2023 IEEE/ASME International Conference on
Advanced Intelligent Mechatronics (AIM)

Template-free Non-revisiting Uniform Coverage Path Planning on Curved Surfaces

Tong Yang[1] *Member, IEEE*, Jaime Valls Miro[2] *Member, IEEE*,
Minh Nguyen[2], Yue Wang[1] *Member, IEEE*, and Rong Xiong[1] *Member, IEEE*

***Abstract*—A novel mechanism to generate non-revisiting uniform coverage (NUC) paths on arbitrarily shaped object surfaces is presented in this work. Given a non-planar surface, non-zero curvature makes traditional homeomorphic fitting of regular template coverage paths from planar regions onto the object surface non-distance-preserving. Any coverage path with a realistic tooling size derived in this way will suffer from overlaps and missing gaps when transformed onto the object surfaces, unable to uniformly cover the target. To overcome this, a discretisation process is adopted to represent the object surface as a uniform unstructured mesh, with resolution set in accordance to the tool size. It is proven that a coverage skeleton path must exist by mesh subdivision refinement which, after a local optimisation step to improve overlap, missing gaps and smoothness, gives rise to template-free superior NUC paths. Extensive simulation examples are presented to prove the validity of the proposed strategy in realistic settings. The proposed scheme is able to achieve 95.9% coverage on benchmark surface tests, outperforming comparable coverage algorithms such as a homeomorphic boustrophedon mapping which can at best achieve 80.9% coverage, or more recent state-of-the-art methods able to reach 94.7% coverage. An accompanying video is supplied with examples, including a real-world implementation of a NUC path tracked by a manipulator. An open-source implementation has been made available.**

***Index Terms*—Coverage Path Planning, Uniform Coverage, Mesh Subdivision**

I. Introduction

THE *Coverage Path Planning* (CPP) [1] task on a non-planar surface seeks to reveal a path such that the number of traversed points is maximised. The problem is present in a wide range of industrial applications, such as terrain coverage [2], bathymetric surveying [3], surface inspection [4] or agricultural field mapping [5]. In certain situations, uniform surface coverage [6] becomes a critical feature for the resulting path. This is a special case that takes into consideration the area effected by the tool on the surface and seeks to maximise the overall surface covered whilst minimising the overlap between tool traces. This is apparent for instance in painting tasks where overspray results in uneven coating and wasted material [7], contact-based surface inspection with inefficient revisiting [8], or abrasive surface finishing processes such as marine growth water-blasting removal from hulls, where minimal over-coverage prevents fatigue cracks [9]. This specialised application, referred to as the *Non-revisiting Uniform Coverage* (NUC) task, is the objective of this work.

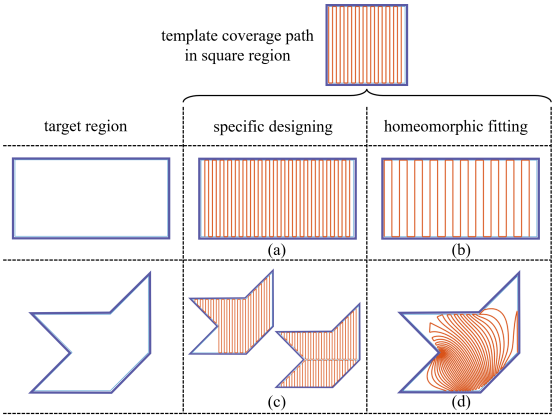


Fig. 1. Problem illustration with the generalising of a back-and-forth square template coverage path in a rectangular (top row) and arbitrary polygon target planar regions (bottom row). (a) A specific coverage path design where the number of path turns is calculated in accordance with the length of the rectangle (i.e. not as established by the template coverage path). (b) The coverage path generated by homeomorphic fitting, which cannot uniformly cover the region, gaps appearing between the path slices as a result of horizontal stretching. (c) An example of the specific path design, illustrating how (top) for arbitrary regions they do not always exist, and (bottom) where cellular decomposition is adopted, separating the target region into sub-cells and designing a coverage path within each one. (d) A conformal area fitting mapping homeomorphism (Schwarz-Christoffel [10]) can be easily fitted into an arbitrary homeomorphic region. However, the coverage path is no longer uniform in the target region, leading to overlaps when using a non-zero size tracing tool, and missing gaps between the path slices.

CPP for planar regions has been intensively investigated. The most popular strategies are based on *cellular decomposition*. For polygonal regions, the area is often firstly decomposed into cells with elemental shapes such as trapezoids, rectangles, and circular regions, and then independently covered by designing template coverage paths within each cell [9], as depicted in Fig. 1(a) and (c). When the region cannot be suitably approximated by polygons, grid-based discretisation [11] is adopted to approximate the coverable region into grids, and template coverage paths [12] are designed in the grid-maps.

An alternative set of algorithms generalise the coverage solution of standard template regions into arbitrary target regions, where the key step is a homeomorphic transformation between the two regions. Two simple examples of planar homeomorphisms from a unit square to a rectangle and an

[1] Tong Yang, Yue Wang, and Rong Xiong are with the State Key Laboratory of Industrial Control and Technology, Zhejiang University, P.R. China. This work is supported by the National Key R&D Program of China under Grant 2021ZD0114500.

[2] Jaime Valls Miro and Minh Nguyen are with the Robotics Institute, University of Technology Sydney, Australia. Jaime Valls Miro is also with AZTI Foundation, and this paper is contribution nº 1162 from AZTI, Marine Research, Basque Research and Technology Alliance (BRTA).

(Corresponding author: Yue Wang and Rong Xiong)

wangyue@iipc.zju.edu.cn, rxiong@zju.edu.cn

arbitrary polygonal area are illustrated in Fig. 1. Mathematically, as long as the regions have the same number of interior boundaries, or "holes" [13] (as in the example), a homeomorphism always exists, whereby the template coverage motion in the unit square (exemplified by a back-and-forth motion, shown in the figure) can always be deformed to fit into the target surface. However, solving for the NUC problem is non-trivial because metric distances cannot be preserved by homeomorphic mappings, as illustrated by Fig. 1 (b) and (d). More formally, no homeomorphic mapping can simultaneously preserve both the topological layout of a template coverage path and the uniformity of the resulting coverage path. The problem is further compounded when the surface is non-planar, for instance when seeking a CPP on the surface of an object to be tracked by a manipulator [14]. With non-zero curvature at every surface point, projecting a template coverage path onto a non-planar surface will always render a non-uniform path in the target region. Fig. 2 illustrates this point by projecting various planar coverage solutions onto a saddle grid surface.

