

Article

Robot Navigation in Complex Workspaces Employing Harmonic Maps and Adaptive Artificial Potential Fields

Panagiotis Vlantis ^{1,*}, Charalampos P. Bechlioulis ^{1,*}  and Kostas J. Kyriakopoulos ² 

¹ Department of Electrical and Computer Engineering, University of Patras, 26504 Patras, Greece; panvlantis@outlook.com

² School of Mechanical Engineering, National Technical University of Athens, 15780 Athens, Greece; kkyria@mail.ntua.gr

* Correspondence: chmpechl@upatras.gr

Abstract: In this work, we address the single robot navigation problem within a planar and arbitrarily connected workspace. In particular, we present an algorithm that transforms any static, compact, planar workspace of arbitrary connectedness and shape to a disk, where the navigation problem can be easily solved. Our solution benefits from the fact that it only requires a fine representation of the workspace boundary (i.e., a set of points), which is easily obtained in practice via SLAM. The proposed transformation, combined with a workspace decomposition strategy that reduces the computational complexity, has been exhaustively tested and has shown excellent performance in complex workspaces. A motion control scheme is also provided for the class of non-holonomic robots with unicycle kinematics, which are commonly used in most industrial applications. Moreover, the tuning of the underlying control parameters is rather straightforward as it affects only the shape of the resulted trajectories and not the critical specifications of collision avoidance and convergence to the goal position. Finally, we validate the efficacy of the proposed navigation strategy via extensive simulations and experimental studies.

Keywords: motion and path planning; collision avoidance; autonomous vehicle navigation; artificial potential fields



Citation: Vlantis, P.; Bechlioulis, C.P.; Kyriakopoulos, K.J. Robot Navigation in Complex Workspaces Employing Harmonic Maps and Adaptive Artificial Potential Fields. *Sensors* **2023**, *23*, 4464. <https://doi.org/10.3390/s23094464>

Academic Editors: Stephen Monk and David Cheneler

Received: 17 April 2023

Revised: 29 April 2023

Accepted: 2 May 2023

Published: 3 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The navigation of autonomous robots in cluttered environments is a widely studied topic in the field of robotics. Popular methodologies that have been employed in the related literature to address it include, but are not limited to, *configuration space decomposition approaches* [1,2]; *probabilistic sampling methods* such as rapidly exploring random trees [3,4], probabilistic roadmaps [5,6] and manifold samples [7,8]; and *optimal control strategies* such as receding horizon control [9,10] and path homotopy invariants [11,12]. Apart from the aforementioned discrete methods regarding the workspace and/or the decision domain, Artificial Potential Fields (APFs) that were originally introduced in [13] generally provide a simpler means of encoding collision avoidance specifications, with their negated gradient functioning as a reference motion direction that drives the robot towards the desired goal configuration. As shown in [14], despite their intuitive nature, this class of controllers suffers unavoidably from the presence of unwanted equilibria induced by the workspace's topology, whose region of attraction may not be trivial. In their seminal work [15], Rimon and Koditschek presented a family of APFs called Navigation Functions (NFs) for point and sphere worlds, as well as a constructive transformation for mapping workspaces cluttered by sequences of star-shaped obstacles into such worlds. However, certain design parameters require tedious tuning to eliminate unwanted local minima and render the transformation a diffeomorphism. In practice, this solution suffers by the fact that the allowable values of the design parameters may cause both the potential and the corresponding transformation to vary too abruptly close to the obstacles (the issue

of “disappearing valleys” [15]), thus pushing the trajectories of the robot very close to them. Density functions for remedying such drawbacks or adjustable NFs for relaxing some generally conservative requirements are presented in [16,17]. Additionally, attempts to extend the NF framework directly to non-sphere worlds can be found in [18,19]. Finally, a novel approach based on power diagrams which can be used for designing tune-free vector fields for navigation within convex workspaces is also presented in [20].

Artificial Harmonic Potential Fields (AHPFs) constitute an interesting subclass of APFs, since they are free of unwanted local minima by construction. However, no simple method exists for constructing safe (with respect to obstacle avoidance), harmonic potentials even for simple workspaces. AHPFs suitable for navigation in realistic environments were originally utilized in [21], where computationally expensive numerical techniques were employed to solve the associated Dirichlet and Neumann problems. Several extensions of the aforementioned methodology followed [22,23], addressing issues such as numerical precision and computation, dynamic environments, etc. The panel method was also employed in [24–26] to build harmonic potentials to coordinate the motion of single and multiple robots in polygonal environments. In [27,28], well-known closed-form solutions of the uncompressed fluid flow around simple geometries was used in order to safely drive a robot among moving obstacles. Harmonic potential fields have also been used in [29,30] to address the Simultaneous Localization and Mapping problem (SLAM) by coordinating the robot motion in unknown environments. Moreover, a methodology based on the evaluation of the harmonic potential field’s streamlines was used in [31,32] for mapping a multiply connected workspace to a disk, collapsing inner obstacles to line segments or arcs. In a recent work [33], the problem of designing closed form harmonic potentials in sphere worlds was addressed by the introduction of a diffeomorphism [34], which allows mapping such workspaces to the euclidean plane with some of its points removed. Finally, extensions of this work addressing topologically complex three-dimensional workspaces or multi-robot scenarios by introducing appropriate constructive workspace transformations can be found in [35,36], respectively.

1.1. Contributions

We address the navigation problem for a robot operating within a static, compact, planar workspace of arbitrary connectedness and shape by designing a control law that safely drives the robot to a given goal position from almost any initial feasible configuration. The goal of this work is twofold. **(A) To cope with the topological complexity of the workspace**, we employed numerical techniques in order to build a transformation that maps the workspace onto a punctured disk and delved into the respective construction in detail. We remark that, although the transformation constructed using this method is an approximation of a harmonic map ideal for navigation, our solution benefits from the fact that it only needs a sufficiently fine polygonal workspace description that can be easily acquired in practice (e.g., through SLAM), contrary to [15,34,36] that require an explicit representation of the workspace boundaries (i.e., as the level sets of sufficiently smooth functions). Moreover, unlike the solutions proposed in [15,36], our approach does not require the decomposition of the workspace obstacles into sequences of simpler overlapping shapes and computes the desired transformation in one step. **(B) To steer the robot to its desired configuration**, we employed a control law based on closed-form AHPFs coupled with adaptive laws for their parameters to eliminate the necessity of explicitly defined local activation neighborhoods around the workspace boundaries for ensuring collision avoidance. Our approach is reactive (closed loop) since it selects the velocity of the robot based on the positions of the robot, the desired goal and the workspace boundary. As such, it is more robust against position measurement errors than other open loop approaches such as configuration space decomposition approaches [2] or probabilistic sampling methods such as rapidly exploring random trees [4], probabilistic roadmaps [6] and manifold samples [8], where an open loop path is initially extracted and executed by a trajectory tracking controller. In this way, even small position errors risk the safe

execution of the calculated plan. We remark that our overall control scheme only requires solving a computationally expensive problem once for a given static workspace, independent of the robot's initial and goal configurations, in contrast to the solutions presented in [21,22]. Finally, we adapt our methodology to the class of differential drive robots, which are commonly encountered in real-world applications and propose an algorithm that decomposes the overall workspace into small neighbouring subsets to render the problem of addressing large workspaces tractable. An overview of the proposed methodology's pros and cons compared to alternative transformations and potential fields can be seen in Tables 1 and 2, respectively.

Preliminary results were included in our conference paper [37]. We have to stress though that the algorithmic calculation of the harmonic map is given in the present work, along with a rigorous formulation of the panel method. A modification of the adaptive laws for the parameters of the underlying potential field is also introduced to simplify the tuning process by eliminating the necessity of heuristically defined local activation neighborhoods around the workspace boundaries for ensuring collision avoidance. Moreover, an extension for tackling the navigation problem under unicycle kinematics is also provided. Finally, new comparative simulation results are provided to highlight the strong points of the proposed method with respect to other related works, accompanied by an experiment employing an actual robot navigating within a complex office workspace.

Table 1. Comparison between the Harmonic Transformation (HM) proposed in this work and the (i) Star-to-Sphere Transformation (SST) [15], (ii) Multi-Agent Navigation Transformation (MANT) [36] and (iii) the Navigation Transformation (NT) [34]. Although HMs require global knowledge of the workspace's geometry to be constructed, HMs are infinitely differentiable and require the domain to be represented by closed polygonal curves (which can be easily obtained using SLAM methodologies), unlike the alternatives that require the domain boundaries to be represented as sets of sufficiently differentiable implicit equations.

	Geometry Representation	Global	Analytic
HM	Points on the boundary	Yes	Yes
SST	Trees of Stars	Yes	Yes
NT	C^2 -manifolds	No	No
MANT	Trees of C^2 -manifolds	No	No

Table 2. Comparison of Adaptive Harmonic Potential Fields (proposed herein) with common alternatives, specifically Rimón–Koditchek Navigation Functions (RKNF) [15], Harmonic Navigation Functions [33] and approximate Harmonic Potential Fields obtained using numerical techniques [21]. Unlike RKNFs that require tuning for ensuring convergence to the goal from almost all initial configurations and HNFs that require tuning for guaranteeing collision avoidance with the workspace boundaries, the proposed control law enjoys both properties by design.

	Convergence	Collision Avoidance	Computational Cost
AHPF	By design	By design	Cheap
HPF	By design	By design	Expensive
RKNF	Requires tuning	By design	Cheap
HNF	By design	Requires tuning	Cheap

1.2. Preliminaries

We use $\mathcal{D}_r(x)$ to denote an open disk with radius $r > 0$ centered at $x \in \mathbb{R}^2$. Additionally, \mathcal{D} and $\partial\mathcal{D}$ denote the closed disk and circle with unit radii centered at the origin of \mathbb{R}^2 , respectively. Furthermore, let $\mathcal{J}_N \triangleq \{1, 2, \dots, N\}$ and $\mathcal{J}_N^* \triangleq \{0\} \cup \mathcal{J}_N$. Given sets $A, B \subseteq \mathbb{R}^n$, we use $\text{cl}(A)$, ∂A , $\text{int}(A)$ and \bar{A} to denote the closure, boundary, interior and complement of A with respect to \mathbb{R}^n , respectively, and $A \setminus B$ to denote the complement of B with respect to A . Furthermore, we use $\mathbf{0}_N$ and $\mathbf{1}_N$ to denote the all-zeros and all-ones column vectors of

length N , respectively, and $\mathbf{0}_{N \times M}$ to denote the $N \times M$ zero matrix. We also define $\mathbf{1}_{N \times M}^k$, $k \in \mathcal{J}_M$ as the $N \times M$ matrix whose k -th column is equal to $\mathbf{1}_N$ and every other column is equal to $\mathbf{0}_N$. Given a vector function $f(x)$, we use $\nabla_x f$ to denote its Jacobian matrix. Furthermore, given an arc C , we use $|C|$ to denote its length. We will also say that a set A is attractive (repulsive) under a potential function ψ when there exists a point $p_0 \notin \text{cl}(A)$ such that if we initialize at p_0 and move along the negated gradient of ψ , we will converge (not converge) to ∂A . Finally, a potential function ψ is called harmonic if it satisfies the Laplace equation, i.e., $\nabla^2 \psi = 0$, where ∇^2 denotes the Laplacian operator. An important property of harmonic functions is the principle of superposition, which follows from the linearity of the Laplace equation. Moreover, the extrema of a non-constant harmonic function occur on the boundary of the domain of definition, thus excluding any local minima/maxima within it (a desirable property for motion planning).

2. Problem Formulation

We consider a robot operating within a compact workspace $\mathcal{W} \subset \mathbb{R}^2$ bounded by a single outer and a finite set of inner disjoint Jordan curves (a Jordan curve is a non-self-intersecting continuous planar closed curve), which correspond to the boundaries of static obstacles. It is assumed that \mathcal{W} can be written as:

$$\mathcal{W} = \overline{\mathcal{W}_0} \setminus \bigcup_{i \in \mathcal{J}_N} \mathcal{W}_i$$

where \mathcal{W}_i , $i \in \mathcal{J}_N^*$ denote regions of \mathbb{R}^2 that the robot cannot occupy (see left subplot in Figure 1). Particularly, the complement of \mathcal{W}_0 is considered to be a bounded, simply connected region that may also include a strict subset of its own boundary (this corresponds to cases when we wish to place the robot's goal configuration on some part of the workspace outer boundary which is not physically occupied by an actual obstacle, e.g., the door of a compartment (refer to Section 5.2 for more details)) and $\mathcal{W}_1, \mathcal{W}_2, \dots, \mathcal{W}_N$ are assumed to be closed, simply connected compact sets that are contained in $\overline{\mathcal{W}_0}$ and are pairwise disjoint. Let $p = [x, y]^T \in \mathbb{R}^2$ denote the robot's position and assume that the robot's motion is described by the single integrator model:

$$\dot{p} = u \tag{1}$$

where $u \in \mathbb{R}^2$ is the corresponding control input vector.

Problem 1. *Our goal is to design a control law to successfully drive a robot with kinematics (1) towards a given goal configuration $p_d \in \mathcal{W}$ from almost any feasible initial configuration $p_{\text{init}} \in \mathcal{W}$, while ensuring collision avoidance, i.e., $p(t) \in \mathcal{W}$ for all $t \geq 0$.*

Remark 1. *The results presented in this work can be readily employed for the navigation of disk robots with radius $R > 0$ by appropriately augmenting the workspace boundaries with the robot's size.*

3. Harmonic Maps for Planar Navigation

In this section, we present a methodology that maps the robot's workspace onto a punctured unit disk, over which the robot's control law is designed. Particularly, our goal is to construct a transformation, $T : \text{cl}(\mathcal{W}) \mapsto \mathcal{D}$, from the closure of the robot's configuration space $\text{cl}(\mathcal{W})$ onto the unit disk \mathcal{D} with the following properties:

1. $T(\cdot)$ maps the outer boundary $\partial \mathcal{W}_0$ to the unit circle $\partial \mathcal{D}$;
2. $T(\cdot)$ maps the boundary $\partial \mathcal{W}_i$, $i \in \mathcal{J}_N$ of each obstacle to a distinct point $q_i = [u_i, v_i]^T \in \text{int}(\mathcal{D})$;
3. $T(\cdot)$ is a diffeomorphism for all $p \in \text{int}(\mathcal{W})$.

To that end, we compute a transformation $\tilde{T}(p) = [\tilde{u}(p), \tilde{v}(p)]^T$, with $\tilde{u}(p)$ and $\tilde{v}(p)$ being harmonic functions with respect to p , by approximating an ideal harmonic map T that meets the aforementioned properties and the existence of which was proven in Theorem 2 of [38], accompanied by sufficient conditions that render it a diffeomorphism as outlined in the corresponding proof.

