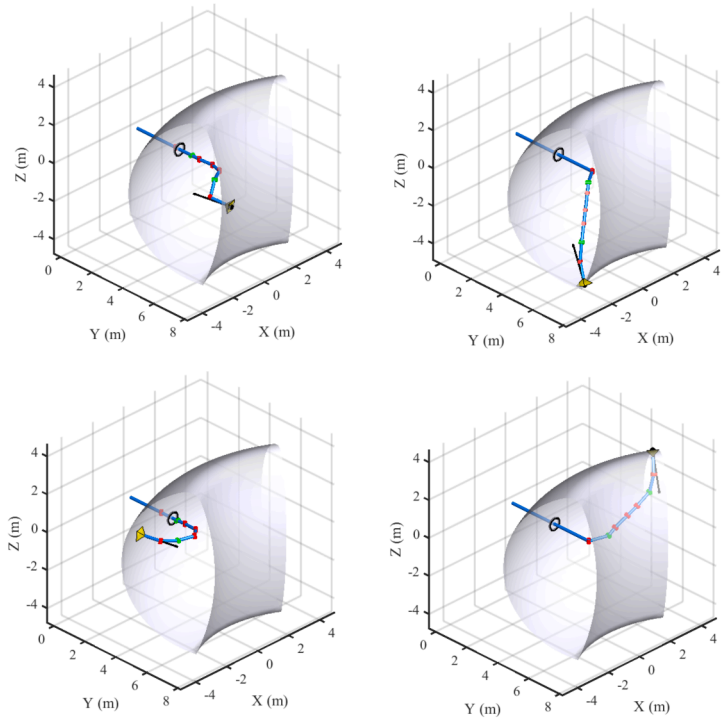

Fig. 8. Four configurations for reachability test points of the optimized manipulator given in Table 1 for k = 8.

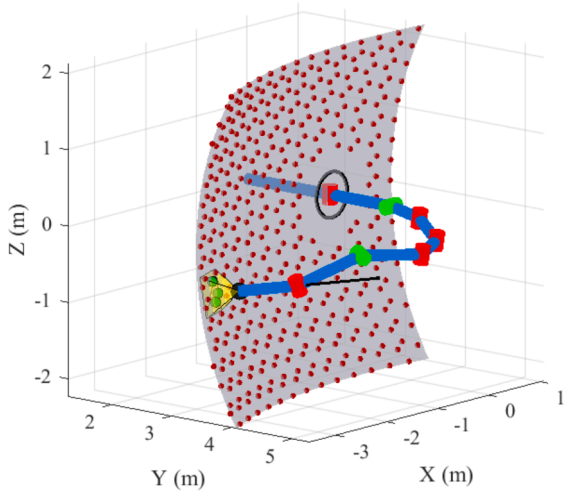

Fig. 9. Graphical representation of the coverage metric.

for heavy-duty maintenance within a ±45° sector of the vacuum vessel. Similar designs with initial yaw joints have been adapted for larger deployers [44,45] to enable remote maintenance of fusion reactors. However, to the authors' knowledge, no optimized kinematic structure for a lightweight rapid inspection manipulator in fusion environments has been reported in the literature, lacking a direct comparison of the resulting topology.

4.1. Comparative results for $k = 4$, $k = 8$, and $k = 16$

To assess the stability of the optimization, three independent runs are conducted for each cluster resolution. The elbow solutions are presented in Table 2 with mean values and standard deviations for each run. All computations are performed on a laptop equipped with an Intel Core i7-13700H (2.40 GHz) processor and 16 GB of RAM. Across the resolutions, the total manipulator length (f_1) stabilized between 5.45 m and 5.65 m, while the static torque (f_2) remained within a consistent range of 570-676 Nm. Notably, the optimized joint configurations across different runs consistently prioritized proximal yaw joints followed by a pitch joint in all of the results, reinforcing the design principle that such an arrangement better handles gravitational loads in long-reach systems. While the computational cost increases with the number of clusters, rising from approximately 5 h for $k = 4$ to more than 10 h for $k = 16$, the convergence of the f_1 and f_2 values shows that even lower cluster resolutions can still identify viable and consistent kinematic candidates.

Table 2
Summary of optimization results across three runs for different cluster resolutions (k).