Save from simpler objects which can be suitably described by basic analytical expressions (spheres, cylinders ...), more complex 3D surfaces are geometrically modelled by polygonal regular meshes. These are in turn often parameterised (e.g. NURBS) as that makes for more effective model manipulation (infinite scaling, textures, mesh completion, re-meshing, etc), hence the representation choice in the computer graphics or 3D engineering domains. A closer inspection of the NUC problem from a meshing perspective offers insightful observations into the structure of the problem that lends itself to an alternative solver proposition that readily reveals uniform coverage paths for non-planar surfaces. Coverage routes obtained by homeomorphic fitting as described above can be effectively regarded as a set of ordered waypoints drawn on the structured convex polygon selections that make up the polyhedral surfaces, e.g. the grid facet centres in Fig. 2 (left) and (middle). Whilst paths must be non-uniform as subjected to the unavoidable phenomenon that occurs when projecting a planar coverage path onto a non-planar surface, it is however possible to obtain a set of waypoints that lay *uniformly distributed* on a freeform surface [15] when the mesh is unstructured, as shown in Fig. 2 (right) [1]. It is clear that with the uniform but unstructured nature of the surface mesh comes the realisation that the waypoints are unordered, and that the design of a non-repeating visiting sequence of all the waypoints to produce a NUC path remains to be solved.

This work will prove that by said discretisation of the surface into an unstructured mesh, followed by the subdivision of all the mesh facets and explicit utilisation of the local connectivity between adjacent facets on the resulting mesh, a template-free coverage skeleton path can be obtained. With the coverage skeleton uniformly laid on the surface, uncovered surface regions and revisited surface regions are uniformly distributed in the vicinity of the coverage skeleton. A freeform surface representation then makes naive local stochastic optimisation agreeable to fine-tune the coverage skeleton to both augment surface coverage and eliminate gaps as the final NUC solution.

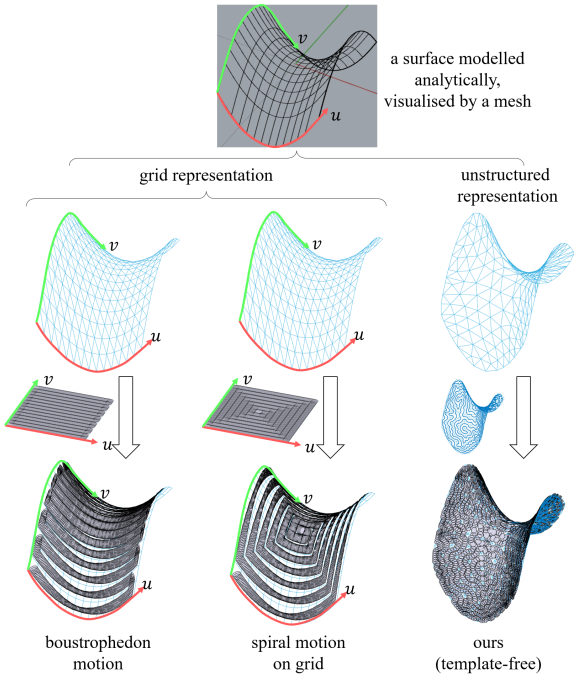


Fig. 2. An illustration of the relationship between the structuredness of the surface mesh representations on a non-planar surface, and the topological layout of coverage paths. Given a structured grid surface, a template boustrophedon (left) and spiral (middle) paths are first designed in a flat square grid, and then fitted onto the target surface through homeomorphism, rendering geometrically non-uniform paths on the surface. By suitably discretising the surface first, a coverage skeleton (not shown) can be effectively generated in-situ for a uniform but unstructured mesh (right).

The novel contributions in this work towards a path planner that constructs a non-revisiting route for uniform coverage motion can thus be summarised as:

1) Reformulating the problem as a planning problem in an unstructured mesh representation.
2) Eliminating topological constraints on the shape of the target surface thus conferring full automation to the coverage task. The scheme relies only on the local connectivity on the surface facets, thus making it suitable to closed surfaces with any number of holes (non-zero genus), and compatible with any prior higher-level cellular decomposition.
3) Providing proof that mesh refinement whereby the facet edges are divided once, is sufficient to guarantee the existence of a non-repetitive sub-facets visiting sequence in the refined mesh.
4) Open sourcing an implementation [2].

The remainder of this paper is organised as follows [3]. Section II reviews existing literature. Section III formally describes the NUC problem, and our motivation to the proposed solution methodology. Section IV describes the proposed

[1] where the centre of each facet is assumed covered by a fixed size footprint representing the tool of interest (painting, blasting, polishing, etc.)

[2] https://github.com/ZJUTongYang/NUC

[3] A video complementing the manuscript with details of the work and additional experimental results can be found in the open-source code repository.

algorithm to generate a guaranteed coverage path for any non-planar surface. Experimental results from simulation and with a real experiment where a NUC path is tracked by a robotic manipulator on an actual object are collected in Section V, with final concluding remarks gathered in Section VI.

II. Related Works

There has been a large body of works focused on the coverage task in planar regions [16], where the mainstream approach is a two-stage process: first dividing the target region into several easy sub-regions, the so-called *cellular decomposition* [17], and then constructing the coverage paths within each sub-region, referred to as *geometric coverage path planning*. Generally, cellular decomposition methods [18] have significantly decreased the difficulty of geometric CPP generation, hence the geometric coverage path in sub-regions can be trivially designed. For example, in trapezoidal sub-regions [19] and rectangular sub-regions [11], the adoption of boustrophedon paths [20] is natural, and in circular cells the spiral coverage is straightforward. For a non-analytical representation of a region, discretising the coverable region into planar grids [11] has been a popular strategy.

Coverage solutions of *non-planar* surfaces are far less mature [21]. Approximating the surface by curvilinear coordinates, [22] proposed an extended scanning curve algorithm. In analogous to algorithms for planar regions, non-planar surfaces are split into easy sub-regions, and a coverage motion is projected from a planar area onto each sub-region. This imposes some challenges: On the one hand, finding a "suitable" surface division during the cellular decomposition processes is non-trivial and may require manual intervention [23]. On the other hand, each sub-region is still non-planar, so the lack of uniformity challenge prevails. Hence, coverage tasks with consideration of non-zero contact tool sizes have mainly been restricted in the literature to near-flat workpieces [24]. In all these approaches, the planar coverage motion was projected onto the target surface, and the inevitable variations in the tool gap width and overlapping derived from the tool motion were neglected.