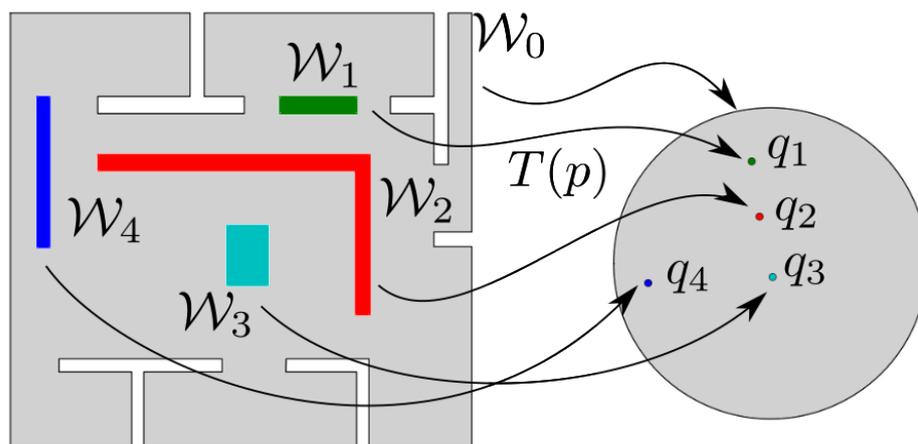


Figure 1. Transformation of a workspace onto a punctured disk.

Particularly, this theorem implies that given an orientation-preserving, weak homeomorphism $T_\partial : \partial\mathcal{W}_0 \rightarrow \partial\mathcal{D}$ (such a transformation can be easily obtained for any given planar Jordan curve C by (1) arbitrarily selecting a point p_o on C , (2) defining $\ell(p), \forall p \in C$ as the length of the arc $\widehat{p_o p}$, assuming one travels from p_o to p on C while having the curve’s interior to its left and (3) choosing $T_\partial(p) = [\cos(2\pi\ell(p)/L), \sin(2\pi\ell(p)/L)]^T$, where $L = |C|$) from the workspace outer boundary $\partial\mathcal{W}_0$ to the boundary of the unit disk, the harmonic map T that satisfies the conditions:

$$T(p) = T_\partial(p) \triangleq [\tilde{u}(p), \tilde{v}(p)]^T, \quad \forall p \in \partial\mathcal{W}_0, \tag{2}$$

$$\int_{\partial\mathcal{W}_i} \frac{\partial T}{\partial n_p} ds = 0, \quad \forall i \in \mathcal{I}_N \tag{3}$$

with n_p denoting the unit vector that is normal to the boundary at the point $p \in \partial\mathcal{W}_i, i \in \mathcal{I}_N$, is a diffeomorphism that maps $\text{cl}(\mathcal{W})$ to the target set and collapses each inner obstacle \mathcal{W}_i onto a distinct point q_i within its interior (see Figure 1). Note that the coordinates of q_i , i.e., the images of the internal obstacles, are not explicitly specified but are computed as part of the solution, such that the aforementioned constraints are satisfied.

Given that closed-form solutions to the aforementioned problem are generally not available for non-trivial domains, in this work, we employed numerical techniques and particularly the Panel Method [24,39,40] (similar formulations can be obtained by employing other numerical techniques such as the Boundary Element Method (BEM), the Finite Element Method (FEM) or the Finite Differences Method (FDM)) in order to construct a harmonic map \tilde{T} that sufficiently approximates T . As such, by subdividing separately the workspace’s outer and inner boundaries into $\tilde{M}_0, \tilde{M}_1, \dots, \tilde{M}_N$ number of elements (see Figure 2), we define the components of $\tilde{T}(p) = [\tilde{u}(p), \tilde{v}(p)]^T$ as follows:

$$\begin{aligned} \tilde{u}(p) &= \sum_{i=0}^N \sum_{j=1}^{\tilde{M}_i} \sum_{l=1}^{\tilde{M}_C} \tilde{C}_{ijl}^x \tilde{H}_{ijl}(p) \\ \tilde{v}(p) &= \sum_{i=0}^N \sum_{j=1}^{\tilde{M}_i} \sum_{l=1}^{\tilde{M}_C} \tilde{C}_{ijl}^y \tilde{H}_{ijl}(p) \end{aligned}$$

$$\tilde{H}_{ijl}(p) = \int_{\tilde{E}_{ij}} \tilde{G}_{ijl}(s) \ln(\|p - \tilde{p}_{i,j}(s)\|) ds \tag{4}$$

where \tilde{M}_C is the number of control parameters per element, \tilde{E}_{ij} denotes the j -th element of the i -th boundary's approximation, $\tilde{p}_{i,j}(s) : [0, |\tilde{E}_{ij}|] \mapsto \tilde{E}_{ij}$ is a bijective parameterization of \tilde{E}_{ij} , $\tilde{G}_{ijl} : [0, |\tilde{E}_{ij}|] \mapsto \mathbb{R}$ is the shape function corresponding to the l -th control parameter of \tilde{E}_{ij} and $\tilde{C}_{ijl}^x, \tilde{C}_{ijl}^y \in \mathbb{R}$ are control parameters that need to be appropriately selected so that \tilde{T} satisfies properties 1–3 for all $l \in \mathcal{I}_{\tilde{M}_C}, j \in \mathcal{I}_{\tilde{M}_i}$ and $i \in \mathcal{I}_N^*$. It is worth noting that for common choices of \tilde{G}_{ijl} (e.g., constant or linear shape functions) and simple types of \tilde{E}_{ij} (e.g., line segments), the integral in (4) can be easily evaluated to obtain a closed-form expression for \tilde{H}_{ijl} . As an illustration, for a line segment element \tilde{E}_{ij} with two control parameters (i.e., $\tilde{M}_C = 2$), a typical choice for linear shape functions (see Figure 2) is $\tilde{G}_{ij1}(s) = s/|\tilde{E}_{ij}|$, $\tilde{G}_{ij2}(s) = (1 - s/|\tilde{E}_{ij}|)$ and $\tilde{p}_{i,j}(s) = \tilde{p}_{i,j,A} + (\tilde{p}_{i,j,B} - \tilde{p}_{i,j,A})s/|\tilde{E}_{ij}|$ for the corresponding parameterization, where $\tilde{p}_{i,j,A}, \tilde{p}_{i,j,B}$ are the element's end-points. To obtain the unknown control parameters as well as the images of the workspace's inner obstacles, one needs to solve two independent linear systems of equations:

$$\tilde{A}\tilde{X} = \tilde{B}_x, \quad \tilde{A}\tilde{Y} = \tilde{B}_y \tag{5}$$

for the unknown vectors:

$$\begin{aligned} \tilde{X} &= [\tilde{C}_{0,1,1}^x, \dots, \tilde{C}_{1,1,1}^x, \dots, \tilde{C}_{N,\tilde{M}_N,\tilde{M}_C}^x, \mathbf{u}_1, \dots, \mathbf{u}_N]^T \\ \tilde{Y} &= [\tilde{C}_{0,1,1}^y, \dots, \tilde{C}_{1,1,1}^y, \dots, \tilde{C}_{N,\tilde{M}_N,\tilde{M}_C}^y, \mathbf{v}_1, \dots, \mathbf{v}_N]^T. \end{aligned}$$

The matrix \tilde{A} and the right hand side vectors \tilde{B}_x and \tilde{B}_y are constructed by selecting a set of $\sum_{i \in \mathcal{I}_N^*} \tilde{m}_i$ arbitrary points $\tilde{p}_{i,j}^*$ (a typical strategy is to select the points $\tilde{p}_{i,j}^*$ uniformly on the outer and inner boundaries of the given domain) such that a) $\tilde{p}_{i,j}^* \in \partial\mathcal{W}_i$ for all $j \in \mathcal{I}_{\tilde{m}_i}$ and $i \in \mathcal{I}_N^*$ and b) $\sum_{i \in \mathcal{I}_N^*} \tilde{m}_i = \tilde{M}_C \sum_{i \in \mathcal{I}_N^*} \tilde{M}_i$, and evaluating (2) and (3) at those points as follows:

$$\begin{aligned} \tilde{A} &= \begin{bmatrix} \tilde{A}_0 & \mathbf{0}_{\tilde{m}_0 \times N} \\ \tilde{A}_1 & -\mathbf{1}_{\tilde{m}_1 \times N} \\ \vdots & \vdots \\ \tilde{A}_N & -\mathbf{1}_{\tilde{m}_N \times N} \\ \tilde{A}_\dagger & \mathbf{0}_{N \times N} \end{bmatrix}, \quad \tilde{B}_x = \begin{bmatrix} \tilde{B}_{x,0} \\ \mathbf{0}_{\tilde{m}_1} \\ \vdots \\ \mathbf{0}_{\tilde{m}_N} \\ \mathbf{0}_N \end{bmatrix}, \quad \tilde{B}_y = \begin{bmatrix} \tilde{B}_{y,0} \\ \mathbf{0}_{\tilde{m}_1} \\ \vdots \\ \mathbf{0}_{\tilde{m}_N} \\ \mathbf{0}_N \end{bmatrix} \\ \tilde{A}_k &= \begin{bmatrix} \tilde{H}_{0,1,1}(\tilde{p}_{k,1}^*) & \cdots & \tilde{H}_{N,\tilde{M}_N,\tilde{M}_C}(\tilde{p}_{k,1}^*) \\ \tilde{H}_{0,1,1}(\tilde{p}_{k,2}^*) & \cdots & \tilde{H}_{N,\tilde{M}_N,\tilde{M}_C}(\tilde{p}_{k,2}^*) \\ \vdots & \vdots & \vdots \\ \tilde{H}_{0,1,1}(\tilde{p}_{k,\tilde{m}_k}^*) & \cdots & \tilde{H}_{N,\tilde{M}_N,\tilde{M}_C}(\tilde{p}_{k,\tilde{m}_k}^*) \end{bmatrix}, \quad \forall k \in \mathcal{I}_N^* \\ \tilde{A}_\dagger &= \begin{bmatrix} \sum_{k=1}^{\tilde{m}_1} \frac{\partial \tilde{H}_{0,1,1}}{\partial n_{0,1}}(\tilde{p}_{1,k}^*) & \cdots & \sum_{k=1}^{\tilde{m}_1} \frac{\partial \tilde{H}_{N,\tilde{M}_N,\tilde{M}_C}}{\partial n_{N,\tilde{M}_N}}(\tilde{p}_{1,k}^*) \\ \sum_{k=1}^{\tilde{m}_2} \frac{\partial \tilde{H}_{0,1,1}}{\partial n_{0,1}}(\tilde{p}_{2,k}^*) & \cdots & \sum_{k=1}^{\tilde{m}_2} \frac{\partial \tilde{H}_{N,\tilde{M}_N,\tilde{M}_C}}{\partial n_{N,\tilde{M}_N}}(\tilde{p}_{2,k}^*) \\ \vdots & \vdots & \vdots \\ \sum_{k=1}^{\tilde{m}_N} \frac{\partial \tilde{H}_{0,1,1}}{\partial n_{0,1}}(\tilde{p}_{N,k}^*) & \cdots & \sum_{k=1}^{\tilde{m}_N} \frac{\partial \tilde{H}_{N,\tilde{M}_N,\tilde{M}_C}}{\partial n_{N,\tilde{M}_N}}(\tilde{p}_{N,k}^*) \end{bmatrix} \end{aligned}$$

$$\tilde{B}_{x,0} = \begin{bmatrix} \bar{u}(\tilde{p}_{0,1}^*) \\ \bar{u}(\tilde{p}_{0,2}^*) \\ \vdots \\ \bar{u}(\tilde{p}_{0,\tilde{m}_0}^*) \end{bmatrix} \quad \tilde{B}_{y,0} = \begin{bmatrix} \bar{v}(\tilde{p}_{0,1}^*) \\ \bar{v}(\tilde{p}_{0,2}^*) \\ \vdots \\ \bar{v}(\tilde{p}_{0,\tilde{m}_0}^*) \end{bmatrix}.$$

Notice that by discretizing the workspace boundaries into a large number of sufficiently small elements, the overall approximation error between the solution \tilde{T} of the aforementioned linear problem and the exact transformation T can be rendered arbitrarily small (see [39,40]). However, the complexity of constructing the mapping is of order $O(\bar{M}^3)$, where \bar{M} denotes the number of total elements of the mapping (i.e., the complexity of the solution of the dense system of linear Equation (5)). Nevertheless, the construction of the transformation, which is the main computational bottleneck, is performed only once at the beginning. Additionally, apart from the straightforward user-defined homeomorphism $T_{\mathcal{D}}$ on the workspace boundary, no tedious trial and error tuning is needed to extract the diffeomorphic transformation \tilde{T} , in contrast to other related works such as the Star-to-Sphere Transformation (SST) [15], the Multi-Agent Navigation Transformation (MANT) [36] and the Navigation Transformation (NT) [34].

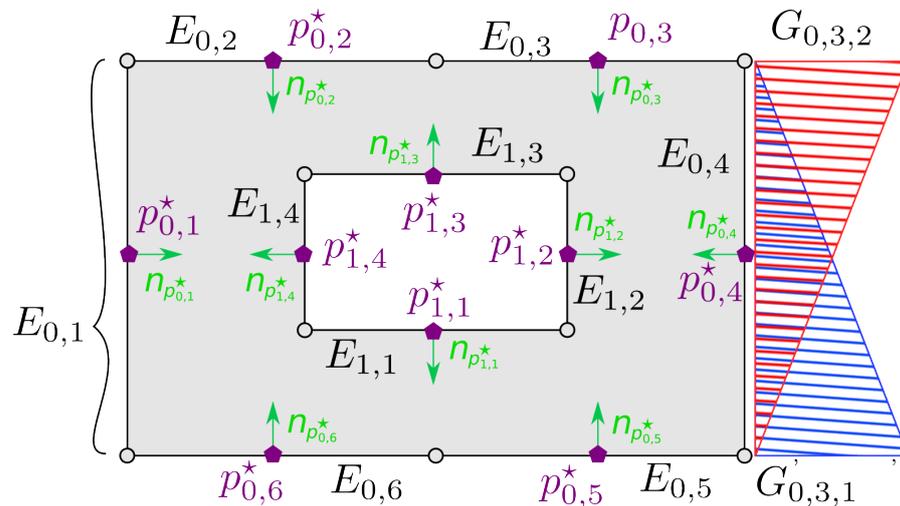


Figure 2. Discretization of a given domain’s boundary using line segment elements. By convention, the outer boundary is considered to be clockwise oriented, whereas inner boundaries are counter-clockwise oriented. The normal direction of each element is depicted using green colored vectors. Furthermore, the values of the two linear shape functions $\tilde{G}_{0,3,1}$ and $\tilde{G}_{0,3,2}$ are plotted along the associated element $\tilde{E}_{0,3}$.

4. Control Design

To address Problem 1, we equip the robot with the aforementioned transformation $q = T(p)$ from the closure of its configuration space \mathcal{W} onto the unit disk \mathcal{D} and an artificial potential $\psi(q, k)$ augmented with an adaptive control law $\dot{k} = f_k(q, k)$ for its parameters $k = [k_d, k_1, k_2, \dots, k_N]^T$. The robot velocity control law is calculated as follows:

$$u = -\mathcal{K}_u s(q, k) J^{-1}(p) \nabla_q \psi(q, k) \tag{6}$$

where $J(p)$ denotes the Jacobian matrix of $T(p)$, $s(q, k) \geq 0$ is a continuously differentiable gain function given by:

$$s(q, k) = \gamma \sigma_p \left(\frac{1 - \|q\|}{\epsilon_p} \right) + (1 - \gamma) \sigma_v \left(\frac{(\nabla_q \psi)^T q}{\epsilon_v + \|\nabla_q \psi\| \|q\|} \right) \tag{7}$$

with

$$\sigma_p(x) = \begin{cases} x^2(3 - 2x), & \text{if } x \leq 1 \\ 1, & \text{if } x > 1 \end{cases}$$

$$\sigma_v(x) = \begin{cases} x^2, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \quad (8)$$

and \mathcal{K}_u , γ , ϵ_p and ϵ_v are scalar constants such that $\mathcal{K}_u, \epsilon_v > 0$ and $\gamma, \epsilon_p \in (0, 1)$. More specifically, $s(q, k)$ consists of two individual terms, with the first vanishing as the robot approaches the workspace's outer boundary (and its distance from the unit circle is less than ϵ_p) and the second vanishing when the robot's velocity points away from the disk's center. The scalar parameter γ can be used for adjusting the contribution of each respective term of $s(q, k)$. Finally, ψ is a harmonic artificial potential field defined on the image $T(\mathcal{W})$ of the workspace \mathcal{W} and whose negated gradient $-\nabla_q \psi(q, k)$ defines the direction of the robot's motion in the real workspace \mathcal{W} via the inverse Jacobian $J^{-1}(p)$. By design, the resultant vector field precludes collisions between the robot and the workspace's inner obstacles and renders the goal configuration almost globally attractive except for a set of measure zero initial configurations. However, since \mathcal{W}_0 may not be repulsive under ψ for an arbitrary, fixed selection of k , we also introduce the adaptive law $f_k(q, k)$ which, along with $s(q, k)$, guarantees forward invariance of the workspace without compromising the convergence and stability properties of the overall system. The following subsections elaborate on each component of the proposed control law individually.