Clusters (k)	Mean f_1 [m]	Mean f_2 [Nm]	Mean Runtime
4	5.52 ± 0.08	581.77 ± 16.5	5 h 02 m 39 s
8	5.51 ± 0.07	657.64 ± 32.2	6 h 50 m 42 s
16	5.55 ± 0.09	631.88 ± 21.3	10 h 29 m 34 s

When evaluating three cluster resolutions, $k = 4, 8, 16$, doubles the test points while systematically monitoring convergence, the aim is to identify the smallest k that preserves pose coverage (within position/normal tolerances) and yields a stable Pareto front. Therefore, for the selected geometry and tolerances, $k = 4$ provides the fastest scan, while $k = 8$ can be reserved for cases requiring exceptionally fine surface detail and to observe the spread over a large number of test points. On the other hand, $k = 16$ demonstrates that an unnecessarily dense selection of points does not contribute to improving results while increasing the computational cost. This result confirms that the conventional approach of selecting a large number of uniformly sparse test points is unnecessary and often undesired.

4.2. Comparative results with other sampling methods

To evaluate the efficiency of the proposed methodology, a comparative study is conducted against three standardized spatial sampling baselines: Farthest Point Sampling (FPS) [24], spatial K-medoid [46], and Uniform Voxel-Grid downsampling [47]. Each method is evalu-