There is a final family of solutions that do not rely on coverage path patterns, and instead seek uniform coverage curves analytically. Isoparametric line sampling for inspection planning [25] considers each facet on the mesh as an analytical planar region. Exact geodesic distances of every point to a given source point can then be calculated facet-wise [26], and a set of analytical iso-contour curves are constructed on the given mesh [13]. Calculating such iso-contour coverage paths is theoretically sound but also costly. Moreover, the detailing in the discrete mesh representation of an object goes beyond an analytic approximation of the object surface, meaning that any local change in the mesh vertices' location can change the distance fields over the whole mesh substantially. In effect, the results are similar to an analytic (e.g. polynomial) approximation of the surface, with severe limitations when it comes to being effectively adapted to real-world industrial applications with complex objects to manipulate.

In practice, geometric template coverage remains the most widely adopted NUC scheme. More variants keep being proposed, e.g. the recent scheme in [27] with boustrophedon motions adapted to traverse non-planar surfaces. The distance between consecutive parallel segments of the deformed boustrophedon motion is determined waypoint-wise to minimise overlap and missing gaps. There remain no theoretical guarantees for geometric uniformity. And whilst cellular decomposition strategies can partly relieve the problem, the challenge of uniform coverage motion within each cell remains.

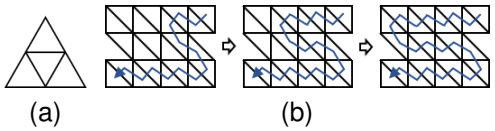


Fig. 3. Illustration of non-revisiting coverage paths. (a) Case where a non-revisiting uniform coverage path does not exist. (b) Illustration of the continuous path deformation process from non-full to full coverage.

III. Problem Statement

Non-revisiting Uniform Coverage. Given a surface and a non-zero size tracing tool, the *non-revisiting uniform coverage* (NUC) task is defined as finding a non-selfcrossing tool path that visits the maximum number of points on the surface, whilst minimising uncovered and overlapped regions.

Uniform Unstructured Mesh and Coverage Skeleton. A non-planar surface can be either non-uniformly described by a structured representation, or uniformly described by an unstructured representation. A process whereby a coarse but uniform coverage skeleton path is first designed onto the surface is proposed, wherein the surface needs to be initially discretised into a mesh as a prime to find a valid coverage skeleton within. However, a noticeable difficulty in generating a coverage skeleton is that it does not always exist on any arbitrary mesh, as seen by the counterexample in Fig. 3a. To overcome this, necessary modifications are proposed for a generic facets coverage solution that is applicable to arbitrary meshes. It is noted that, by definition, any non-revisiting coverage path is always non-selfcrossing, thus by continuously deforming (shrinking) a coverage path, it will always be transformed to the locally shortest path connecting the starting and ending facet. We are inspired to consider its opposite as proof (see Fig. 3b for a visual illustration). Given a trivial initial path, by continuously deforming (stretching) it will be finally transformed into a non-revisiting full coverage path. In this paper we exercise this strategy, on uniform unstructured meshes, for coverage skeleton generation.

IV. NUC Algorithm

The steps to generate the NUC solution for an arbitrary surface are presented in this section. Whilst the coverage skeleton generation (Step 1 and 2) can be applied standalone to coverage tasks on discrete meshes, the full solution is tailored to an analytic representation of the surface (Step 3). Theoretical proofs that a linear-complexity coverage skeleton exists in any sub-divided mesh, with no constraints on the global shape (rectangular, circular, convex ...) or the connectivity of the surface, are also provided therein.

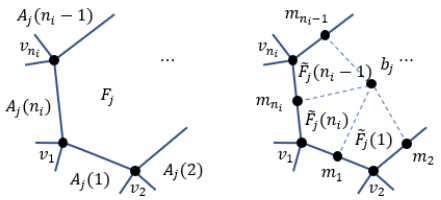


Fig. 4. Mesh refinement: facet partitioning by edge division.

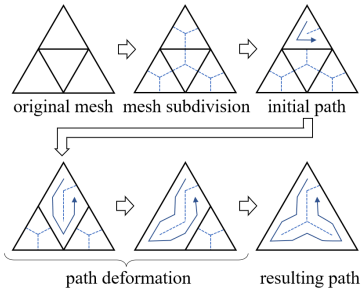


Fig. 5. Flowchart of the proposed non-revisiting uniform coverage solution by edge subdivision. The NUC of the original mesh does not exist. However, the NUC of its subdivision always exists and can be obtained by path deformation.

Step 1: Surface Discretisation

The surface must first be discretised into a mesh. The mesh need not be structured, whilst the uniformity of facets is the focus of this work. This is an established topic (e.g. in computer graphics) where various approaches are permissible, such as resampling-based [28], Voronoi-based [29], and virtual-interaction-force-based [30] methods. A more uniform construction of the discrete mesh certainly improves the uniformity of the solved path with the proposed algorithm. To reveal the main benefit of producing a coverage skeleton, the simplest open-sourced isotropic explicit re-meshing technique [31], available in MeshLab [32], is employed.

The choice of resolution for the unstructured mesh must be such that the generated coverage skeleton is neither too dense (resulting in overlap) nor too sparse (resulting in gaps). Anecdotal evidence gathered from experimentation indicates that the optimised coverage skeletons rarely envelop a surface point more than two times. Hence the recommendation is to select a mesh resolution that allows for the uncovered and overlap rate to play out to be nearly equivalent. As both uncovered and overlap rates are monotonic with respect to mesh resolution (i.e., increasing mesh resolution leads to denser coverage skeleton, lower uncovered rate, and higher overlap rate), the most appropriate resolution can be determined in practice by considering these two rates after a few trials.

Step 2: Coverage Skeleton Construction

Edge Subdivision. See Fig. 4 for a visual illustration of the edge subdivision process. Given any n-edge facet F_j, denoting its vertices as $v_1, \cdots, v_n$ in cyclic order, the subdivision of F_j is a set of n quadrilaterals, constructed in a specific way: Denote the barycentre of facet as b and the midpoint of edge connecting v_i and v_{i+1} as m_i, the facet is divided by curves that connect b_j and $v_1, \cdots, v_n$, where for convex facets the curves can be straight line segments. The i-th sub-facet is defined as the quadrilateral:

$$[b_j, m_i, v_{i+1}, m_{i+1}], i = 1, \cdots, n \tag{1}$$

where the indices are in cyclic order, hence $v_{n+1} = v_1$ and $m_{n+1} = m_1$. And the bracket $[\cdot]$ means that the quadrilaterals have the same orientation as F_j. By the subdivision process, each edge of the original mesh is divided into two sub-edges. It should be noted that valid edge subdivision processes are not unique.