4.1. Artificial Harmonic Potential Fields

We construct an artificial harmonic potential field on the disk space \mathcal{D} employing point sources placed at the desired configuration $q_d = T(p_d)$ as well as at the points $q_i = T(\partial\mathcal{W}_i)$, $\forall i \in \mathcal{I}_N$ that correspond to the inner obstacles, as follows:

$$\phi(q, k) = k_d \ln \left(\frac{\|q - q_d\|}{2} \right) - \sum_{i=1}^N k_i \ln \left(\frac{\|q - q_i\|}{2} \right) \quad (9)$$

where $k_d > 0$ and $k_i \geq 0$ denote harmonic source strengths which vary according to adaptive laws that are presented later. An interesting property of the above potential field, which stems from the maximum principle for harmonic functions, is that, for fixed k , the only minima of ϕ are located at q_d and, possibly, at infinity. As a direct consequence of this property, the Hessian $\nabla_q^2 \phi$ computed at a non-degenerate critical point of ϕ in our domain's interior has one positive and one negative eigenvalue with the same magnitude, e.g., λ and $-\lambda$ with $\lambda > 0$.

Next, we define a reference potential ψ based on ϕ , which is given by:

$$\psi(q, k) = \frac{1 + \tanh(\phi(q, k)/w_\phi)}{2} \quad (10)$$

where w_ϕ is a positive scaling constant. Note that ψ maps the extended real line to the closed interval $[0, 1]$. As $\tanh(\phi/w_\phi)$ is a strictly increasing function, the only critical points of ψ are the ones inherited from ϕ with their indices preserved. Furthermore, the gradient of ψ with respect to q , given by

$$\nabla_q \psi = \frac{1 - (\tanh(\phi/w_\phi))^2}{2w_\phi} \nabla_q \phi, \quad (11)$$

is well defined and bounded for all $q \in \mathcal{D}$.

If the workspace was radially unbounded, selecting k fixed with $k_d > \sum_{i=1}^N k_i$ would render the potential field (10) sufficient for navigation. The author of [33] addresses bounded workspaces that are diffeomorphic to sphere worlds by simply mapping the

outer bounding circle to infinity. In this work, we would like to be able to place q_d on regions of $\partial\mathcal{D}$ that are not physically occupied by obstacles (such as passages to other compartments, see, for example, Section 5.2); thus, we cannot follow the same procedure since that would render the effect of the sole attractor on the robot null. Instead, we design appropriate adaptive laws for the parameters k of ϕ to render the outer boundary repulsive and establish the forward completeness of the proposed scheme at all times.

Before proceeding with the definition of the adaptive law, we first state two propositions that will be used in the subsequent analysis, the proofs of which can be found in the Appendix A.

Proposition 1. *Let $k_d > 0$ and $q' \in \partial\mathcal{D} \setminus \{q_d\}$. There exists $k' > 0$ such that if $k_i < k', \forall i \in \mathcal{I}_N$, then q' is repulsive under ψ .*

Proposition 2. *If k_i are non-negative and bounded, there exists $k'_d > 0$ such that ψ is Morse for all $k_d \geq k'_d$.*

4.2. Adaptive Laws

We now present the adaptive law $\dot{k} = f_k(q, k)$ that updates the parameters of the potential field ψ . Its primary goal is to render (a) the workspace outer boundary repulsive and (b) any critical point of ϕ in the vicinity of the robot non-degenerate, a property that will be used later in the analysis. In particular, we consider f_k of the form:

$$\begin{aligned} \dot{k}_d &= \xi_1(\lambda + \|\nabla_q \phi\|; \epsilon_1) \\ \dot{k}_i &= (\bar{k}_i - k_i)w_i \ell_i g_i - \mathcal{K}_k k_i h_i w_0 (g_0 + \xi_1(s; \epsilon_2)), \quad \forall i \in \mathcal{I}_N \end{aligned} \quad (12)$$

where w_i and $g_i, i \in \mathcal{I}_N^*$, as well as $h_i, i \in \mathcal{I}_N$, are functions to be defined later, $\bar{k}_i, i \in \mathcal{I}_N$ are desired upper bounds for k_i , λ denotes the non-negative eigenvalue of $\nabla_q^2 \phi$, \mathcal{K}_k is a positive control gain and ϵ_1 and ϵ_2 are small positive constants. The continuously differentiable switch $\xi_1(x; \epsilon)$ and functions $\ell_i(q)$ are, respectively, given by:

$$\begin{aligned} \xi_1(x; \epsilon) &= 1 - \sigma_p(x/\epsilon) \\ \ell_i(q) &= -\frac{\mathcal{K}_u s(q, k)}{\ln\left(\frac{\|q - q_i\|}{2}\right)}. \end{aligned} \quad (13)$$

According to Proposition 1, our first requirement can be accomplished by designing f_k to reduce k_i as the robot approaches $\partial\mathcal{D}$. To do so without compromising the inherent inner obstacle collision avoidance properties of ϕ , we need to also ensure that each k_i does not vanish within some neighborhood of q_i for all $i \in \mathcal{I}_N$. To that end, firstly we define g_i , employing the smoothly vanishing function defined in (8) to serve as pseudo-metrics of the alignment between the robot's velocity and the directions towards the goal and inner obstacles, respectively, given by:

$$g_i = \sigma_v(\bar{g}_i), \quad \forall i \in \mathcal{I}_N^* \quad (14)$$

with

$$\begin{aligned} \bar{g}_0 &= \frac{1}{4} \left(\alpha \|\nabla_q \psi\| \|q - q_d\| - (\nabla_q \psi)^T (q - q_d) \right) \\ \bar{g}_i &= \frac{1}{2} (\nabla_q \psi)^T (q - q_i), \quad \forall i \in \mathcal{I}_N \end{aligned}$$

where $\alpha \in (0, 1]$ is a fixed constant that is used for selecting the desired alignment between the robot's motion and the direction to the goal. We also define the accompanying weights

w_i as follows to ensure that only one term of (12) dominates as the robot approaches a particular boundary of \mathcal{W} :

$$\begin{aligned} w_0 &= \frac{\xi_2(\bar{w}_0; \epsilon_3)}{\bar{w}_0 + \sum_{j=1}^N (\bar{k}_j \bar{w}_j)} \\ w_i &= \frac{\bar{w}_i}{\bar{w}_0 + \sum_{j=1}^N (\bar{k}_j \bar{w}_j)}, \quad \forall i \in \mathcal{I}_N \end{aligned} \quad (15)$$

with

$$\begin{aligned} \bar{w}_i &= \bar{r}_i / (r_i + \bar{r}_i), \quad \forall i \in \mathcal{I}_N^*, \\ r_0 &= (1 - \|q\|)^2, \quad r_i = \|q - q_i\|^2, \quad \forall i \in \mathcal{I}_N \\ \bar{r}_i &= \sqrt[m]{\sum_{j \neq i} (r_j)^m}, \quad \forall i \in \mathcal{I}_N \end{aligned} \quad (16)$$

$$\xi_2(x; \epsilon) = \begin{cases} 0, & \text{if } x < \epsilon \\ \left(\frac{x-\epsilon}{1-\epsilon}\right)^2 (3 - 2\frac{x-\epsilon}{1-\epsilon}), & \text{if } \epsilon \leq x \leq 1 \\ 1, & \text{otherwise} \end{cases}$$

for a scalar constant $\epsilon_3 \in (0, 1)$ in (15) and some integer $m < -1$ in (16) that serves as a smooth under-approximation of $\min_{j \neq i} (r_j)$, $i \in \mathcal{I}_N$. Finally, the weights h_i , $i \in \mathcal{I}_N$ are defined as follows:

$$h_i = 1 + \frac{\sigma_v(\bar{h}_i)}{1 + \sum_{j \in \mathcal{I}_N} \sigma_v(\bar{h}_j)}$$

with

$$\bar{h}_i = k_i \left(1 - (\tanh(\phi/w_\phi))\right)^2 \left(\frac{q_d - q}{\|q_d - q\|^2}\right)^T \frac{q_i - q}{\|q_i - q\|^2}$$

whose purpose is to accelerate the decay of those k_i that contribute the most to the component of $\nabla_q \psi$ that pushes the robot toward the workspace's outer boundary.

Regarding the second requirement, as shown in Proposition 1, selecting a k_d above a certain threshold is sufficient to render ϕ free of degenerate equilibria. On the other hand, for a given \bar{k}_i , increasing k_d steers the robot closer to the workspace's inner obstacles. Nevertheless, since the robot may never actually enter the vicinity of a degenerate equilibrium, instead of setting k_d sufficiently large a priori, the adaptive law for the parameter k_d is introduced to increase k_d only when it is actually needed, thus alleviating the aforementioned shortcoming.

4.3. Stability Analysis

Let us consider the overall system:

$$\dot{z} = f_z(z) \quad (17)$$

where $z = (q, k)$ and $f_z(z) = (f_q, f_k)$ with $f_q = Ju$. Furthermore, let Ω denote the image of \mathcal{W} under T , i.e., $\Omega = T(\mathcal{W})$. Note that Ω consists of $\text{int}(\mathcal{D})$, possibly with a subset of $\partial\mathcal{D}$, with the points q_i removed. In this section, we elaborate on the stability properties of (17) under the proposed control scheme (6) and (12). First, we formalize the safety properties of the closed-loop system dynamics, which guarantee that our robot does not collide with any obstacle.

Proposition 3. *The workspace \mathcal{W} is invariant under the dynamics (17) with control laws (6) and (12), i.e., $p(t) \in \mathcal{W}$ for all $t \geq 0$.*

Proof. For the proof, refer to the Appendix A. \square

Having eliminated the possibility of the robot colliding with the workspace's boundaries, we proceed by showing that all critical points of ψ , where (17) may converge to, are either non-degenerate saddles or q_d . Additionally, we show that the latter is a stable equilibrium.

Proposition 4. *The artificial potential ψ decreases along the trajectories of the closed-loop system and its time derivative vanishes only at its critical points. Additionally, the preimage of q_d is a set of stable equilibria of (1).*

Proof. For the proof, refer to the Appendix A. \square

Proposition 5. *Let $z^* = (q^*, k^*)$ be a critical point of the closed-loop system dynamics with $q^* \in \Omega \setminus \{q_d\}$. Then, q^* is a non-degenerate saddle point of ψ .*

Proof. For the proof, refer to the Appendix A. \square

Finally, we conclude this section with the main theoretical findings.

Theorem 1. *System (1) under the control law (6) and (12) converges safely to p_d , for almost all initial configurations, thus addressing successfully Problem 1.*

Proof. For the proof, refer to the Appendix A. \square

Remark 2. *Owing to the adaptive laws (12) that modify the harmonic source strengths online to secure the safety and convergence properties at all times, the selection of the fixed control parameters in the proposed scheme, i.e., $\mathcal{K}_u, \gamma, \epsilon_p, \epsilon_v, w_\phi, \mathcal{K}_k, \epsilon_1, \epsilon_2, \alpha$ and ϵ_3 , is straightforward as it affects only the trajectory evolution within the workspace and not the aforementioned critical properties. Consequently, their values should be set freely as opposed to NFs, where the selection of the main parameters severely affects the convergence properties of the adopted scheme and cannot be conducted constructively for generic workspaces of arbitrary topology.*

5. Extensions

In this section, we present certain extensions of the proposed approach to (a) address the safe navigation problem for unicycle robots which are frequently encountered in many application domains and (b) tackle computational complexity issues that affect the numerical computation of the harmonic map presented in Section 3 as the size of the workspace increases.

5.1. Unicycle Robot Kinematics

In this subsection, we consider robots whose motion is subjected to Pfaffian constraints of the form:

$$\begin{aligned} \dot{p} &= n_\theta v \\ \dot{\theta} &= \omega \end{aligned} \quad (18)$$

where $\theta \in [0, 2\pi)$ denotes the robot's orientation, $n_\theta = [\cos(\theta), \sin(\theta)]^T$, and $v, \omega \in \mathbb{R}$ are control inputs corresponding to the robot's linear and angular velocities, respectively. First, let us define the robot's kinematics in the image of the configuration space via the proposed transformation as follows:

$$\begin{aligned} \dot{q} &= n_{\hat{\theta}} \hat{v} \\ \dot{\hat{\theta}} &= \hat{\omega} \end{aligned} \quad (19)$$

Note that the orientations θ and $\hat{\theta}$ are related by:

$$n_{\hat{\theta}} = \frac{J_p n_\theta}{\|J_p n_\theta\|}$$

where $J_p = J(p)$. To safely drive the robot to its goal configuration, we consider the following control laws:

$$\begin{aligned} \hat{v} &= -\mathcal{K}_v s_v(q, \hat{\theta}, k) (n_{\hat{\theta}})^T \nabla_q \psi(q, k) \\ \hat{\omega} &= -\mathcal{K}_\omega (n_{\hat{\theta}}^\perp)^T \nabla_q \psi(q, k) \end{aligned} \tag{20}$$

with $\mathcal{K}_v, \mathcal{K}_\omega \in \mathbb{R}$ positive constant gains, $n_{\hat{\theta}}^\perp = [-\sin(\theta), \cos(\theta)]^T$ and

$$s_v(q, \hat{\theta}, k) = \gamma \sigma_p \left(\frac{1 - \|q\|}{\epsilon_p} \right) + (1 - \gamma) \sigma_v \left(\frac{(n_{\hat{\theta}}^T \nabla_q \psi) n_{\hat{\theta}}^T q}{\epsilon_v + |n_{\hat{\theta}}^T \nabla_q \psi| \|q\|} \right).$$

Additionally, we need to employ a modified version of the adaptive law for the potential field parameters, which is obtained by substituting s with s_v in (12) and (13) and $\bar{g}_i, i \in \mathcal{I}_N^*$, with

$$\begin{aligned} \bar{g}_{v,0} &= \frac{1}{4} \left(\alpha |n_{\hat{\theta}}^T \nabla_q \psi| \|q - q_d\| - (n_{\hat{\theta}}^T \nabla_q \psi) n_{\hat{\theta}}^T (q - q_d) \right) \\ \bar{g}_{v,i} &= \frac{1}{2} (n_{\hat{\theta}}^T \nabla_q \psi) n_{\hat{\theta}}^T (q - q_i), \quad \forall i \in \mathcal{I}_N \end{aligned}$$

respectively, in (14). Finally, by expressing the aforementioned control laws to the robot’s actual configuration space, we obtain:

$$\begin{aligned} v &= v \hat{v} \\ \omega &= \omega_{dq} + \omega_{d\hat{\theta}} \end{aligned} \tag{21}$$

where ω_{dq} and $\omega_{d\hat{\theta}}$ are terms corresponding to angular velocities induced by translational and rotational motion of the robot in the workspace’s image, respectively, given by:

$$\begin{aligned} \omega_{dq} &= -\hat{v}^2 \left((J_p [n_\theta \quad v n_\theta^\perp])^{-1} \frac{\partial}{\partial n_\theta} J_p n_\theta \right)^T \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \omega_{d\hat{\theta}} &= \hat{\omega} \left((J_p [n_\theta \quad v n_\theta^\perp])^{-1} n_{\hat{\theta}}^\perp \right)^T \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{aligned}$$

with $v = \|J_p^{-1} n_{\hat{\theta}}\|$ and $\frac{\partial}{\partial n_\theta} J_p$ denoting the directional derivative of J_p along n_θ .