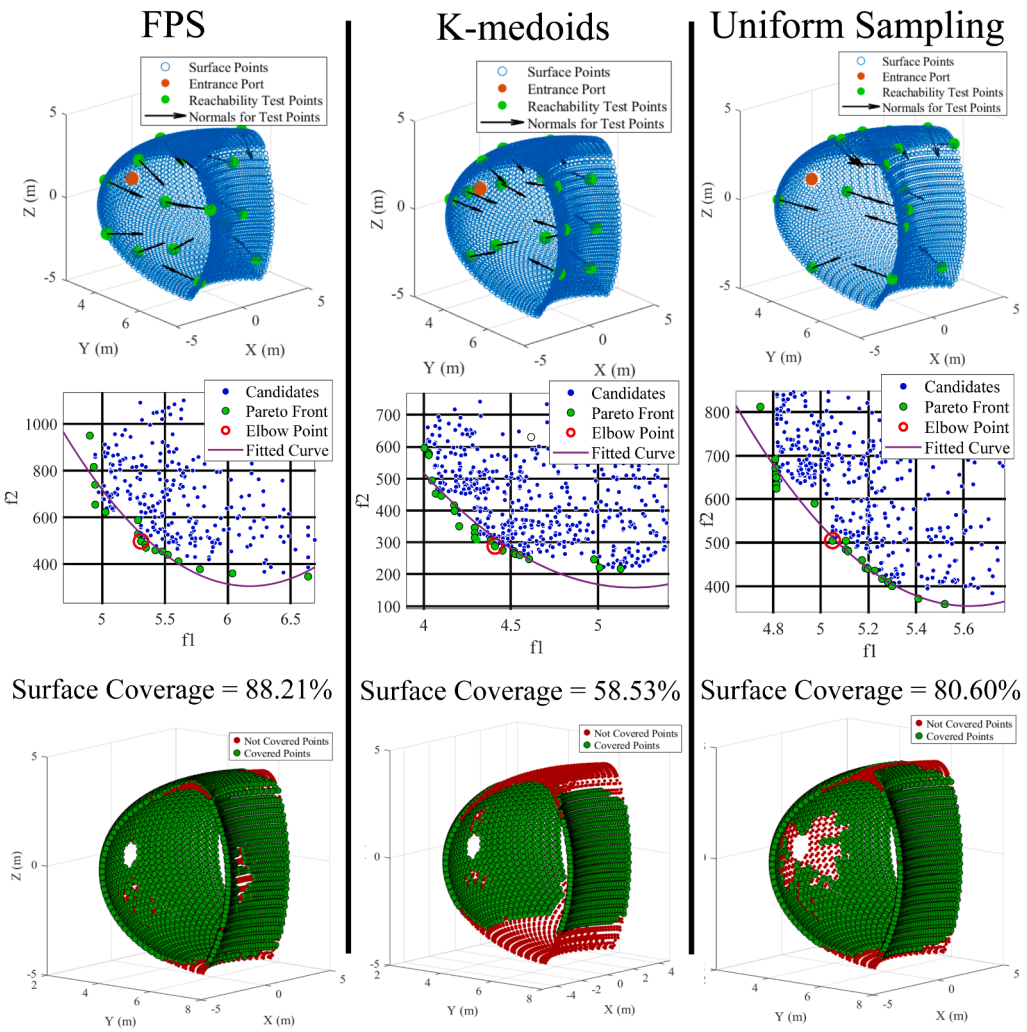

Fig. 10. Comparative analysis of optimized kinematic structures using different surface downsampling methods: Farthest Point Sampling (FPS), K-medoids, and Uniform Sampling. (Top) Distribution of reachability test points ($N = 16$) mapped onto the vacuum vessel (VV) surface for each method. (Middle) Pareto front plots illustrating the trade-off between the two optimization objectives for each sampling configuration. (Bottom) Evaluation of the chosen elbow-point structures based on the Surface Coverage ($Cov\%$) metric, calculated across a validation set of 1500 uniform surface points as illustrated in Fig. 9. Detailed parameters for the optimized configurations are provided in Table 3.

ated at sampling numbers of $N = (16, 32, 64)$, which is the same number of reachability test points used in the optimization with $k = (4, 8, 16)$. For the proposed surface clustering method, each of the k clusters contributes one medoid and three boundary points, resulting in $N = 4k$ reachability test points (i.e., $N = 16$, 32, and 64 for $k = 4$, 8, and 16, respectively). The baseline methods are evaluated at matching sample sizes to ensure a fair comparison.

For each optimized kinematic structure, inverse kinematics is solved for all 1500 validation points. Three metrics are then computed: Position Error (PE%), the percentage of the 1500 points for which the end-effector positional error exceeds 0.01 m; Orientation Error (OE%), the percentage for which the angular deviation between the tool axis and the surface normal exceeds 25 deg; and Surface Coverage (Cov%), the percentage of points that can be enveloped by the sampled poses. These thresholds are selected to align with the strict requirements for kinematic positioning in remote maintenance. While real-world operations will incorporate additional error contributions (such as link deflections, joint backlash, and control/feedback variances) the tolerances for the kinematic optimisation (PE and OE) are kept intentionally tighter than expected physical positioning error for kinematic checking. This ensures that the generated kinematic structures provide a robust foundation for the overall positioning requirements.

The coverage metric is evaluated by simulating a truncated pyramid Field-of-View (FOV) volume at each successfully reached pose. As illustrated in Fig. 9, each end-effector configuration is modeled with an 80° FOV and an operational depth ranging from 0.1 m to 0.4 m. For each target, a reachability test point is defined 0.3 m away from the surface along the normal vector (indicated in black). A surface point is classified as covered (indicated in green) if it resides within the intersection of the end-effector's FOV volume and the target surface. The final metric, $Cov\%$, is calculated as the ratio of the number of covered points to the total number of points in the validation set. This metric effectively quantifies the capacity of a specific optimized kinematic structure to envelop the operational area using the same discrete set of sampled poses for the visual inspection case.

In Fig. 10 the case for $N = 16$ for the sampling methods used for comparison is illustrated by their reachability point, Pareto plot, and surface coverage representation. Furthermore, Table 3 presents a comparison of all methods, including the proposed method, with numerical results for the performance metrics (PE%, OE%, and Cov%).Notably, the FPS method appears to perform better when the Pareto-optimal elbow solution is evaluated across 1500 reachability points. The hybrid Surface Clustering method consistently outperforms spatial-only baselines by leveraging normal-vector alignment and specific reachability

Table 3
Performance evaluation of kinematic structures optimized for reachability test points (N) using different down-sampling methods.

Method	N = 16			N = 32			N = 64		
	PE%	OE%	Cov%	PE%	OE%	Cov%	PE%	OE%	Cov%
Uniform Grid Sampling	12.40	19.33	80.60	5.93	7.53	87.94	2	4.33	96.16
FPS	11	10.27	88.21	4.00	4.60	90.43	0.40	2.73	97.51
K-Medoids	40.60	36.07	58.53	24	19.47	72.83	5.00	5.67	93.58
Surface Clustering	2.47	6.67	97.46	2.33	4.73	97.95	1.80	3.94	98.61

distances, ensuring superior geometric fidelity even at low sampling densities.