Coverage Skeleton Generation. This module aims to non-repetitively visit all sub-facets of the refined mesh. Two adjacent facets F_i (with n_i edges) and F_j (with n_j edges) are taken here as an example.

First, the set of adjacent facets of each facet F_i is collected as A_i in counter-clockwise order. Hence the index of F_j in A_i and the index of F_i in A_j are known, denoted as $\mathrm{loc1}$ and $\mathrm{loc2}$ respectively. Then, the set of sub-facets of all facets following the sub-division steps shown above have been collected as

$$\begin{aligned} \hat{F}_i &= \{\hat{F}_i(1), \cdots, \hat{F}_i(n_i)\} \\ \hat{F}_j &= \{\hat{F}_j(1), \cdots, \hat{F}_j(n_j)\} \end{aligned} \tag{2}$$

where the connectivity between F_i and F_j is now replaced by the adjacency of two pairs of sub-facets $\{\hat{F}_i(\mathrm{loc1}-1), \hat{F}_j(\mathrm{loc2})\}$ and $\{\hat{F}_i(\mathrm{loc1}), \hat{F}_j(\mathrm{loc2}-1)\}$. Here all indices are referred cyclically, as such $\hat{F}_i(0)$ and $\hat{F}_j(0)$ are actually $\hat{F}_i(n_i)$ and $\hat{F}_j(n_j)$, respectively.

Then, starting from an initial trivial path $\hat{R} = \hat{F}_1$, adjacent facets are incrementally incorporated into the path by continuous path deformation. Let facet i have been covered, and the indices $\mathrm{loc1}$ and $\mathrm{loc2}$ are defined the same as before. The cyclic order of $\hat{F}_j$ is adjusted such that it starts from $\mathrm{loc2}$ and ends at $\mathrm{loc2}-1$,

$$\begin{aligned} &[\hat{F}_j(1), \cdots, \hat{F}_j(\mathrm{loc2}-1), \hat{F}_j(\mathrm{loc2}), \cdots, \hat{F}_j(n_j)] \\ \rightsquigarrow &[\hat{F}_j(\mathrm{loc2}), \cdots, \hat{F}_j(n_j), \hat{F}_j(1), \cdots, \hat{F}_j(\mathrm{loc2}-1)] \end{aligned} \tag{3}$$

and insert it between $\hat{F}_i(\mathrm{loc1}-1)$ and $\hat{F}_i(\mathrm{loc1})$ in $\hat{R}$,

$$\begin{aligned} &[\hat{R}(1), \cdots, \hat{F}_i(\mathrm{loc1}-1), \hat{F}_i(\mathrm{loc1}), \cdots, \hat{R}(\mathrm{end})] \\ \rightsquigarrow &[\hat{R}(1), \cdots, \hat{F}_i(\mathrm{loc1}-1), \\ &\quad \hat{F}_j(\mathrm{loc2}), \cdots, \hat{F}_j(\mathrm{end}), \hat{F}_j(1), \cdots, \hat{F}_j(\mathrm{loc2}-1), \\ &\quad \hat{F}_i(\mathrm{loc1}), \cdots, \hat{R}(\mathrm{end})] \end{aligned} \tag{4}$$

Finally, the tool motion is constructed by collecting the centres of sub-facets in $\hat{R}$ in order. A step-by-step illustration of the construction of the NUC resulting path is shown in Fig. 5. Differing strategies in selecting the next adjacent facet to be incorporated will lead to distinct coverage skeletons which will have slightly different coverage performances, whilst guaranteeing coverage of all sub-facets. In this work, the skeleton is consistently generated by manipulating the first facet in the list of uncovered adjacent facets.

Theorem 1. *Any mesh, after being edge sub-divided once, must admit a coverage skeleton that enables non-repetitive and full coverage of all sub-facets, provided that the facet containing the starting sub-facet and the ending sub-facet is*

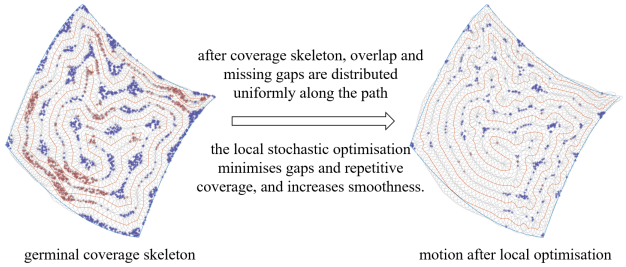


Fig. 6. Local stochastic optimisation.

chosen such that other facets still form a connected mesh. (A simple counterexample is provided in Fig. 7.)

Proof. Since the sub-facet sequence will not expand into a facet that has been visited, non-repetitiveness is guaranteed. Hence only the full coverage property needs proof. It is proven by contradiction.

Assume that a facet is left uncovered after Step 2 has finished, whose index is j. Since the mesh is connected, the uncovered facet must be adjacent to some covered facets. Let the covered facet that facet j is adjacent be indexed as i. Then, they have satisfied the conditions of the above-mentioned path deformation process. Thus a contradiction has been reached because the coverage skeleton generation process has not yet finished: the path that covers facet i can still be further deformed to cover facet j. □

Theorem 2. *The coverage skeleton generation is of linear complexity.*

Proof. **Assumptions.** We assume the most commonly used setting, whereby the unstructured mesh is stored as a facet-vertex mesh, and for simplicity the mesh is assumed triangular.

Facet Adjacency. To determine the adjacency between facets, the adjacency matrix A is leveraged. Let A be a full-zero matrix at the beginning. The entry $A(i, j)$ is set to k if facet k has vertex i and j in order. After processing all ordered pairs of vertices from the facet list, facet $A(i, j)$ and $A(j, i)$ (for any arbitrary i and j) are identified as adjacent facets, provided that they are both non-zero. This process is performed in linear time complexity.