The stability properties of the aforementioned closed-loop system dynamics are formalized below.

Theorem 2. *The workspace \mathcal{W} is invariant under the dynamics of (18) equipped with the proposed control law. Additionally, the robot will asymptotically converge either to an interior critical point of ϕ or to the pre-image of q_d , which is stable.*

Proof. For the proof, refer to the Appendix A. \square

Remark 3. *The result of Theorem 2 is weaker compared to that of Theorem 1, since there is no guarantee that the set of configurations which converge to a critical point of ϕ (other than the pre-image of q_d) has Lebesgue measure zero.*

5.2. Atlas of Harmonic Maps

As the size of the workspace increases, the problem of computing the transformation T grows in complexity as well, because the resources required by commonly employed numerical techniques that can solve the problem presented in Section 3 are polynomial in the number of elements used for representing \mathcal{W} . Alternatively, to cope with large workspaces efficiently, we propose instead the construction of an atlas $\mathcal{A} \triangleq \{(\mathcal{P}_i, T_i) \mid i \in$

\mathcal{J}_{N_A} obtained by separating the workspace \mathcal{W} into N_A overlapping subsets $\mathcal{P}_i \subset \mathcal{W}$, such that $\bigcup_{i \in \mathcal{J}_{N_A}} \mathcal{P}_i = \mathcal{W}$ and constructing a separate harmonic map T_i for each \mathcal{P}_i (see Figure 3).

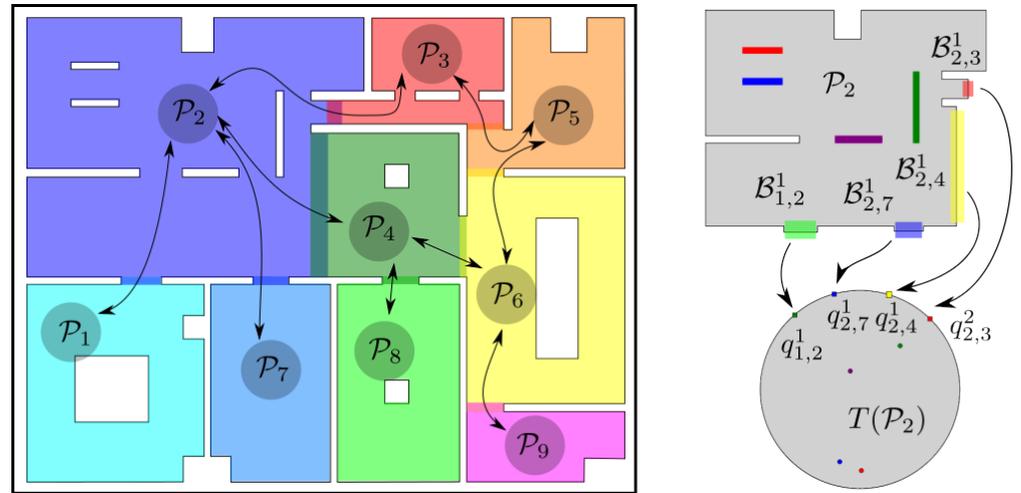


Figure 3. The partition of a complex workspace into overlapping subsets along with the corresponding graph and the transformation T_2 of the second partition \mathcal{P}_2 .

This essentially allows us to solve many small (and computationally less intensive) problems instead of a large one, thus reducing the overall resources required for addressing a given workspace. Therefore, given such a partitioning of \mathcal{W} , we define the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{\mathcal{P}_i \mid i \in \mathcal{J}_{N_A}\}$ denotes the set of corresponding nodes (workspace partitions) and \mathcal{E} denotes the set of edges between the elements of \mathcal{V} , with each edge indicating a feasible transition from one partition to another, i.e., $(i, j) \in \mathcal{E}$ if and only if $(\text{cl}(\mathcal{P}_i) \cap \text{cl}(\mathcal{P}_j)) \neq \emptyset$. Note that \mathcal{G} is undirected by definition, i.e., $(i, j) \in \mathcal{E}$ only if $(j, i) \in \mathcal{E}$. Additionally, since the workspace is connected, \mathcal{G} should also be connected. Thus, for a given atlas \mathcal{A} , an initial configuration p_{init} and a final configuration p_d , we can employ standard graph search algorithms to obtain a sequence of indices $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$ corresponding to partitions that the robot can traverse to reach its goal. (In general, more than one such sequence of partitions may exist connecting the initial and the final configurations. However, the selection of one that corresponds to some sort of “optimal” path is beyond the scope of this work.) Additionally, note that since the partitioning of \mathcal{W} does not need to be fine, the size of \mathcal{G} will generally be small, rendering the cost of finding \mathcal{S} negligible.

We now concentrate on how the transition between two consecutive elements of \mathcal{S} is implemented. Let $\mathcal{C}_{i,j} \triangleq \text{cl}(\mathcal{P}_i) \cap \text{cl}(\mathcal{P}_j)$ denote the common region of $\text{cl}(\mathcal{P}_i)$ and $\text{cl}(\mathcal{P}_j)$ and let $\mathcal{B}_{i,j} \triangleq \partial\mathcal{P}_i \cap \mathcal{P}_j$ denote the set of points on the boundary of \mathcal{P}_i that also belong to \mathcal{P}_j and are not occupied by obstacles for all $i \in \mathcal{J}_{N_A}$ and all j such that $(i, j) \in \mathcal{E}$. Without loss of generality, we assume that \mathcal{A} is constructed such that the sets $\mathcal{B}_{\ell,i} \cap \mathcal{B}_{\ell,j}$ are either empty or consist of isolated points. We note that in order to successfully complete the transition between two consecutive nodes \mathcal{P}_i and \mathcal{P}_j of \mathcal{S} , it suffices for the robot to reach any single point of $\mathcal{B}_{i,j}$ from \mathcal{P}_i . We also observe that each $\mathcal{B}_{i,j}$ may consist of one or more disjoint components $\mathcal{B}_{i,j}^\ell$, $\ell \in \mathcal{L}(i, j)$, with $\mathcal{L}(i, j)$ being some valid indexing of those. By exploiting the fact that Theorem 2 [38] imposes a weak homeomorphism requirement on T_i , we can construct each T_i such that each disjoint subset of $\partial\mathcal{P}_i$ collapses into a separate point, i.e., $T_i(\mathcal{B}_{i,j}^\ell) = q_{i,j}^\ell \in \partial\mathcal{D}$ (see Figure 3), which, in turn, implies that selecting $q_{i,j}^\ell$ as an intermediate goal configuration suffices to render the entire $\mathcal{B}_{i,j}^\ell$ attractive. Building upon this fact, for each consecutive pair of \mathcal{P}_i and \mathcal{P}_j in \mathcal{S} , we (arbitrarily) select a $\mathcal{B}_{i,j}^\ell$ and we construct a transformation $T_i : \mathcal{P}_i \mapsto \mathcal{D}$, with $q^{[i]} = T_i(p)$, and artificial potential field $\phi_i(q^{[i]}, k^{[i]})$ with goal configuration $q_d^{[i]} = q_{i,j}^\ell$. Additionally, to smooth the transition

between consecutive partitions, when they overlap, we propose the following modified control law for the robot:

$$u = u^{[i]} + \eta_{c,i,j} \cdot \eta_{t,i,j} \cdot u^{[j]}, \quad \forall p \in C_{i,j} \quad (22)$$

where $u^{[i]}$ and $u^{[j]}$ denote the control inputs as defined in (6) and evaluated using ψ_i, T_i and ψ_j, T_j , respectively; the function $\eta_{t,i,j} : C_{i,j} \mapsto [0, 1]$ is any smooth bump function such that

$$\eta_{t,i,j}(p) = \begin{cases} 0, & \text{if } p \in \mathcal{B}_{j,i} \\ 1, & \text{if } p \in \mathcal{B}_{i,j} \end{cases}$$

and

$$\eta_{c,i,j}(p, k^{[i]}, k^{[j]}) = \begin{cases} \frac{(\zeta_{i,j})^2}{\epsilon_4 + (\zeta_{i,j})^2}, & \text{if } \zeta_{i,j} \geq 0 \\ 0, & \text{if } \zeta_{i,j} < 0 \end{cases}$$

with $\zeta_{i,j} = (\nabla_p \psi_i)^T \cdot (\nabla_p \psi_j)$; and $\epsilon_4 > 0$ is a fixed parameter. What this modification essentially does is incrementally add an extra component, with the direction of $\nabla_p \psi_j$, to the robot's velocity when that component is cosine similar (two vectors u and v are cosine similar if their inner product is positive) with $\nabla_p \psi_i$. We note that $\eta_{c,i,j} \rightarrow 1$ and $\eta_{t,i,j} \rightarrow 1$ as the robot approaches the boundary of the corresponding partition. We also remark that once the robot has completed its transition to \mathcal{P}_j , we do not concern ourselves with $u^{[i]}$ anymore, i.e., $u = u^{[j]}$ even if p returns to $C_{i,j}$. The overall scheme employed for navigating a holonomic robot to its goal configuration using an atlas constructed as described above can be found in Algorithm 1.

Regarding the stability analysis of the modified system, by following the same procedure as in Section 4.3 and by virtue of $\eta_{c,i,j}$, it is trivial to verify the following statement.

Theorem 3. *System (1) equipped with Algorithm 1 converges safely to a given goal configuration $p_d \in \mathcal{W}$ from almost all initial configurations $p_{\text{init}} \in \mathcal{W}$.*

Proof. For the proof, refer to the Appendix A. \square

Algorithm 1 Atlas-based motion planning scheme for a holonomic robot

Require: $\mathcal{A}, p_{\text{init}}, p_d$
 $\mathcal{S} \leftarrow \text{FINDPATHTOGOAL}(\mathcal{G}, p_{\text{init}}, p_d)$
Initialize $k^{[s]}$ for all $s \in \mathcal{S}$.
for all i in \mathcal{I}_{n-1} **do**
 $s, s' \leftarrow s_i, s_{i+1}$
 Select (arbitrary) ℓ such that $\ell \in \mathcal{L}(s, s')$.
 Place goal configuration of ψ_s at $q_{s,s'}^\ell$.
end for
Place goal configuration of ψ_{s_n} at $T_{s_n}(p_d)$.
 $\ell \leftarrow 1$
loop
 if $\ell = n$ or $p \in \mathcal{P}_{s_\ell} \setminus \mathcal{P}_{s_{\ell+1}}$ **then**
 Update p using (6) and $k^{[s_\ell]}$ using (12).
 else if $p \in \mathcal{C}_{s_\ell, s_{\ell+1}}$ **then**
 Update p using (22) with $i = s_\ell$ and $j = s_{\ell+1}$.
 Update $k^{[s_\ell]}$ and $k^{[s_{\ell+1}]}$ using (12).
 else
 $\ell \leftarrow \ell + 1$
 end if
end loop

6. Simulations and Experimental Results

In order to demonstrate the efficacy of the proposed control scheme, we have conducted various simulation and experimental studies, the results of which are presented in this section. The algorithm that computes the harmonic transformation and its Jacobian was implemented in C++, while the proposed control protocols were implemented in Python. Code implementations can be accessed at <https://github.com/maxchaos/hntf2d> (accessed on 16 April 2023). All simulations were carried out on a PC with an Intel i5 processor operating at 2.2 GHz, with 4 GB RAM and running a GNU/Linux operating system. For more details regarding both simulations and experiments, the reader may refer to the accompanying video material at <https://youtu.be/I6WUS81iDh4> (accessed on 16 April 2023).

6.1. Simulations—Full Workspace Transformation

In the first case study, a single transformation of the entire $8\text{ m} \times 5\text{ m}$ workspace (see Figure 3) was constructed and the robot was instructed to navigate to various goal configurations starting from the same initial position. The initial configuration and the parameters of our controller were selected such as to better demonstrate the guaranteed collision avoidance properties of our scheme. Particularly, the initial values for the parameters of the adaptive law were selected as $k_d = 20$, $k_i = 1$ and $\bar{k}_i = 20$ for all $i \in \mathcal{I}_{10}$. The values of the remaining parameters were $\mathcal{K}_u = 100$, $w_\phi = 20$, $\mathcal{K}_k = 100$, $\alpha = 1$, $\epsilon_p = 0.025$, $\epsilon_v = 0.1$, $\gamma = 0.7$, $\epsilon_1 = 0.01$, $\epsilon_2 = 0.1$ and $\epsilon_3 = 0.1$. The goal configurations and the trajectories executed by the robot, both in the real and transformed workspace, are illustrated in Figure 4.

The simulations were conducted using the Euler method with 10 ms steps. Regarding the computational complexity of the control scheme, the construction of the harmonic transformation for this large workspace that was carried out offline once required 5.4 s to complete for a sufficient approximation of the workspace boundary with 3680 segments. Finally, the online computation of the transformation $T(p)$ and its Jacobian $J(p)$ required an average of 6.0 ms per step.

6.2. Simulations—Atlas of Harmonic Maps

In this case study, we decomposed the aforementioned workspace into separate partitions (see Figure 3) and constructed a harmonic transformation T_i for each one (we adopted the door of each room as the common boundary between neighboring partitions). The robot was initialized at the same position as the previous study and it was instructed to navigate towards the same set of individual goal configurations. The initial values selected for the parameters of the adaptive law were $k^{[i]} = N^{[i]} + 3$, $k_j^{[i]} = 1$ and $\bar{k}_j^{[i]} = k^{[i]}$ for all $j \in \mathcal{I}_{N^{[i]}}$ and $i \in \mathcal{I}_{N_A}$, where $N^{[i]}$ denotes the amount of obstacles inside the corresponding partition. All remaining control parameters were selected as in Section 6.1. The trajectories of the robot are depicted in Figure 5. The time spent to construct the corresponding harmonic transformations varied from 0.019s to 0.211s (depending on the amount of elements required for sufficiently approximating each room, ranging between 320 and 1000 segments) and was significantly much less than the full map construction of the previous case (5.4 s). Additionally, the online computation of $T_i(p)$ and $J_i(p)$ in each of these rooms required an average time between 1.0 ms and 2.2 ms per step, respectively. Finally, it should be noted that in this case, the workspace inner obstacles were mapped to points further away from the boundaries of the partitions, which is an interesting result as it alleviates possible numerical issues that may arise in the computation of the transformation near the obstacles (the condition number of the Jacobian of the transformation is improved). It should be stressed that the length of the paths in the second case was less (improvement of 0.5 m on average), owing to the fact that the robot gets closer to the workspace boundary since the individual transformations in each room obtain a better conditioned Jacobian (condition number 0.212 against 0.093) and thus are more fine than the first approach, where a transformation is built for the whole workspace.