The comparative results in Table 3 provide a practical guideline for early-stage design synthesis. While standard heuristic methods such as Uniform Sampling and FPS have been utilized uniform spatial distribution [24], they are fundamentally task-agnostic and can lead to performance degradation in constrained robotics applications [29]. As N increases to 64 for $k = 16$, these methods begin to approximate the performance metrics of the proposed method. However, at lower resolutions ($N = 16$) for $k = 4$, Surface Clustering significantly outperforms alternatives, achieving 97.46% coverage compared to only 58.53% for K-Medoids, 80.60% for a Uniform Grid, or 88.21% for FPS. Considering the scaling of computational time from $k = 4$ to $k = 8$, the proposed clustering approach enables an effective preliminary scan. By identifying stable kinematic structures at low cluster counts without requiring high-fidelity fine-tuning, the framework reduces early-stage design runtime.

5. Conclusion

This study presented a kinematic optimization framework for long-reach articulated mechatronic systems, i.e., robotic manipulators, specialised for service applications inside industry confinements through port access. The proposed design framework is presented based on the use case of inspection inside a fusion vacuum vessel. The core novelty lies in a hybrid surface-clustering of CAD geometries (combining position, normal, and distance-to-port) with a globally aware selection of a specific number of points, and a lightweight PCA-plane projected collision checking. With these, the kinematic optimization pipeline with NSGA-II is introduced to ensure these components produce compact, coverage-preserving reachability sets that significantly reduce computational cost while maintaining task fidelity. While the model used for the case study provides a foundational geometry for identifying optimal link-joint configurations. The representation serves as a necessary filtering stage to isolate high-performing candidates for further engineering refinement.

After optimization, once Pareto outcomes stabilized, the designer can fix the smallest cluster count that preserves coverage and then efficiently explore kinematic constraints (e.g. changing maximum allowed link lengths, inspection tolerances, or viewing distances) at substantially reduced computational cost. The framework demonstrates that clustering-based task reduction facilitates systematic, early-stage kinematic synthesis for confined environments, while remaining adaptable to changes in vessel geometry, constraints, or inspection requirements.

With the case study presented for the nuclear fusion vacuum vessel inspection, a multi-objective NSGA-II jointly optimizes the manipulator's kinematic topology and dimensions on the reduced task sets to minimize total length (f_1) and static torque (f_2). Three independent optimization runs for each cluster resolution ($k = 4, 8, 16$) confirm the framework's robustness, with elbow solutions consistently converging to total link lengths of ~5.45–5.65 m and proximal yaw joints emerging as the dominant structural trend across all runs. The comparative results confirm that conventional sampling methods can also identify satisfactory kinematic structures when a sufficiently large number of task points is used. However, the proposed surface clustering approach achieves consistent results with a significantly reduced set of reachability points, which is particularly advantageous for iterative design workflows where vessel geometry, port dimensions, or inspection requirements may evolve, enabling rapid re-evaluation without the computational cost of dense sampling.

This study establishes a robust pipeline for early-stage kinematic synthesis, utilizing simplified CAD models to maintain the computational efficiency necessary for the high-volume iterations of the Genetic Algorithm. As such, effects including structural deflection, vibrations, joint backlash, and control accuracy are not captured in the current framework and represent limitations that must be addressed before physical implementation. Future research will focus on integrating these optimized structures into high-fidelity multi-physics environments to evaluate dynamic performance, including structural vibrations and deflection under varying payloads, which are critical factors for long-reach manipulators in fusion environments. Parallel to these simulations, further investigation into remote actuation methods and hardware feasibility tests will be conducted to ensure the optimized structures can be physically realized within the reactor's stringent environmental constraints. Building upon the preliminary IK strategy presented here, subsequent work will evolve the framework into a trajectory planner. This follow investigation and implementation of advanced control and active vibration suppression strategies to ensure both collision avoidance and precision positioning. Finally, the optimized structures identified through this methodology will be validated via experimental testing on a physical prototype to bridge the gap between kinematic synthesis and operational reality.