Coverage Skeleton Generation. The coverage skeleton is generated by managing uncovered facets using a stack. When facet i is newly covered, its adjacent facets–say j, k, l–are pushed into the stack with the adjacency to i, in the form of (i, j), (i, k), and (i, l). In the next iteration, an element is popped from the stack. If the facet has already been covered, facets are continuously popped until an uncovered facet is encountered, when continuous path deformation is applied to cover it. Since each facet is manipulated only once during the iterative loop, the process is also linear in time complexity, □

Step 3: Stochastic Optimisation

After the coverage skeleton is generated in the refined mesh, each waypoint is replaced by the closet surface point, thus mapping the coverage skeleton back onto the original surface. In this way, the coverage skeleton becomes the germinal uniform coverage motion on the surface. The path is further optimised by taking advantage of the parametric representation of the surface (a spline surface is used in this work), whereby each waypoint is randomly adjusted in the direction perpendicular to the path. Given the coverage path to be optimised, a tool waypoint is randomly selected, and a random orientation in the U-V coordinate is chosen for the movement of the waypoint, along with a random movement distance (limited to a maximum of 0.1 times the tool size to ensure that the optimisation is local). If the coverage performance is improved by the revised waypoint locale, the modification is accepted. Otherwise, it is discarded. This stochastic optimisation does not fundamentally affect the global layout provided by the initial coverage skeleton, with the overall path length remaining akin. See Fig. 6 for illustration. The stochastic optimisation process loops in an any-time manner, improving or remaining as is at each step.

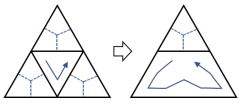


Fig. 7. Coverage skeleton nonexistence derived from assignment of first facet.

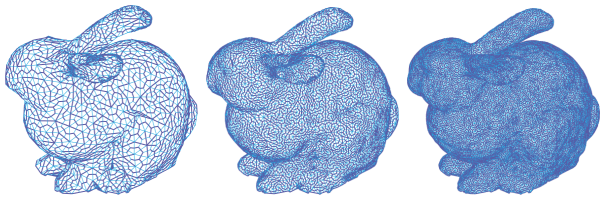


Fig. 8. Coverage skeleton generated on the Stanford bunny mesh with different discretisation resolutions.

V. Simulated and Real-World Experiments

The proposed algorithm has been tested on a range of benchmark 3D surfaces—a saddle surface with varying curvatures, the Standford bunny, and a wok on a real experiment. Some preliminary definitions of the metrics used in the experimental comparisons are first provided. A supplementary video with further details of the simulations and a real-world implementation of a coverage task with a robotic manipulator is also supplied here: https://github.com/ZJUTongYang/NUC

A. Experimental Settings

Tool Contact Model. An equivalent tool-surface contact model to [27], briefly re-stated here, is adopted for this work. See Fig. 10. The target surface curvature is assumed not inordinately steep with respect to the tool radius, so that the contact area can be represented as a simply-connected region. The tool position is a point on the surface, and the tool orientation is perpendicular to the surface tangent plane. The maximal allowable intrusion distance of the tool to the surface is set as d. The constraint that the coverable region for a given tool pose is approximated by an ellipse, parabola, or hyperbola [27] is relaxed in this work, expanding to a target surface of non-constant curvature anywhere.

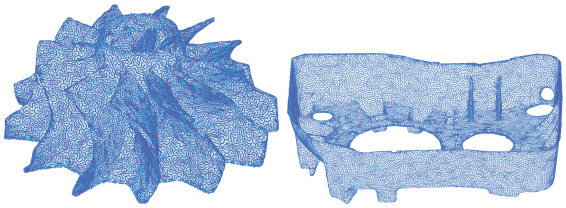


Fig. 9. Coverage skeleton on a turbine (left) and a gearbox object (right).

TABLE I
NUC PERFORMANCE ON TESTING OBJECTS

Object	Facets	Re-meshing[1]	Subdivision	CPP
Bunny (sparse)	1760	3.25s	72.78ms	57.00ms
Bunny (middle)	4934	3.23s	277.20ms	276.70ms
Bunny (dense)	12475	6.88s	958.34ms	1.21s
Turbine	17977	55.09s	1.61s	2.23s
Gearbox	21301	43.95s	2.05s	3.14s

[1] Re-meshing step carried out prior in Meshlab [32].

Coverage Rate Calculation. To calculate precise coverage rates, a sampling-based approach has been adopted: For each algorithm, the resultant path is represented by a sequence of dense waypoints (>2000 in this work). A large number of uniformly distributed surface sample points (10^4 in this work) are collected by rejection sampling. The set of tool poses' indices that can cover each surface sample point is then computed. An empty set denotes an uncovered surface point, whilst a continuous sequence of indices denotes a non-repetitively covered surface point. If the set forms n continuous sequences of indices, the surface point is identified as repetitively covered n times. The rates of uncovered (U.Rate), covered (C.Rate), and repetitively covered surface points (O.Rate) are estimated as the proportion of their respective sets.

Performance Metrics. To evaluate how repetitively the surface points are covered, a weighted *Cost* of each coverage motion is proposed: Let there be N surface points, and the i-th point is covered by n_i times, then

$$Cost = \sum_{i=1}^{N} \frac{|n_i - 1|}{N} \tag{5}$$

Using this weighted cost, a point being repetitively covered two times weighs the same as a point being uncovered, and a point being repetitively covered by $n(>1)$ times weighs the same as $(n-1)$ points being uncovered.

B. Coverage Skeleton Complexity

The coverage skeleton has been applied to arbitrary meshes to examine the computational time of the coverage skeleton generation phase. Five different meshes are utilised for the test, including the Standford bunny's surface with three different resolutions, a turbine object, and a gearbox object, as shown in Fig. 8 and Fig. 9. The number of facets in the meshes and the computational time for re-meshing, sub-division, and coverage skeleton generation (CPP) are listed in Table. I. These results indicate that the computational bottleneck for the proposed NUC scheme is strongly associated to the re-meshing step, whilst the proposed algorithm runs efficiently in comparison.

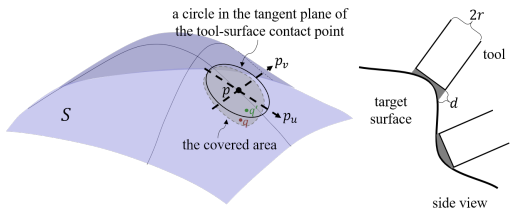


Fig. 10. Illustration of the covered surface area of a given tool pose. Given the tool-surface contact point denoted as p where the tangent vectors are p_u and p_v, the projection of a surface point q onto the tangent plane is assumed as q'. q can be covered only if the distance between q and q' is less than d, and the distance between q' to p is less than r, the tool radius. Here d and r are given thresholds.