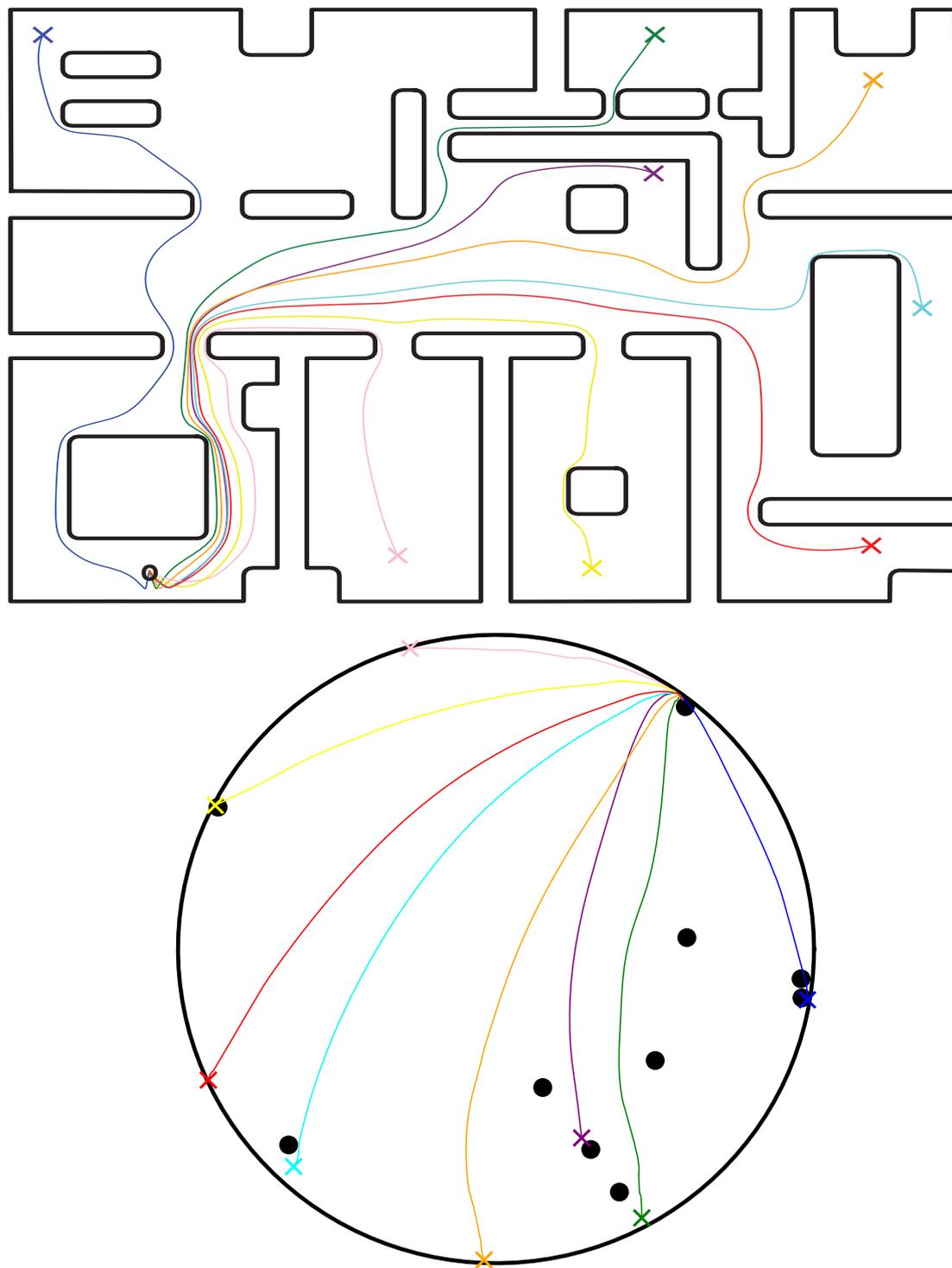


Figure 4. Robot trajectories illustrated with various colors in both the actual workspace (**upper plot**) and their image (**bottom plot**) using a full workspace transformation. The robot starts from an arbitrary location at the bottom left part of the workspace (black circle) and navigates to different goal configurations (colored crosses). Note that the ten black dots in the lower plot correspond to the points where the internal obstacles of the actual workspace have been transformed.

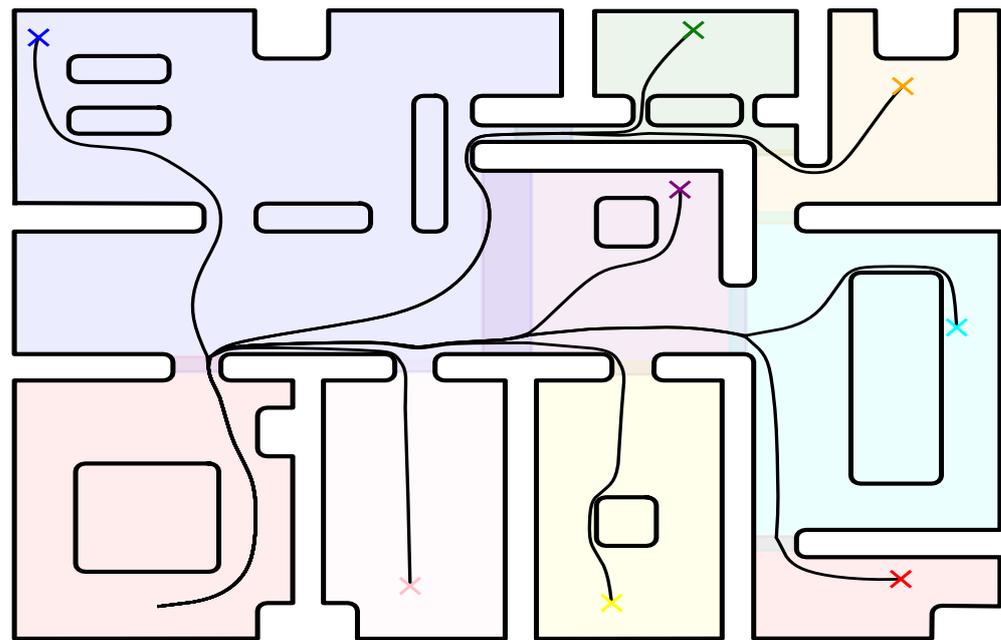


Figure 5. The resulting robot trajectories for various goal configurations depicted with colored crosses, using an atlas of the workspace.

6.3. Comparative Study—Workspace Transformation

In this subsection, we provide a comparative study of the harmonic map presented in this work against readily available workspace transformation methods employed in the motion planning literature. Particularly, we consider four $4\text{ m} \times 4\text{ m}$ compact workspaces, each associated with a pair of initial and goal positions, and construct appropriate transformations for each one by employing the methodology presented in this work (HM), as well as (i) the Star-to-Sphere Transformation (SST) [15], (ii) the Multi-Agent Navigation Transformation (MANT) [36] and (iii) the Navigation Transformation (NT) [34] (with the aforementioned Star-to-Sphere transformation serving as the underlying map). The trajectories of the robot executed while tracing the line segment connecting the initial and goal configurations in the images of each domain can be seen in Figure 6. We note that manual tuning of the compared transformations was necessary in order to render each a diffeomorphism but without making them too steep around the obstacles. Furthermore, the domain boundaries considered here had to be sufficiently smooth in order for methodologies such as MANT to be applicable. Finally, we remark that the trajectories corresponding to the proposed transformation are, in general, less abrupt compared to the rest, a property attributed to the fact that our approach is global as opposed to the other transformations, i.e., the distortion caused by each obstacle is not limited to some narrow neighborhood around it. The total length, maximum curvature and distance from the obstacles of each executed trajectory can be seen in Tables 3–5, respectively. We can see from these values that the actual trajectories yielded using harmonic maps are among the shorter and smoother ones, although they tend to approach the obstacles more than the rest.

6.4. Comparative Study—Control Law

In this subsection, we provide a comparative study of our control scheme against other motion planning methodologies.

6.4.1. APF-Based Schemes

To demonstrate the efficacy of the proposed control scheme compared to other APF-based schemes, we considered the $12\text{ m} \times 16\text{ m}$ workspace depicted in Figure 7, for which we constructed a harmonic map as described in Section 3. Next, we equipped a holonomic robot with three alternative control laws and instructed it to visit four distinct

goal positions using these controllers, starting each time from a fixed initial configuration. Particularly, we considered a conventional navigation function-based controller (NF) [15] augmented by [17], for the selection of its notorious parameter, and a harmonic navigation function-based controller (HNF) [33], in addition to our adaptive control scheme (AHNF) described in Section 4. We note that all three control laws considered here make use of the same underlying harmonic map T constructed as described above in order to drive the robot to its instructed goal positions. The trajectories executed by the robot can be seen in Figure 7. We remark that, in general, our approach steers the robot away from inner obstacles that lie between its initial and goal configurations, unlike “greedy” schemes such as the conventional NF-based controller, while keeping the traced paths shorter compared to HNFs with fixed source weights, a property attributed to the proposed adaptive laws (12) which penalize misalignment between the robot’s velocity and the direction towards the goal configuration.

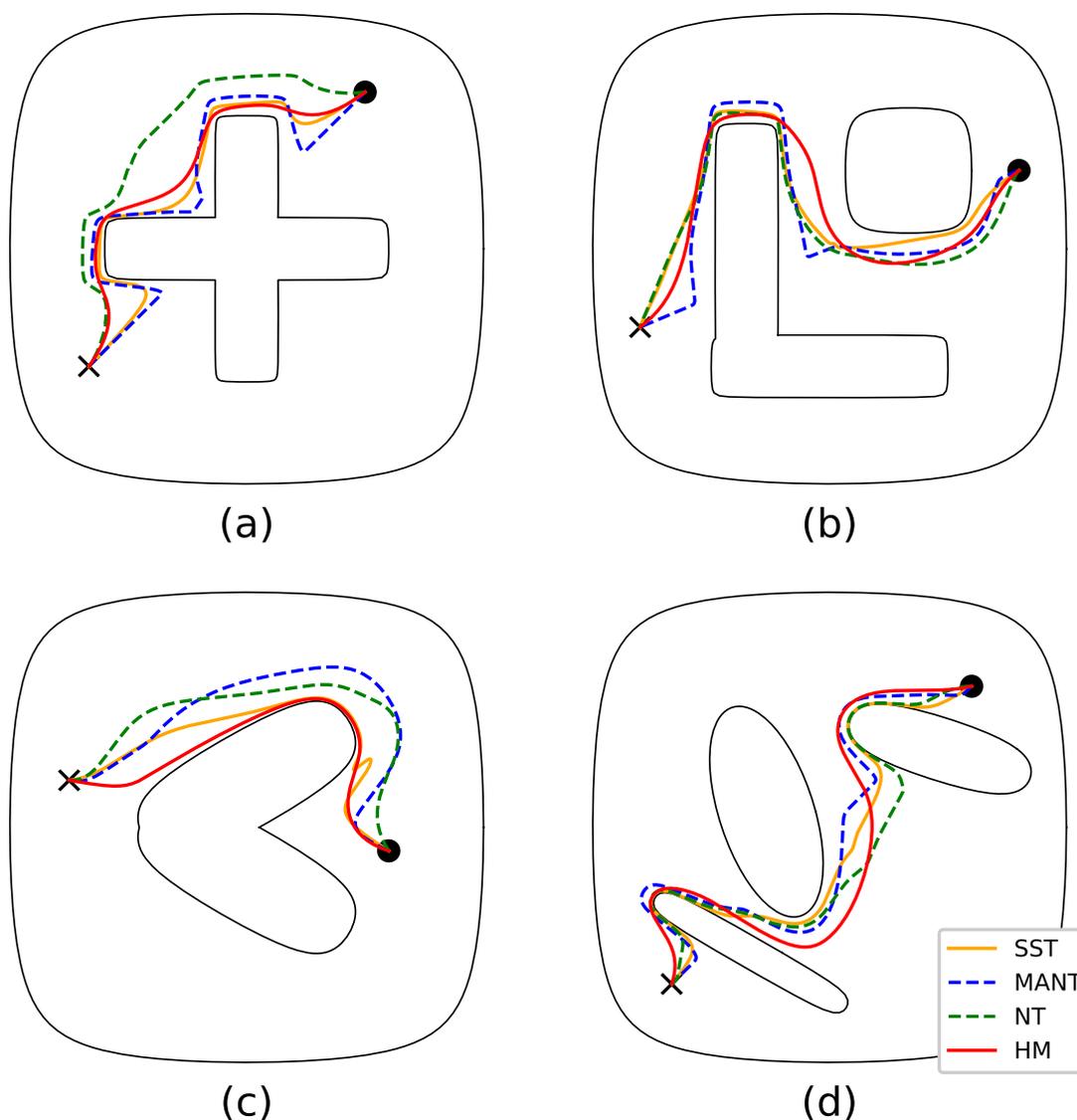


Figure 6. Robot trajectories from initial configuration (black circle) straight to the desired configuration (black cross) by employing various domain transformation methods in each workspace (a–d).

Table 3. Trajectory lengths (m) executed by employing the four alternative transformations in each workspace displayed in Figure 6.

	a	b	c	d
SST	3.63	4.18	2.12	4.09
MANT	4.26	4.69	2.34	4.45
NT	3.35	4.30	2.18	4.22
HM	3.19	4.21	2.05	4.32

Table 4. Maximum value of curvature (m^{-1}) associated with each trajectory displayed in Figure 6.

	a	b	c	d
SST	4.22	5.43	86.93	2.16
MANT	1.23	1.47	13.56	2.06
NT	66.97	25.23	14.89	6.92
HM	2.47	2.49	14.76	2.77

Table 5. Minimum distance (m) between each robot trajectory and the corresponding workspace boundaries displayed on Figure 6.

	a	b	c	d
SST	0.0303	0.0283	0.0159	0.0063
MANT	0.0644	0.1253	0.1870	0.0648
NT	0.1386	0.0506	0.0915	0.0058
HM	0.0335	0.0377	0.0103	0.0181

The total length and distance from the obstacles of each executed trajectory can be seen in Tables 6 and 7, respectively. First, we have to stress that the length trajectory corresponds to the travelled path towards the goal configuration and thus needs to be small, whereas the minimum distance to the workspace boundary refers to the closest point of the trajectory to the workspace boundary and thus needs to be large to have a safe trajectory. Consequently, note from Table 6 that the NF scheme yielded shorter path lengths than the proposed method in two cases (blue and yellow); nevertheless, such paths approach closer to the workspace boundary as indicated in Table 7, thus resulting in more risky paths. On the other hand, the Adaptive Harmonic Potential Field yields a good trade-off between path length and minimum distance to the boundary, since it achieves the shortest paths for two cases without compromising them, as is the case with the NF. On the other hand, the HPF tend to travel around the obstacle closer to the outer workspace boundary and hence exhibit more safe trajectories but they are significantly longer than the other two schemes.