CRediT authorship contribution statement

Emre Uzunoglu: Writing – original draft, Validation, Software, Methodology, Conceptualization; **Dohee Lee:** Writing – review & editing, Investigation, Conceptualization; **Luca Raimondi:** Writing – original draft, Methodology, Investigation, Conceptualization; **Kaiqiang Zhang:** Writing – review & editing, Supervision, Methodology, Conceptualization; **Hongtack Kim:** Investigation, Conceptualization; **Youngmin Park:** Investigation, Conceptualization; **Kwonhee Hong:** Investigation, Conceptualization; **Namil Her:** Supervision, Funding acquisition; **Robert Skilton:** Supervision, Funding acquisition.

Data availability

The authors do not have permission to share data.

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.

Acknowledgements

This work is funded by the UK Department for Science, Innovation and Technology via the International Science Partnerships Fund, and the Republic of Korea's Ministry of Science and ICT under the grant PG2402.

The research leading to some outcomes are supported by the UKAEA/EPSRC Fusion Grant 2022/27 EP/W006839/1. The views and opinions expressed herein do not necessarily reflect those of the UK Atomic Energy Authority.

References

[1] A. Pistone, D. Ludovico, L. De Mari Casareto Dal Verme, S. Leggieri, C. Canali, D.G. Caldwell, Modelling and control of manipulators for inspection and maintenance in challenging environments: a literature review, Ann. Rev. Control 57 (2024) 100949. https://doi.org/10.1016/j.arcontrol.2024.100949

[2] K. Dandan, A. Ananiev, I. Kalaykov, SIRO: the silos surface cleaning robot concept, in: 2013 IEEE International Conference on Mechatronics and Automation, IEEE, 2013, pp. 657–661.

[3] J. Park, D.H. Kim, J.W. Seo, D.C. Lee, S.H. Kim, G.W. Chu, C.K. Hong, M.S. Kim, Y.S. Suh, S. Han, Design of robotic system for inspection, maintenance and repair of LNG cargo tank, in: Proceedings of the International Ocean and Polar Engineering Conference, ISOPE International Society of Offshore and Polar Engineers, 2014, pp. 906–912.

[4] W. Chun, R. Rimando, W. Hamel, DOE robot perspective: past, present, and future, Nucl. Sci. Eng. 199 (8) (2025) 1273–1291.

[5] I.T. Chapman, N.R. Walkden, An overview of shared technical challenges for magnetic and inertial fusion power plant development, Philos. Trans. R. Soc. A 379 (2021) 20200019. https://doi.org/10.1098/rsta.2020.0019

[6] I.T. Chapman, A.W. Morris, UKAEA capabilities to address the challenges on the path to delivering fusion power, Philos. Trans. R. Soc. A 377 (2019) 20170436. https://doi.org/10.1098/rsta.2017.0436

[7] G. Endo, A. Horigome, A. Takata, Super dragon: a 10-m-long coupled tendon-driven articulated manipulator, IEEE Rob. Autom. Lett. 4 (2) (2019) 934–941. https://doi.org/10.1109/LRA.2019.2894855

[8] Y. Perrot, others, ITER articulated inspection arm (AIA): R&D progress on vacuum and temperature technology for remote handling, Fusion Eng. Des. 75-79 (2005) 537–541. https://doi.org/10.1016/j.fusengdes.2005.06.028

[9] M. Russo, L. Raimondi, X. Dong, D. Axinte, J. Kell, Task-oriented optimal dimensional synthesis of robotic manipulators with limited mobility, Robot. Computer-Integrated Manuf. 69 (2021) 102096. https://doi.org/10.1016/j.rcim.2020.102096

[10] Y. He, M. Yang, Z. Xu, S. Li, B. Zhang, Design of tunnel steel arch looping manipulator with multiple actuators in limited space, Sci. Prog. 106 (2023) 00368504231180025. https://doi.org/10.1177/00368504231180025

[11] M. Schappler, P. Jahn, A. Raatz, T. Ortmaier, Combined structural and dimensional synthesis of a parallel robot for cryogenic handling tasks, in: T. Schüppstuhl, K. Tracht, A. Raatz (Eds.), Annals of Scientific Society for Assembly, Handling and Industrial Robotics 2021, Springer, Cham, 2022, pp. 65–77. https://doi.org/10.1007/978-3-030-74032-0_6