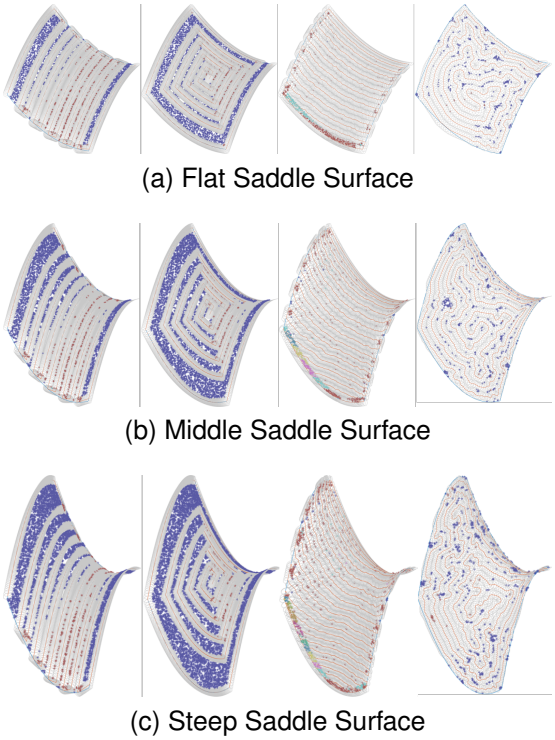


Fig. 11. Coverage of three different saddle surfaces using different coverage algorithms. From left to right: boustrophedon motion, spiral motion, [27] and the proposed algorithm. Uncovered surface points are marked in blue. Points with perfect coverage (being covered for exactly one time) are hidden. Revisited points are shown in other colours.

C. Comparisons on Saddle Surfaces

The performance of the algorithm is compared in detail with various coverage path planning algorithms by extensive testing on a classic non-planar object, a saddle-shaped surface. See Fig. V-B for illustration and Table. II for the related statistics.

Performance of Boustrophedon Motion. The boustrophedon coverage motion has been designed on the curvilinear coordinate of a spline surface. Given the shape of the tool, the number of back-and-forth motions in the boustrophedon coverage has been carefully set such that repetitive coverage is avoided at the centre of the saddle surface. Although the paths are uniform in the parametric curvilinear coordinates, they end up non-uniformly laid on the surface. Moreover, the sharp turnings during the motion cause repetitive coverage. The performance of the boustrophedon motion is clearly not

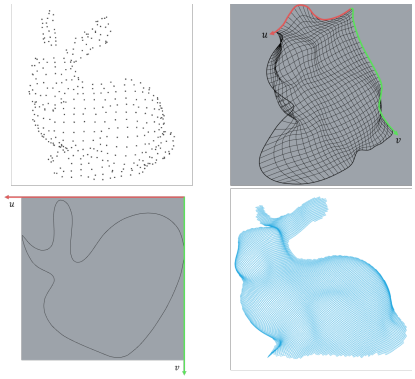


Fig. 12. A benchmark example using the Stanford bunny: from an initial point cloud representation (top left), a parameterised B-spline surface is obtained (top right). The closed curve in U-V coordinates (bottom left) captures the internal sub-region of interest from the data. The resulting trimmed parametric surface is shown on the bottom right.

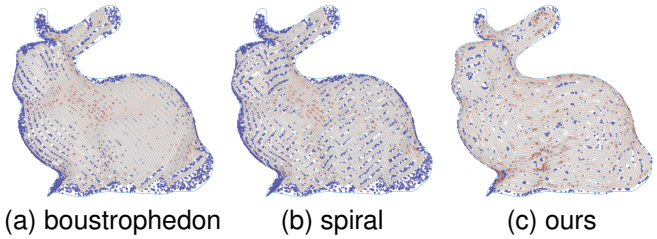


Fig. 13. Coverage of the frontal part of the Stanford bunny.

competitive against other algorithms.

Performance of Spiral Motion. Following the same vein, the spiral motion is also designed on the surfaces. The number of cycles in the spiral motion has been chosen such that repetitive coverage is avoided, which however leads to large areas being uncovered. The coverage rate of the spiral motion for the flat, middle, and steep saddle surfaces are 72.47%, 61.33%, and 55.10%, respectively.

Performance of [27]. As per our analysis, the uneven surface is intrinsically not a rectangular region, thus the boustrophedon motion must revisit or miss some parts on the surface, no matter how it is deformed. Now that [27] prefers non-revisiting coverage at the beginning part of the motion, severe overlaps have to appear at the final segment of the motion. See Fig. V-B for illustration. With the surface being increasingly curved, although a large proportion of the surface (except the final segment of the coverage motion) is still covered perfectly, there exist surface points that are covered for 3, 6, and 10 times, respectively, resulting in deteriorating $Cost$ performance (0.067, 0.109, and 0.165, respectively).

Performance of Ours. Using the proposed algorithm, the surface is re-meshed into an unstructured mesh with appropriate resolution as per the description in Section IV Step 1, where the coverage skeleton is generated. Stochastic optimisation is then adopted to optimise the coverage performance. See Fig. V-B for the illustrations of the proposed algorithm. Results show that the proposed algorithm establishes the highest coverage (single) rate among all existing algorithms in all testing cases, and also obtains the best performance under the weighted $Cost$ metric.

TABLE II
PERFORMANCE OF VARIOUS COVERAGE PATHS

Methods	Surface	U. Rate	C. Rate	O. Rate	Max.	$Cost$	L.	T.
boust.	SF	14.56%	80.94%	4.50%	3	0.192	140.15	51.24
spiral		25.38%	72.47%	2.15%	**2**	0.275	126.55	54.60
[27]		**0.01%**	94.70%	5.29%	3	0.067	178.73	285.46
ours		3.99%	**95.81%**	**0.20%**	**2**	**0.042**	134.60	185.97
boust.	SM	23.44%	73.39%	3.17%	3	0.267	185.40	58.43
spiral		37.51%	61.33%	1.16%	**2**	0.387	158.69	58.20
[27]		**0.18%**	94.36%	5.46%	6	0.109	285.49	411.48
ours		3.97%	**95.90%**	**0.13%**	**2**	**0.041**	197.63	290.85
boust.	SS	28.28%	68.67%	3.50%	3	0.313	239.28	61.95
spiral		44.17%	55.10%	0.82%	**2**	0.450	198.14	59.62
[27]		**0.23%**	92.82%	6.95%	10	0.165	428.62	528.64
ours		5.15%	**94.52%**	**0.33%**	**2**	**0.055**	272.80	420.65
boust	FB	29.27%	68.16%	2.57%	**3**	0.319	385.74	293.34
spiral		34.18%	63.50%	**2.32%**	**3**	0.365	357.41	506.93
ours		**11.59%**	**82.69%**	5.72%	**3**	**0.174**	492.98	829.06