6.4.2. Sampling-Based Scheme

To compare the control scheme proposed in this work against sampling-based methods, we considered a holonomic point-sized robot positioned inside a 6 m × 8 m compact workspace and a desired goal configuration. To complete this task, we employed two different controllers, namely the one proposed in this work and an admissible planner based on an improved probabilistic roadmap method (PRM) [6]. The trajectories executed by the robot using our control law as well as two of the trajectories generated by the PRM-based planner can be seen in Figure 8. The construction of the associated transformation took 31 s to complete for a given boundary approximation made of 7842 elements, whereas the PRM-based planner required approximately 24 s on average over 10 successful runs to yield a solution (we have to stress that we ran 14 trials to get 10 solutions, since four runs did not complete as they exceeded the 500 s calculation time), using the same boundary approximation for collision checking. The robot trajectories exhibited similar lengths in both algorithms (22.5 m for our method against 21.8 m on average), although no path optimization was employed in our case. Additionally, the proposed scheme resulted in

a smoother robot trajectory (based on the resulting sequence of points in both cases, we calculated the minimum curvature radius as 0.23 m for our method against 0.12 m on average for the PRM method). On the other hand, note that our approach solves the motion planning problem for any pair of initial and final configurations within the workspace, whereas the sampling-based scheme considers only one go-to problem. Thus, a different initial or final configuration would require a new solution with the PRM method. On the contrary, the proposed transformation needs to be calculated only once to solve the motion planning problem for any pair of initial and final configurations. Finally, it should be noted that for a narrower corridor in Figure 8, the sampling-based approach failed to derive a solution with a reasonable execution time (no solution was calculated within 500 s), since the probability of sampling connected points within this snaky passage reduces drastically. On the contrary, the proposed transformation took 38 s to complete for the same number of elements (i.e., 7842 elements).

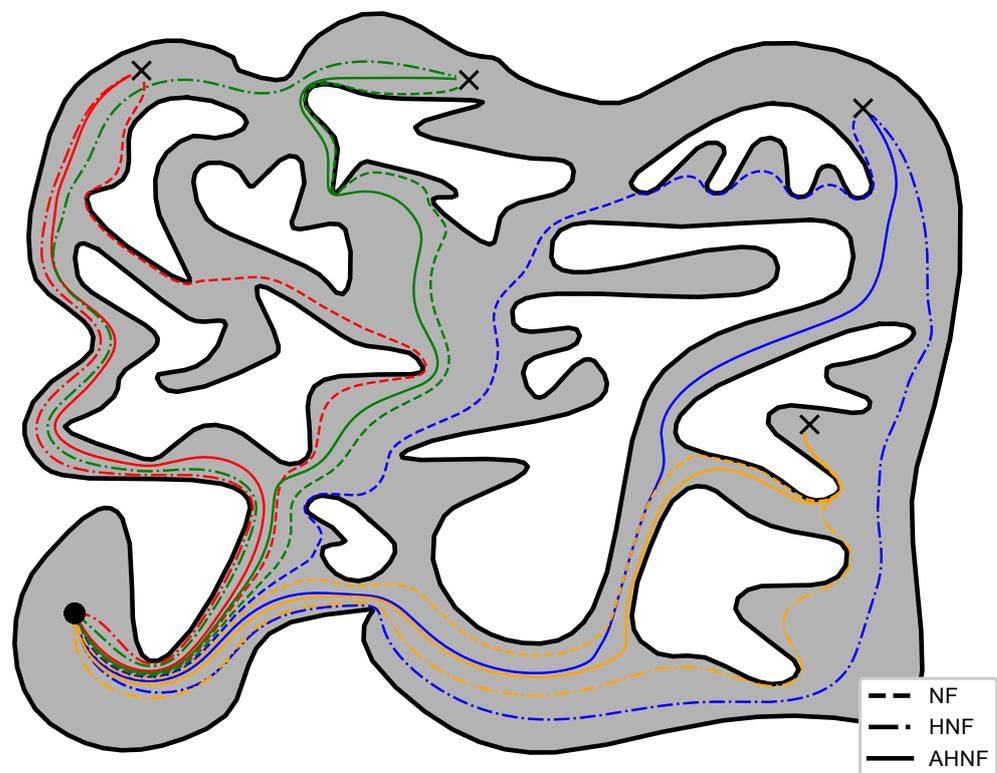


Figure 7. Trajectories of the robot navigating to four distinct goal configurations (black crosses) with red, green, yellow and blue color starting from the same initial position (black circle) while using various alternative APF-based controllers.

Table 6. Length of trajectories (m) executed by each controller (Rimon–Koditchev Navigation Functions, Harmonic Navigation Functions and Adaptive Harmonic Potential Fields) in Figure 7.

	Red	Green	Blue	Yellow
NF	19.781	20.427	22.090	18.397
HNF	18.224	22.538	26.959	20.062
AHNF	17.874	19.419	23.364	18.595

Table 7. Minimum distance (m) between the corresponding workspace boundaries and each trajectory displayed in Figure 7.

	Red	Green	Blue	Yellow
NF	0.1158	0.0102	0.1210	0.1103
HNF	0.3347	0.2135	0.2591	0.2166
AHPF	0.1310	0.0352	0.2043	0.1854

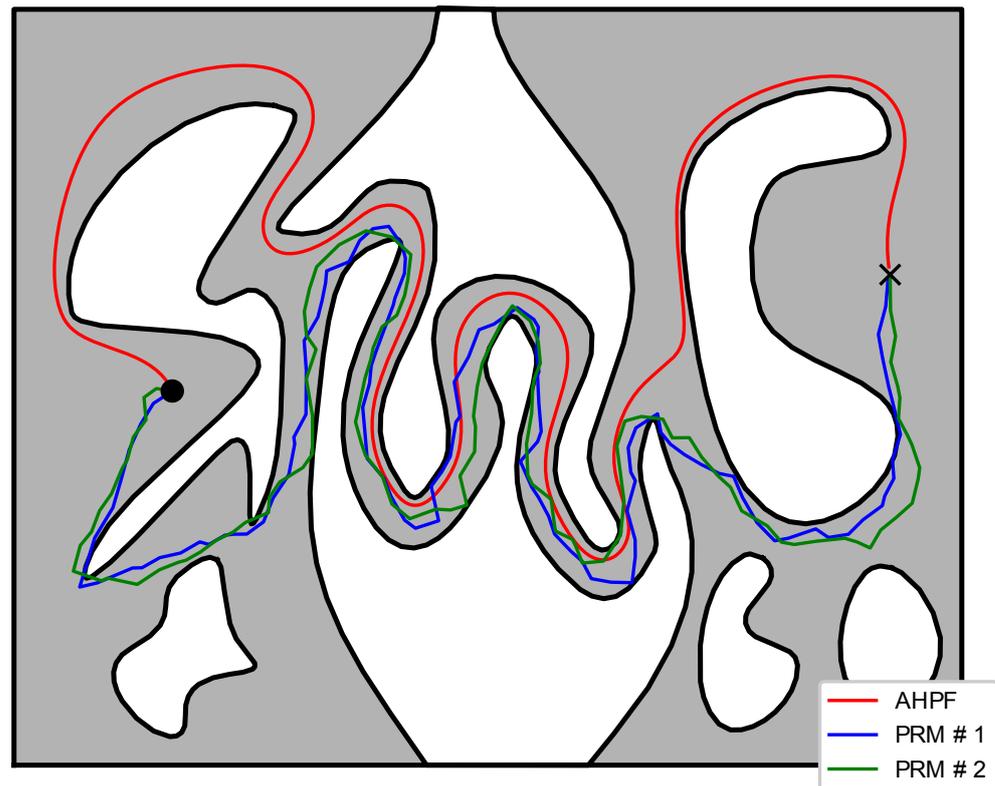


Figure 8. Trajectories of the robot navigating to its goal configuration (black cross) generated using the proposed control law and a PRM-based planner.

6.5. Experiments

In order to verify the results presented in Section 5.1, real experiments were conducted on a non-holonomic robotic platform (Robotnik Summit-XL) operating within the 10 m × 25 m compact workspace that is depicted in Figures 9 and 10. The boundaries of the workspace were obtained using readily available SLAM algorithms and were later augmented with the robot's shape (approximated by a disk). The workspace was partitioned into six overlapping subsets and the robot was instructed to visit three different goal configurations, each located in a different room. An off-the-shelf localization algorithm was employed for estimating the robot's position and orientation using its on-board sensors (laser scanners and RGB-D cameras), providing feedback at approximately 5 Hz to the robot's linear and angular commanded velocities. The construction of the associated transformations over the six subsets of the workspace took from 1.3 s for the simple and smaller partitions with 800 elements to 3.1 s for the more complex ones employing 1500 elements. On the other hand, the evaluation of the mapping as well as its Jacobian took less than 6 ms on average, which was satisfactory given the low position update rate. Note that our algorithm successfully managed to drive the robot safely (the minimum distance to the workspace boundary was 0.15 m when passing through the doors) to its specified goal configurations, as one can verify from the trajectories (see Figure 9, Figure 11 and the accompanying video material). However, an issue that needs to be pointed out is the

oscillating behavior that the robot exhibited in the configuration space's image, particularly in subsets p1 and p2 as depicted in Figure 11. Such behavior is attributed both to (a) the relative slow update of the robot's pose estimation and (b) the inversion of the Jacobian which is ill-conditioned close to extremely narrow passages of the domain. Nevertheless, such shortcomings can be alleviated by a better choice of partitions, e.g., by partitioning the domain into more subsets with less complex shapes. As a future research direction, we shall investigate whether the condition number of the Jacobian of the transformation is a fine criterion, since the condition number is usually used to measure how sensitive a function is to changes or errors in the input, and the output error results from an error in the input via the Jacobian.

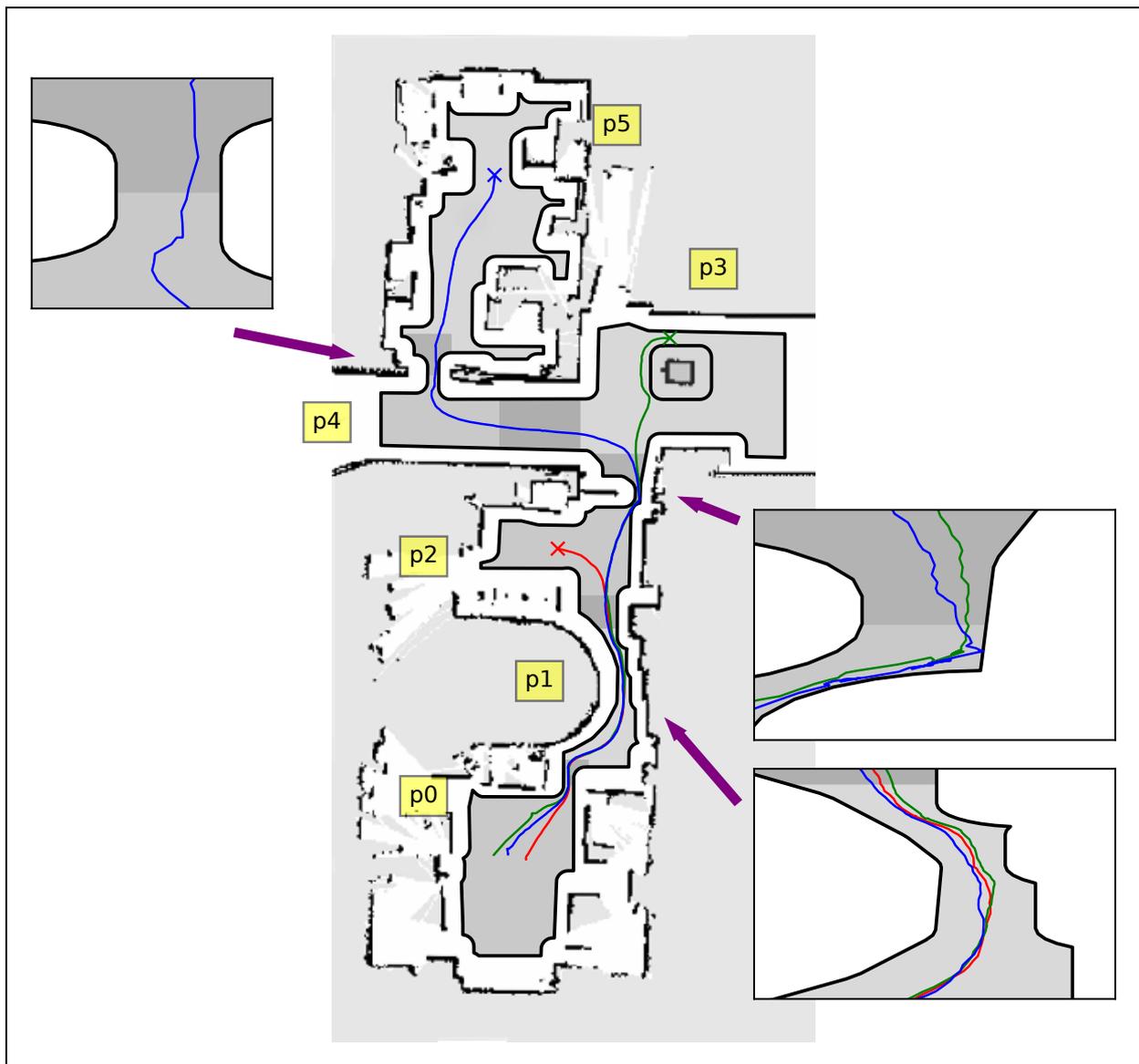


Figure 9. Trajectories of the unicycle-like robot in the real workspace, with each color (red, green and blue) corresponding to a different goal configuration. Dark gray regions indicate areas where partitions overlap.