[12] S.B. Liu, M. Althoff, Optimizing performance in automation through modular robots, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2020, pp. 4044–4050. https://doi.org/10.1109/ICRA40945.2020.9197110

[13] S. Kucuk, Z. Bingul, Robot workspace optimization based on a novel local and global performance indices, in: Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE), Vol.4, 2005, pp. 1593–1598. https://doi.org/10.1109/ISIE.2005.1529155

[14] D. Ramirez, J. Kotlarski, T. Ortmaier, Combined structural and dimensional synthesis of serial robot manipulators, in: V. Parenti-Castelli, W. Schiehlen (Eds.), ROMANSY 21 - Robot Design, Dynamics and Control, Springer, Cham, 2016, pp. 207–216. https://doi.org/10.1007/978-3-319-33714-2_23

[15] S. Ha, S. Coros, A. Alspach, J.M. Bern, J. Kim, K. Yamane, Computational design of robotic devices from high-level motion specifications, IEEE Trans. Rob. 34 (5) (2018) 1240–1251. https://doi.org/10.1109/TRO.2018.2830419

[16] K. Kawaharazuka, T. Makabe, K. Okada, M. Inaba, Daily assistive modular robot design based on multi-objective black-box optimization, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2023, pp. 9970–9977. https://doi.org/10.1109/IROS55552.2023.10342041

[17] J. Whitman, R. Bhirangi, M. Travers, H. Choset, Modular robot design synthesis with deep reinforcement learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol.34, 2020, pp. 10418–10425. Issue: 06. https://doi.org/10.1609/aaai.v34i06.6611

[18] G. Chen, X. Xu, L. Wang, W. Zhang, A multi-objective optimization design method of shift manipulator for robot driver using SA-PSA, Struct. Multidiscip. Optim. 65 (2022) 204. https://doi.org/10.1007/s00158-022-03301-1

[19] J.D. Sanjuan De Caro,others, Evaluation of objective functions for the optimal design of an assistive robot, Micromachines 13 (12) (2022) 2206. https://doi.org/10.3390/mi13122206

[20] O.W. Maaroof, M.I.C. Dede, L. Aydin, A robot arm design optimization method by using a kinematic redundancy resolution technique, Robotics 11 (1) (2022) 1. https://doi.org/10.3390/robotics11010001

[21] M.J. Bakari, K.M. Zied, D.W. Seward, Development of a multi-arm mobile robot for nuclear decommissioning tasks, Int. J. Adv. Rob. Syst. 4 (2007) 51. https://doi.org/10.5772/5621

[22] C. Canali, A. Pistone, D. Ludovico, P. Guardiani, R. Gagliardi, L. De Mari Casareto Dal Verme, others, Design of a novel long-reach cable-driven hyper-redundant snake-like manipulator for inspection and maintenance, Appl. Sci. 12 (7) (2022) 3348. https://doi.org/10.3390/app12073348

[23] M. ManoahStephen, et al., In-vessel inspection system: development and testing activities of high vacuum and temperature technologies for fusion remote handling, Fusion Eng. Des. 202 (2024) 114368. https://doi.org/10.1016/j.fusengdes.2023.114368

[24] Y. Eldar, M. Lindenbaum, M. Porat, Y.Y. Zeevi, The farthest point strategy for progressive image sampling, IEEE Trans. Image Process. 6 (9) (1997) 1305–1315.

[25] C.R. Qi, L. Yi, H. Su, L.J. Guibas, PointNet++: deep hierarchical feature learning on point sets in a metric space, in: Advances in Neural Information Processing Systems vol. 30 (2017).

[26] J. Li, J. Zhou, Y. Xiong, X. Chen, C. Chakrabarti, An adjustable farthest point sampling method for approximately-sorted point cloud data, in: 2022 IEEE Workshop on Signal Processing Systems (SiPS), IEEE, 2022, pp. 1–6. https://doi.org/10.1109/SiPS55648.2022.9975149

[27] O. Dovrat, I. Lang, S. Avidan, Learning to sample, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2760–2769.

[28] I. Lang, A. Manor, S. Avidan, SampleNet: differentiable point cloud sampling, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 7578–7588.

[29] P. Zhang, R. Xie, J. Sun, W. Li, Z. Su, AS-PD: An Arbitrary-Size Downsampling Framework for Point Clouds, (2023). arXiv preprint arXiv:2211.01110.