SF: saddle(flat), SM: saddle(middle), SS: saddle(steep), FB: frontal bunny
U. Rate: Uncovered rate, C. Rate: Covered(single) rate, O. Rate: Overlapped rate
Max. : The maximum number of visiting of a surface point
L.: Coverage path length (unit: cm)
T.: The sum of 3D orientation turnings of the tool trajectory (unit: rad)

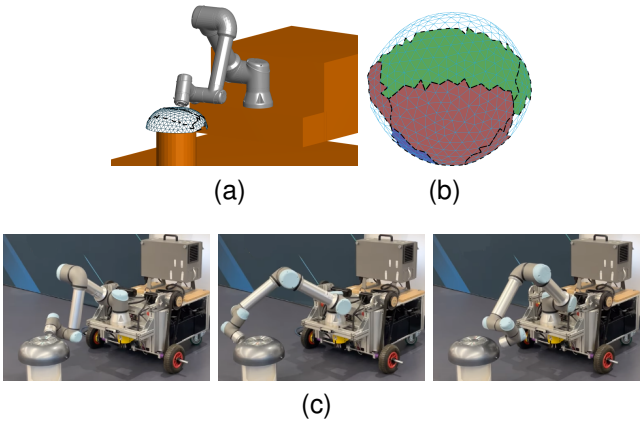


Fig. 14. Illustration of the real-world experiment. (a) The simulated manipulator scene. (b) The coloured topological graph generated by [14]. (c) Video stills of each continuous coverage motion segment.

D. Comparisons on Bunny Surface

The algorithms are also applied to the Stanford bunny's frontal surface. A trimmed B-spline surface is fitted to the bunny's mesh using Point Cloud Library (PCL) [33]. See Fig. 12 for illustration. The parametric subregion that the trimmed surface occupies is concave, hence Wen's method [27] is not applicable. See Fig. 13 for the illustrations of the boustrophedon motion, spiral motion, and the proposed algorithm on the bunny's frontal surface. Quantified results are also collected in Table. II. Results show that the proposed algorithm obtains the least uncovered rate, the highest covered (single) rate, and the best $Cost$ metric.

E. Real World Experiments

A real-world NUC task is illustrated to show the proposed algorithm in action on a real robot, motivated by the inspection of a cooking utensil (a wok) with a robot arm (a fixed-based UR5). Extracts are depicted in Fig. 14. The tool's locations are set above the surface by 2cm, with end effector

poses constrained to remain perpendicular to the surface's tangent plane at all times. As such, the tool poses are five dimensional, since the tool can rotate freely around the normal vector of the surface. Thus, in the experiments we lock the last joint in the UR5, so that the coverage problem remains within a non-redundant setting. The kinematics of the UR5 satisfies the Pieper solubility condition, which allows us to calculate analytically all possible inverse kinematic configurations. Due to the non-linear manipulator kinematics and collision avoidance, a simple spiral motion generated on the wok surface can not be continuously tracked. Thus, a cellular decomposition algorithm [14] has been used to decompose the surface into three sub-regions (Fig. 14(b)), each of which is proven to be continuously coverable by the manipulator (some representative configurations are shown in Fig. 14(c)). The NUC solution was then generated by the proposed algorithm in each sub-region. Two reconfiguration motions are required to concatenate the three coverage motion segments into a single one, which is accomplished with an RRT planner. The joint-space coverage path is interpolated cubically into a smooth and constant-speed trajectory and executed by a controllers (C++ in ROS). The reader is referred to the supplementary video for the full motion including manipulator reconfigurations.

VI. Conclusion

A novel construction for the coverage planning of any given non-planar surface has been proposed in this work. The non-revisiting uniform coverage (NUC) path of the surface is equivalent to visiting a set of non-repeated, uniformly sampled waypoints which must be unstructurally laid on the surface. This research has considered the coverage path generation by letting a coverage skeleton be constructed from an unstructured mesh which uniformly represents the target surface area. The existence of a coverage skeleton on the edge-subdivided polygonal mesh of any arbitrarily-shaped object surface has been proven, and a practical algorithm has been proposed to find a valid NUC resulting path. This makes the proposed scheme prominently applicable in coverage tasks under non-linear manipulator constraints. Extensive illustrations and a real-world implementation of a manipulator coverage task have been presented to show the validity of the proposed algorithm. These have been supplemented by a detailed video and an open source implementation in C++ for the benefit of the research community.

References

[1] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robot. and Auton. Syst.*, vol. 61, no. 12, pp. 1258–1276, 2013.

[2] E. Eiben, I. Godage, I. Kanj, and G. Xia, "On the problem of covering a 3-d terrain," in *AAAI*, vol. 34, no. 06, 2020, pp. 10 361–10 368.

[3] F. Balampanis, I. Maza, and A. Ollero, "Spiral-like coverage path planning for multiple heterogeneous uas operating in coastal regions," in *ICUAS*, 2017, pp. 617–624.

[4] K. O. Ellefsen, H. A. Lepikson, and J. C. Albiez, "Multiobjective coverage path planning: Enabling automated inspection of complex, real-world structures," *Appl. Soft Comput.*, vol. 61, pp. 264–282, 2017.

[5] T. Oksanen and A. Visala, "Coverage path planning algorithms for agricultural field machines," *J. Field Robot.*, vol. 26, no. 8, pp. 651–668, 2009.

[6] P. N. Atkar, A. Greenfield, D. C. Conner, H. Choset, and A. A. Rizzi, "Uniform coverage of automotive surface patches," *Int. J. Robot. Res.*, vol. 24, no. 11, pp. 883–898, 2005.

[7] H. Chen, T. Fuhlbrigge, and X. Li, "Automated industrial robot path planning for spray painting process: a review," in *CASE*, 2008, pp. 522–527.

[8] Y. Liu, W. Zhao, R. Sun, and X. Yue, "Optimal path planning for automated dimensional inspection of free-form surfaces," *J. Manuf. Syst.*, vol. 56, pp. 84–92, 2020.

[9] M. Hassan and D. Liu, "A deformable spiral based algorithm to smooth coverage path planning for marine growth removal," in *IROS*, 2018, pp. 1913–1918.

[10] T. A. Driscoll and L. N. Trefethen, *Schwarz-christoffel mapping*. Cambridge University Press, 2002, vol. 8.

[11] Y. Gabriely and E. Rimon, "Spiral-stc: An on-line coverage algorithm of grid environments by a mobile robot," in *ICRA*, vol. 1, 2002, pp. 954–960.