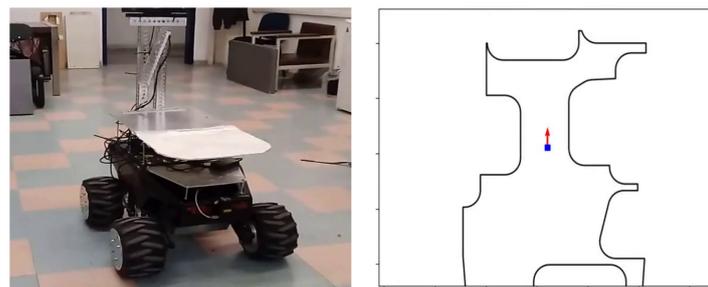
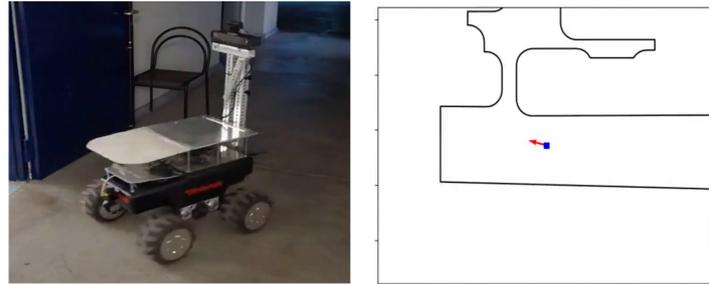
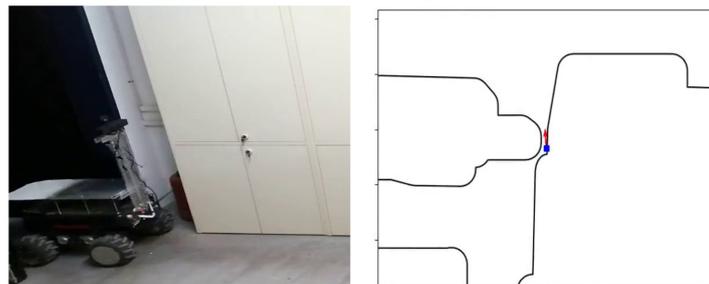
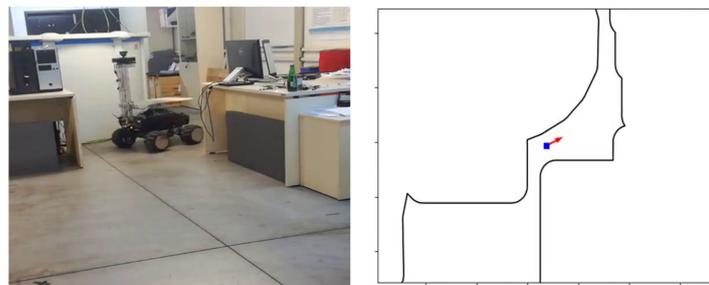
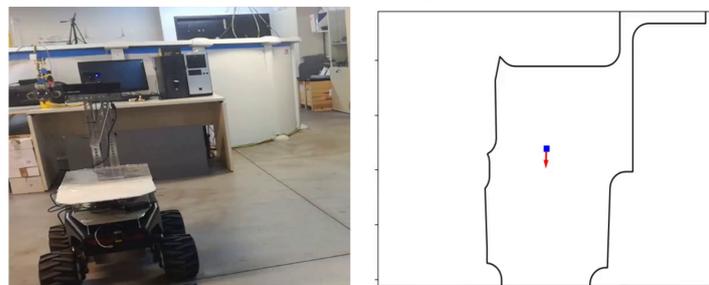
Partition p_5 - Final ConfigurationPartition p_4 Partition p_2 Partition p_1 Partition p_0 - Initial Configuration

Figure 10. Snapshots of the robot navigating through the workspace. The figures on the right illustrate the position and orientation of the robot with respect to the workspace map.

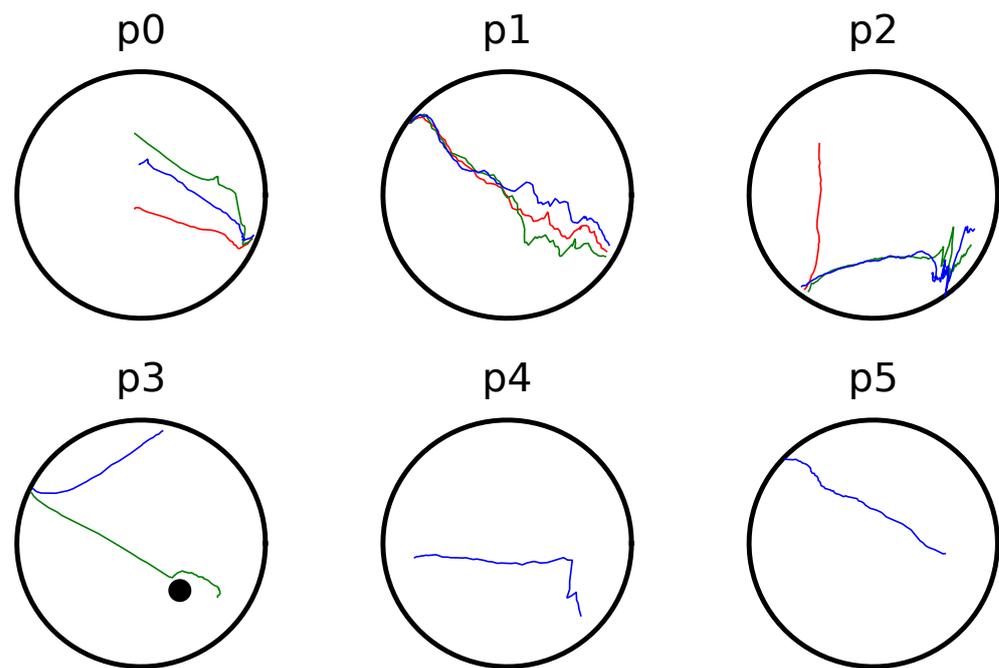


Figure 11. Robot image trajectories within each partition of the workspace for three experiments in red, green and blue color.

7. Conclusions and Future Work

In this work, we employed harmonic map theory to devise a transformation of complex workspaces directly to point worlds that are appropriate for robot navigation. Subsequently, we presented a novel motion planning control scheme based on closed-form harmonic potential fields equipped with appropriate adaptive laws for their parameters, which can safely navigate a robot to its goal state from almost all initial configurations. Additionally, we extended our approach to accommodate the navigation problem of non-holonomic robots and kept the numeric computations tractable for large workspaces.

Regarding future directions, our aim is first to increase the applicability of the proposed navigation framework by addressing partially known dynamic workspaces, which is far from being straightforward. To remedy the issue of calculation time in this case, we shall adopt a sensitivity analysis approach so that we do not solve the whole problem from scratch, but find how the solution deviates when a small change in the workspace occurs. In this way, we envision a reasonable calculation time (except from the first calculation) that would result in an almost real-time calculation of the transformation and thus allow us to consider even moving obstacles in dynamic environments. However, critical issues have to be studied concerning cases where the workspace changes topologically (e.g., in the case of antagonistically moving obstacles) and this results in significant changes in the transformation. In the same vein, switching in the transformation output might raise practical issues such as chattering that have to be carefully considered. Note that the aforementioned research direction could also serve as a first step towards the solution of the multi-robot motion planning problem, where for each robot all other robots should be considered as moving obstacles, operating antagonistically to achieve their goal configurations. Finally, another challenging research direction concerns the extension to 3D workspaces. Unfortunately, the harmonic maps have been studied only for 2D workspaces, since they rely heavily on complex analyses. Nevertheless, we propose to decompose the 3D motion planning problem into several 2D sub-problems, where the proposed solution works, and then combine them (e.g., decompose the motion along the z -axis and on the x - y plane).

Author Contributions: Methodology, P.V. and C.P.B.; Validation, P.V.; Formal analysis, P.V. and C.P.B.; Writing—original draft, P.V.; Writing—review & editing, C.P.B. and K.J.K.; Supervision, C.P.B. and K.J.K. All authors have read and agreed to the published version of the manuscript.

Funding: The publication fees of this manuscript have been financed by the Research Council of the University of Patras.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Appendix A.1. Proof of Proposition 1

By construction, it holds that $1 - (\tanh(\phi/w_\phi))^2 > 0$ for all $q \in \partial\mathcal{D} \setminus \{q_d\}$. The gradient of ϕ with respect to q is given by

$$\nabla_q \phi = k_d \frac{q - q_d}{\|q - q_d\|^2} - \sum_{i=1}^N k_i \frac{q - q_i}{\|q - q_i\|^2}. \quad (\text{A1})$$

Computing the inner product of $\nabla_q \phi$ and q yields:

$$\begin{aligned} (\nabla_q \phi)^T q &= k_d \frac{\|q\|^2 - q_d^T q}{\|q - q_d\|^2} - \sum_{i=1}^N k_i \frac{(q - q_i)^T q}{\|q - q_i\|^2} \\ &\geq k_d \frac{1 - q_d^T q}{\|q - q_d\|^2} - \max_i(k_i) \sum_{i=1}^N \frac{1}{\|q - q_i\|}. \end{aligned} \quad (\text{A2})$$

Given that all q_i lie within $\text{int}(\mathcal{D})$, the second term on the right-hand side of (A2) is finite for all $q \in \partial\mathcal{D}$. Similarly, the first term on the right-hand side of (A2) is positive for all $q \neq q_d$. Let $q' \in \partial\mathcal{D} \setminus \{q_d\}$. Additionally, the continuity of $(1 - q_d^T q)/\|q - q_d\|^2$ and $(1 - \tanh(\phi/w_\phi))^2/(2w_\phi)$ implies that there exists a closed neighborhood $\mathcal{F}(q')$ of q' , not containing q_d , where both are positive. Hence, selecting

$$k' = k_d \min_{q \in \mathcal{F}(q')} \left(\frac{1 - q_d^T q}{\|q - q_d\|^2} \frac{1}{\sum_{i=1}^N \frac{1}{\|q - q_i\|}} \right)$$

ensures that $(\nabla_q \phi)^T q > 0$ for all $q \in \mathcal{F}(q')$. Moreover, computing the derivative of $d = 1 - \|q\|^2$ with respect to time for all $q \in \mathcal{F}(q')$ and assuming $k_i < k', \forall i \in \mathcal{I}_N$ yields $\dot{d} = 2\mathcal{K}_u s \nabla_q \psi^T q > 0$; thus, the distance from the workspace boundary increases, which concludes the proof.

Appendix A.2. Proof of Proposition 2

Similarly to the proof of Proposition 3 in [33], we proceed by defining $\tilde{q}_d \triangleq q - q_d$, $\tilde{q}_i \triangleq q - q_i$ for all $i \in \mathcal{I}_N$. Let also $\hat{q}_d \triangleq \tilde{q}_d/\|\tilde{q}_d\|$ and $\hat{q}_i \triangleq \tilde{q}_i/\|\tilde{q}_i\|$. Accordingly, the Hessian of ϕ can be computed by:

$$\nabla_{\tilde{q}}^2 \phi = \frac{k_d}{\|\tilde{q}_d\|^2} (\mathcal{I}_2 - 2\hat{q}_d \hat{q}_d^T) - \sum_{i \in \mathcal{I}_N} \frac{k_i}{\|\tilde{q}_i\|^2} (\mathcal{I}_2 - 2\hat{q}_i \hat{q}_i^T). \quad (\text{A3})$$

Note that at a critical point of ϕ it holds that:

$$\begin{aligned} k_d \frac{\hat{q}_d}{\|\tilde{q}_d\|} &= \sum_{i \in \mathcal{I}_N} k_i \frac{\hat{q}_i}{\|\tilde{q}_i\|} \implies \\ \frac{k_d^2}{\|\tilde{q}_d\|^2} \hat{q}_d \hat{q}_d^T &= \sum_{i \in \mathcal{I}_N} \frac{k_i^2}{\|\tilde{q}_i\|^2} \hat{q}_i \hat{q}_i^T + \\ &\quad \sum_{i \in \mathcal{I}_N} \sum_{j \in \mathcal{I}_N \setminus \{i\}} \frac{k_i k_j}{\|\tilde{q}_i\| \|\tilde{q}_j\|} (\hat{q}_i \hat{q}_j^T + \hat{q}_j \hat{q}_i^T). \end{aligned} \quad (\text{A4})$$

Substituting (A4) into (A3) and re-arranging the terms yields:

$$\begin{aligned} \nabla_p^2 \phi &= \left(\frac{k_d}{\|\tilde{q}_d\|^2} - \sum_{i \in \mathcal{I}_N} \frac{k_i}{\|\tilde{q}_i\|^2} \right) \mathcal{I} + 2 \left(\sum_{i \in \mathcal{I}_N} \frac{k_i(k_d - k_i)}{k_d} \frac{1}{\|\tilde{q}_i\|^2} \hat{q}_i \hat{q}_i^T - \right. \\ &\quad \left. \frac{1}{k_d} \sum_{i \in \mathcal{I}_N} \sum_{j \in \mathcal{I}_N \setminus \{i\}} \frac{k_i k_j}{\|\tilde{q}_i\| \|\tilde{q}_j\|} (\hat{q}_i \hat{q}_j^T + \hat{q}_j \hat{q}_i^T) \right). \end{aligned}$$

Next, we argue that for any given set of radii $\rho_i > 0$ such that $\mathcal{D}_{\rho_i}(q_i)$, $i \in \mathcal{I}_N$ are disjoint disks that lie entirely within our domain, there exists $k'_d > 0$ such that no critical point of ϕ exists within $\mathcal{D} \setminus \bigcup_{i \in \mathcal{I}_N} \mathcal{D}_{\rho_i}(q_i)$ for all $k_d > k'_d$. This implies that, by choosing a sufficiently large k_d , each critical point of ϕ belongs to a single $\mathcal{D}_{\rho_i}(q_i)$. Let q^* be a critical point and $\ell = \operatorname{argmin}_{i \in \mathcal{I}_N} \|q^* - q_i\|$. To show that $\nabla_q^2 \phi(q^*)$ is not degenerate, it suffices to show that its eigenvalue $\lambda(q^*)$ is positive. We recall that λ is lower bounded by the quadratic form $\hat{x}^T \nabla_q^2 \phi \hat{x}$ for all $\|\hat{x}\| = 1$. By considering the direction of \tilde{q}_ℓ and after some tedious calculations, we obtain:

$$\begin{aligned} \hat{q}_\ell^T \nabla_q^2 \phi(q^*) \hat{q}_\ell &= \frac{k_d}{\|\tilde{q}_d\|^2} - \sum_{i \in \mathcal{I}_N \setminus \{\ell\}} \frac{k_i}{\|\tilde{q}_i\|^2} \\ &\quad + \frac{k_\ell}{\|\tilde{q}_\ell\|^2} \left(\frac{k_d - 2k_\ell}{k_d} - 4 \frac{\|\tilde{q}_\ell\|}{k_d} \sum_{i \in \mathcal{I}_N \setminus \{\ell\}} \frac{k_i}{\|\tilde{q}_i\|} 2(\hat{q}_\ell^T \hat{q}_i) \right) \\ &\quad + 2 \sum_{i \in \mathcal{I}_N \setminus \{\ell\}} \frac{k_i(k_d - k_i)}{k_d} \frac{1}{\|\tilde{q}_i\|^2} (\hat{q}_\ell^T \hat{q}_i)^2 \\ &\quad - \frac{2}{k_d} \sum_{i \in \mathcal{I}_N \setminus \{\ell\}} \sum_{j \in \mathcal{I}_N \setminus \{i, \ell\}} \frac{k_i k_j}{\|\tilde{q}_i\| \|\tilde{q}_j\|} 2(\hat{q}_\ell^T \hat{q}_i)(\hat{q}_\ell^T \hat{q}_j). \end{aligned} \quad (\text{A5})$$

The first right-hand side term of (A5) is strictly positive. Since all k_i are bounded and non-negative, choosing a sufficiently large k_d renders the second and third right-hand side terms non-negative. Furthermore, note that the fourth and fifth right-hand side terms are bounded for all $q^* \in \mathcal{D}_{\rho_\ell}(q_\ell)$. Thus, by choosing a sufficiently large k_d , the first three terms of (A5) can be made dominant, thus rendering $\hat{q}_\ell^T \nabla_q^2 \phi \hat{q}_\ell$ positive at q^* , which concludes the proof.

Appendix A.3. Proof of Proposition 3

Firstly, we will show that the robot cannot escape through the workspace's outer boundary. Let us assume that $q \rightarrow q' \in \partial \mathcal{D} \setminus \{q_d\}$. Then, $\dot{q} \rightarrow 0$ by virtue of (7), since $s(q, k) = 0$ for all $\|q\| = 1$ with $(\nabla_q \phi)^T q \leq 0$. Additionally, $w_0 \rightarrow 1$ and $w_i \rightarrow 0$, for all $i \in \mathcal{I}_N$. Thus, $\dot{k}_i < 0$ holds within a neighborhood of $\partial \mathcal{D}$, while $k_i > 0$, which implies that $k_i \rightarrow 0$ for all $i \in \mathcal{I}_N$. Moreover, Proposition 1 dictates that there exists $k' > 0$ for which any point in $\partial \mathcal{D} \setminus \{q_d\}$ is repulsive under ψ . Since (12) dictates that all k_i become less than k' in finite time, this contradicts our supposition.