[30] B. Siciliano, O. Khatib, T. Kröger (Eds.), Springer Handbook of Robotics, Springer, Berlin, 2008.

[31] C. Bachmann, L. Ciupinski, C. Gliss, T. Franke, T. Härtl, P. Marek, F. Maviglia, R. Mozzillo, R. Pielmeier, T. Schiller, P. Spaeh, T. Steinbacher, M. Stetka, T. Todd, C. Vorpahl, Containment structures and port configurations, Fusion Eng. Des. 174 (2022) 112966. https://doi.org/10.1016/j.fusengdes.2021.112966

[32] Z. Yao, H. Wu, Y. Yang, Y. Cheng, H. Pan, T. Zhang, R. Yin, On-line precision control of CFETR multipurpose overload robot using deformation model, Fusion Eng. Des. 174 (2022) 112967. https://doi.org/10.1016/j.fusengdes.2021.112967

[33] International Atomic Energy Agency, Articulated Inspection Arm for ITER, a demonstration in the Tore Supra Tokamak, Technical Report INIS-FR–36023948, CEA/Cadarache, Association EURATOM-CEA, 2004. Presented at the ITER Remote Handling Meeting, outlining the motorized carriage and additional linear joint deployment system. https://inis.iaea.org/records/nvxrr-v6996/files/36023948.pdf.

[34] J. Chalfoun, C. Bidard, D. Keller, Y. Perrot, G. Piolain, Design and flexible modeling of a long reach articulated carrier for inspection, in: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2007, pp. 4013–4019.

[35] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197. https://doi.org/10.1109/4235.996017

[36] A.M. Ikotun, A.E. Ezugwu, L. Abualigah, B. Abuhaija, J. Heming, et al., K-means clustering algorithms: a comprehensive review, variants analysis, and advances in the era of big data, Inf. Sci. 622 (2023) 178–210. https://doi.org/10.1016/j.ins.2022.11.139

[37] D. Arthur, S. Vassilvitskii, K-means++: the advantages of careful seeding, in: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), SIAM, 2007, pp. 1027–1035.

[38] T. Sugihara, Solvability-unconcerned inverse kinematics by the Levenberg–Marquardt method, IEEE Trans. Rob. 27 (5) (2011) 984–991. https://doi.org/10.1109/TRO.2011.2148230

[39] Y. Nakamura, H. Hanafusa, Inverse kinematic solutions with singularity robustness for robot manipulator control, ASME J. Dyn. Syst. Meas. Control 108 (3) (1986) 163–171. https://doi.org/10.1115/1.3143764

[40] A.A. Maciejewski, C.A. Klein, Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments, Int. J. Rob. Res. 4 (3) (1985) 109–117.

[41] MathWorks, gamultiobj - Multiobjective genetic algorithm, 2025, Accessed 22 September 2025. https://www.mathworks.com/help/gads/gamultiobj.{html}.

[42] S. Wei, M. Niethammer, The fairness-accuracy pareto front, Stat. Anal. Data Min. ASA Data Sci. J. 15 (3) (2022) 287–302. https://doi.org/10.1002/sam.11560

[43] I. Giagkiozis, P.J. Fleming, Methods for multi-objective optimization: an analysis, in: 2015 IEEE Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2015, pp. 2820–2827.

[44] C.H. Choi, R. Hery, J. Palmer, Concept of operation of the agile robot transporter for ITER in-vessel maintenance, Fusion Eng. Des. 202 (2024) 114384. https://doi.org/10.1016/j.fusengdes.2024.114384

[45] D. McGarrigle, E. Flynn, C. Kennedy, A. Loving, S. Budden, Analysis of existing and proposed maintenance deployment systems toward DEMO MPD development, IEEE Trans. Plasma Sci. 50 (11) (2022) 4485–4490. https://doi.org/10.1109/TPS.2022.3163466

[46] MathWorks, kmedoids - k-medoids clustering algorithm, 2025, Accessed 01 March 2025. https://www.mathworks.com/help/stats/kmedoids.html.

[47] MathWorks, pcdownsample- Downsample 3-D point cloud algorithm, 2025, Accessed 01 March 2025. https://www.mathworks.com/help/vision/ref/pcdownsample.html.