[12] Y.-H. Choi, T.-K. Lee, S.-H. Baek, and S.-Y. Oh, "Online complete coverage path planning for mobile robots based on linked spiral paths using constrained inverse distance transform," in *IROS*, 2009, pp. 5788–5793.

[13] C. Wu, C. Dai, X. Gong, Y.-J. Liu, J. Wang, X. D. Gu, and C. C. Wang, "Energy-efficient coverage path planning for general terrain surfaces," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2584–2591, 2019.

[14] T. Yang, J. Valls Miro, Q. Lai, Y. Wang, and R. Xiong, "Cellular decomposition for non-repetitive coverage task with minimum discontinuities," *IEEE/ASME T. Mech.*, pp. 1698–1708, 2020.

[15] S. M. Obeidat and S. Raman, "An intelligent sampling method for inspecting free-form surfaces," *Int. J. Adv. Manuf. Tech.*, vol. 40, no. 11, pp. 1125–1136, 2009.

[16] S. C. Wong and B. A. MacDonald, "A topological coverage algorithm for mobile robots," in *IROS*, vol. 2, 2003, pp. 1685–1690.

[17] E. U. Acar, H. Choset, A. A. Rizzi, P. N. Atkar, and D. Hull, "Morse decompositions for coverage tasks," *Int. J. Robot. Res.*, vol. 21, no. 4, pp. 331–344, 2002.

[18] R. Mannadiar and I. Rekleitis, "Optimal coverage of a known arbitrary environment," in *ICRA*, 2010, pp. 5525–5530.

[19] J.-C. Latombe, *Robot motion planning*. Springer Science & Business Media, 2012, vol. 124.

[20] H. Choset, "Coverage of known spaces: The boustrophedon cellular decomposition," *Auton. Robot.*, vol. 9, no. 3, pp. 247–253, 2000.

[21] S. Huo, D. Navarro-Alarcon, and D. T. Chik, "A robotic defect inspection system for free-form specular surfaces," in *ICRA*, 2021, pp. 11 364–11 370.

[22] H.-y. Tam, "Toward the uniform coverage of surfaces by scanning curves," *Comput. Aided. Des.*, vol. 31, no. 9, pp. 585–596, 1999.

[23] Z. Zhou, Y. Zhang, and K. Tang, "Sweep scan path planning for efficient freeform surface inspection on five-axis cmm," *Comput. Aided. Des.*, vol. 77, pp. 1–17, 2016.

[24] P. M. Bhatt, C. Gong, A. M. Kabir, R. K. Malhan, B. C. Shah, and S. K. Gupta, "Incorporating tool contact considerations in tool-path planning for robotic operations," in *MSEC*, vol. 1, 2020.

[25] D. F. ElKott and S. C. Veldhuis, "Isoparametric line sampling for the inspection planning of sculptured surfaces," *Comput. Aided. Des.*, vol. 37, no. 2, pp. 189–200, 2005.

[26] Y.-J. Liu, Z. Chen, and K. Tang, "Construction of iso-contours, bisectors, and voronoi diagrams on triangulated surfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1502–1517, 2010.

[27] Y. Wen, D. J. Jaeger, and P. R. Pagilla, "Uniform coverage tool path generation for robotic surface finishing of curved surfaces," *IEEE Robot. Autom. Lett*, vol. 7, no. 2, pp. 4931–4938, 2022.

[28] C. Lv, W. Lin, and B. Zhao, "Intrinsic and isotropic resampling for 3d point clouds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3274–3291, 2022.

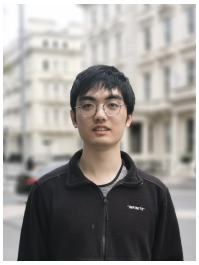
[29] S. Valette and J.-M. Chassery, "Approximated centroidal voronoi diagrams for uniform polygonal mesh coarsening," in *Comput. Graph. Forum*, vol. 23, no. 3. Wiley Online Library, 2004, pp. 381–389.

[30] Q. Zeng, S. Zhu, Z. Li, and X. Guo, "Self-adaptive triangular mesh generation framework for free-form single-layer reticulated shells based on virtual interaction forces," *Autom. Constr.*, vol. 148, p. 104750, 2023.

[31] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Mesh optimization," in *SIGGRAPH*, 1993, p. 19–26.

[32] P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, G. Ranzuglia *et al.*, "Meshlab: an open-source mesh processing tool," in *Eurographics Italian Chapter Conference*, vol. 2008, 2008, pp. 129–136.

[33] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *ICRA*, May 9-13 2011.



Tong Yang (Member, IEEE) is a Ph.D. candidate in the Robotics Laboratory at Zhejiang University, where he received his B.Eng. degree in Control Science and Engineering in 2018. He is a visiting researcher at the Centre for Autonomous Systems at the University of Technology Sydney in 2019 and at the Personal Robotics Laboratory at Imperial College London in 2023. His research interests include planning problems for mobile robots, tethered robots, and manipulators, as well as their mathematical underpinnings such as algebraic topology, graph theory, and differential geometry.

Jaime Valls Miro (Member, IEEE) received his B.Eng. and M.Eng. degrees in Computer Science (Systems Engineering) from the Valencia Polytechnic University (Spain), and the Ph.D. degree in robotics and control systems from Middlesex University (UK). He is currently an Associate Professor at the Robotics Institute, University of Technology Sydney (Australia), and a Research Professor with the AZTI Foundation (Bilbao, Spain). His areas of interest span across the field of robotics, most notably modelling sensor behaviours for perception and mapping, computational intelligence in HRI - assistive robotics in particular, and robot navigation.

Minh Nguyen received his B.Eng. in Mechatronic Engineering from the University of Technology Sydney (Australia). He is currently a Research Assistant at the Robotics Institute, University of Technology Sydney (Australia). His area of interest is robotics motion and planning.

Yue Wang (Member, IEEE) received his Ph.D degree in Control Science and Engineering from Department of Control Science and Engineering, Zhejiang University, Hangzhou, P.R. China in 2016. He is currently an Associate Professor in the Department of Control Science and Engineering, Zhejiang University, Hangzhou, P.R. China. His latest research interests include mobile robotics and robot perception.

Rong Xiong (Member, IEEE) received her Ph.D degree in Control Science and Engineering from the Department of Control Science and Engineering, Zhejiang University, Hangzhou, P.R. China in 2009. She is currently a Professor in the Department of Control Science and Engineering, Zhejiang University, Hangzhou, P.R. China. Her latest research interests include motion planning and SLAM.