Next, we consider collision avoidance between the robot and the inner obstacles. Let us assume that the robot approaches obstacle i . By construction, $w_i \rightarrow 1$ while $\nabla_q \psi \rightarrow 0$ and $\bar{w}_j \rightarrow 0$ for all $j \in \mathcal{I}_N^* \setminus \{i\}$. Note that there exists a neighborhood \mathcal{N}_i of q_i such that $w_0 = 0$ for all $q \in \mathcal{N}_i$ due to continuity of \bar{w}_0 and $\xi_2(\bar{w}_0; \epsilon_3)$. Additionally, since the robot is assumed to approach q_i , $\dot{q}^T(q - q_i)$ cannot be identically zero inside \mathcal{N}_i . As such, as long as $k_i < \bar{k}_i$, $\dot{k}_i \geq 0$ inside \mathcal{N}_i without $\dot{k}_i = 0$ for all $q \in \mathcal{N}_i$. This implies that $k_i \nearrow 0$ as $q \rightarrow q_i$, thus rendering q_i a local maximum of ψ . Thus, there exists a neighborhood of q_i inside which $(\nabla_q \psi)^T(q - q_i) > 0$, which contradicts our assumption.

Appendix A.4. Proof of Proposition 4

Let $V \triangleq \psi(q, k)$, as defined in (10), be a candidate Lyapunov function, which is non-negative and vanishes only when $q = q_d$. Differentiating V along the system's trajectories yields:

$$\dot{V} = \frac{1 - \tanh(\phi/w_\phi)^2}{2w_\phi} \left((\nabla_q \phi)^T \dot{q} + \ln\left(\frac{\|q - q_d\|}{2}\right) \dot{k}_d - \sum_{i \in \mathcal{I}_N} \ln\left(\frac{\|q - q_i\|}{2}\right) \dot{k}_i \right). \quad (\text{A6})$$

Given that $\dot{q} = J\dot{p}$, the first term of (A6) can be further expanded as follows:

$$(\nabla_q \phi)^T \dot{q} = -\mathcal{K}_{us} \frac{1 - \tanh(\phi/w_\phi)^2}{2w_\phi} \|J^T \nabla_q \phi\|^2, \quad (\text{A7})$$

which is non-positive for all $q \in \Omega$ and becomes zero only at the critical points of ψ . The second term of (A6) is non-positive since $\dot{k}_d \geq 0$ by construction and invariance of \mathcal{W} (see Proposition 3) implies $\|q - q_d\| \leq 2$ which, in turn, implies $\ln\left(\frac{\|q - q_d\|}{2}\right) \leq 0$. Similarly, the sign of each term of the sum is determined solely by the sign of the corresponding \dot{k}_i . Substituting (12) yields:

$$- \sum_{i \in \mathcal{I}_N} \ln\left(\frac{\|q - q_i\|}{2}\right) \dot{k}_i \leq - \sum_{i \in \mathcal{I}_N} \ln\left(\frac{\|q - q_i\|}{2}\right) (\bar{k}_i - k_i) w_i \ell_i g_i. \quad (\text{A8})$$

Given that $g_i \leq \|\nabla_q \psi\|^2$ and $\sum_{i \in \mathcal{I}_N} \bar{k}_i w_i \leq 1$ by construction, expanding ℓ_i into the right-hand side of (A8) leads to:

$$\begin{aligned} - \sum_{i \in \mathcal{I}_N} \ln\left(\frac{\|q - q_i\|}{2}\right) (\bar{k}_i - k_i) w_i \ell_i g_i &\leq \mathcal{K}_{us} \|\nabla_q \psi\|^2 \sum_{i \in \mathcal{I}_N} (\bar{k}_i - k_i) w_i \\ &\leq \mathcal{K}_{us} \frac{1 - \tanh(\phi/w_\phi)^2}{2w_\phi} \|\nabla_q \phi\|^2. \end{aligned} \quad (\text{A9})$$

Thus, (A6) is non-positive. Therefore, by invoking Lyapunov's Stability Theorem (Theorem 3.1 [41]) we may conclude that q_d is stable. Finally, LaSalle's Theorem (Theorem 3.4 [41]) dictates that the system will converge to the largest invariant set, which, in our case, consists of the critical points of ψ , thus concluding the proof.

Appendix A.5. Proof of Proposition 5

At the critical point z^* of system (17), the Hessian $\nabla_q^2 \phi$ of ϕ is non-degenerate, since otherwise $\dot{k}_d \neq 0$. Additionally, $q^* \in \Omega \setminus \{q_d\}$ implies that $1 - (\tanh(\phi/w_\phi))^2 \neq 0$. These two facts mean that $\nabla_q^2 \psi$ has two eigenvalues at z^* , namely λ and $-\lambda$, with $\lambda > 0$. Computing the Jacobian of f_q with respect to q at z^* yields:

$$\begin{aligned} \nabla_q f_q &= -\mathcal{K}_u (\nabla_q \psi)^T \nabla_q s - \mathcal{K}_{us} \nabla_q^2 \psi \\ &= -\mathcal{K}_{us} \nabla_q^2 \psi \end{aligned}$$

since $\nabla_q \psi(q^*) = 0$. Furthermore, by construction of the adaptive law (12), the Jacobian of f_k with respect to z at z^* is $\mathbf{0}_{(1+N) \times (3+N)}$. Thus, linearization of the system f_z at z^* yields

$$\nabla_z f_z(z^*) = -\mathcal{K}_{us} \frac{1 - \tanh(\phi/w_\phi)^2}{2w_\phi} \begin{bmatrix} \nabla_q^2 \phi & \frac{\partial \phi}{\partial k_d} & \frac{\partial \phi}{\partial k_1} & \dots & \frac{\partial \phi}{\partial k_N} \\ \mathbf{0}_{(1+N) \times (3+N)} \end{bmatrix}.$$

Since the top-left block $\nabla_q^2 \phi$ is invertible at z^* , using the well-known property of block matrix determinants, we can see that $\nabla_{z^*} f_z$ has two non-zero eigenvalues, particularly the eigenvalues of $\nabla_q^2 \psi$ and a zero eigenvalue with multiplicity $1 + N$. Thus, $\nabla_z f_z(z^*)$ has exactly one positive eigenvalue, rendering z^* a saddle point of (17) (Theorem 3.7 [41]).

Appendix A.6. Proof of Theorem 1

In Proposition 4, we have proven that $\dot{\psi} < 0$ for all $q \in \Omega \setminus \{q_d\}$, except for the critical points of ϕ that lie in it. Lasalle’s Invariance Theorem (Theorem 3.4 [41]) dictates that system (17) will converge to either (a) the desired configuration q_d , (b) the obstacles q_i or (c) a critical point $z^* = (q^*, k^*)$ with $q^* \in \Omega \setminus \{q_d\}$. We know from Proposition 3 that the critical points of case (b) are repulsive; therefore, no trajectory of the system may converge to them. Regarding the critical point z^* corresponding to case (c), Proposition 5 dictates that it must be a non-isolated, degenerate equilibrium of the whole of system (17), since $\nabla_z f_z$ has one positive, one negative and several zero eigenvalues. Let \bar{k}_d be the upper bound of k_d that the closed-loop system can possibly attain, as indicated by Proposition 2. In order to prove that the set of initial conditions leading to these points has zero Lebesgue measure, we will study the properties of the gradient-like system (by definition, a gradient-like system is a pair of a scalar cost functions and a dynamical system for which each non-equilibrium initial condition moves the state towards a new one whose cost is less than that of the initial state) $(\psi(z), F_{z,\tau}(z))$ in the domain \mathcal{S}_z , where the scalar potential $\psi(z)$ is treated as a function to be minimized, the map $F_{z,\tau}(z) : \mathcal{S}_z \mapsto \mathbb{R}^{N+3}$ is given by

$$F_{z,\tau}(z(t)) \triangleq z(t + \tau) = z(t) + \int_t^{t+\tau} f_z(z(s)) ds$$

for any $\tau > 0$ and $\mathcal{S}_z \triangleq \mathcal{D} \times [1, \bar{k}_d] \times [0, \bar{k}_1] \times \dots \times [0, \bar{k}_N]$. Note that \mathcal{S}_z is convex and closed. Additionally, the map $F_{z,\tau}(z)$ is a locally Lipschitz diffeomorphism in \mathcal{S}_z and \mathcal{S}_z is forward invariant under $F_{z,\tau}(z)$ (by virtue of Proposition 3 and design of adaptive law (12)) for all $\tau > 0$. Furthermore, the unwanted equilibria of $F_{z,\tau}$ are strict saddles. Thus, following similar arguments as the proof of Theorem 3 in [42], we conclude that the set of all initial conditions that converge to these saddles has zero Lebesgue measure, which implies that almost every trajectory of the system converges to q_d , i.e., the only stable equilibrium of (17), thus completing the proof.

Appendix A.7. Proof of Theorem 2

We begin by noting that, by virtue of (21), we only need to study the trajectories of (19) in the workspace’s image, since that motion is traced exactly by our robot. Considering the first part of the Theorem 2, we note that by following the same arguments as in the proof of Proposition 3, we may conclude that the robot cannot escape through the workspace’s outer boundary. Likewise, assuming that $q \rightarrow q_i$ for some $i \in \mathcal{I}_N$ implies that $(n_\theta^T J^T \nabla_q \psi)^T n_{\hat{\theta}}^T (q - q_i)$ cannot be identically zero in a neighborhood of q_i . As such, since $\dot{k}_i \geq 0$ in the neighborhood of q_i , k_i cannot vanish as the robot approaches q_i , which contradicts our original supposition.

To prove the second part of the Theorem 2, first we show that the only equilibria of the closed-loop system coincide with the critical points of ψ . Assuming that $s_v \neq 0$, it is readily seen that both inner products in (20) vanish simultaneously only when $\nabla_q \psi = 0$. Considering now the case when $s_v = 0$, we note that this can only happen when $q \in \partial \mathcal{D}$ and $n_{\hat{\theta}}$ is tangent to $\partial \mathcal{D}$. For $\dot{\omega}$ to also vanish when $s_v = 0$, the gradient $\nabla_q \psi$ should also be

tangent to $\partial\mathcal{D}$. Recalling that the adaptive laws for k ensure that $\nabla_q\psi$ will eventually point inwards, we conclude that no equilibria other than the critical points of ψ exist.

Next, we consider ψ as a Lyapunov candidate function, whose derivative along the systems trajectories is given by (A6) (note that ψ does not depend on θ). Substituting (20) into the first term of (A6) yields:

$$(\nabla_q\phi)^T \dot{q} = -\mathcal{K}_{vs_v} \frac{1 - \tanh(\phi/w_\phi)^2}{2w_\phi} \left((n_{\hat{\theta}})^T \nabla_q\phi \right)^2. \quad (\text{A10})$$

Regarding the remaining terms of (A6), given that $g_{v,i} \leq (n_{\hat{\theta}}^T \nabla_q\psi)^2$, one can readily verify that:

$$- \sum_{i \in \mathcal{I}_N} \ln\left(\frac{\|q - q_i\|}{2}\right) \dot{k}_i \leq -\mathcal{K}_{vs_v} \frac{1 - \tanh(\phi/w_\phi)^2}{2w_\phi} \left((n_{\hat{\theta}})^T \nabla_q\phi \right)^2. \quad (\text{A11})$$

Thus, invoking Lyapunov's Stability Theorem (Theorem 3.1 [41]) and LaSalle's Theorem (Theorem 3.4 [41]) concludes the proof similarly to Proposition 4.

Appendix A.8. Proof of Theorem 3

Regarding the robot's safety under the closed-loop system, we note that when $p \in \mathcal{P}_{s_\ell} \setminus \mathcal{P}_{s_{\ell+1}}$ or $p \in \mathcal{P}_{s_n}$ for all $\ell < n$, the individual control laws render every point on the corresponding partition boundaries repulsive. When $p \in \mathcal{C}_{s_\ell, s_{\ell+1}}$, we note that, by construction, both $u^{[s_\ell]}$ and $u^{[s_{\ell+1}]}$ vanish when the robot approaches any point of $\partial\mathcal{P}_{s_\ell} \cap \partial\mathcal{P}_{s_{\ell+1}}$, preventing the robot from escaping. Additionally, the adaptive laws of each individual potential field will eventually render both $\nabla_p\psi_{s_\ell}$ and $\nabla_p\psi_{s_{\ell+1}}$ inward-looking with respect to \mathcal{W} , rendering $\partial\mathcal{P}_{s_\ell} \cap \partial\mathcal{P}_{s_{\ell+1}}$ repulsive.

While $p \in \mathcal{P}_{s_\ell}$, we consider $V \triangleq \psi_{s_\ell}$ as a Lyapunov function candidate and we examine its time derivative along the system's trajectories when $p \in \mathcal{C}_{s_\ell, s_{\ell+1}}$:

$$\begin{aligned} \dot{V} &= (\nabla_p\psi_{s_\ell})^T \dot{p} + (\nabla_{k^{[s_\ell]}}\psi_{s_\ell})^T \dot{k}^{[s_\ell]} \\ &= (\nabla_p\psi_{s_\ell})^T \left(u^{[s_\ell]} + \eta_{c, s_\ell, s_{\ell+1}} \eta_{t, s_\ell, s_{\ell+1}} u^{[s_{\ell+1}]} \right) + (\nabla_{k^{[s_\ell]}}\psi_{s_\ell})^T k^{[s_\ell]} \\ &= (\nabla_p\psi_{s_\ell})^T u^{[s_\ell]} + (\nabla_{k^{[s_\ell]}}\psi_{s_\ell})^T k^{[s_\ell]} + \eta_{c, s_\ell, s_{\ell+1}} \eta_{t, s_\ell, s_{\ell+1}} (\nabla_p\psi_{s_\ell})^T u^{[s_{\ell+1}]}. \end{aligned} \quad (\text{A12})$$

We recall that the first two right-hand side terms of (A12) are non-positive, as shown in Proposition 4. Likewise, the last term is rendered non-positive by virtue of $\eta_{c, s_\ell, s_{\ell+1}}$. Additionally, we note that the equilibrium points of the system in $p \in \mathcal{C}_{s_\ell, s_{\ell+1}}$ correspond only to critical points of ψ_{s_ℓ} . By virtue of $\eta_{c, s_\ell, s_{\ell+1}}$, which vanishes at a critical point of ψ_{s_ℓ} , along with its derivative, one can easily verify that the Jacobian of (22) is equal to the one of $u^{[s_\ell]}$, whose properties were studied in Proposition 2. Finally, following a similar procedure as in the proof of Theorem 1, we conclude that the system will converge to the specified goal configuration for almost all initial configurations.

References

1. Canny, J. *The Complexity of Robot Motion Planning*; MIT Press: Cambridge, MA, USA, 1988.
2. Mesesan, G.; Roa, M.A.; Icer, E.; Althoff, M. Hierarchical Path Planner Using Workspace Decomposition and Parallel Task-Space RRTs. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Madrid, Spain, 1–5 October 2018; pp. 6524–6531.
3. Karaman, S.; Frazzoli, E. Sampling-based algorithms for optimal motion planning. *Int. J. Robot. Res.* **2011**, *30*, 846–894. [[CrossRef](#)]
4. Chi, W.; Wang, C.; Wang, J.; Meng, M.Q. Risk-DTRRT-Based Optimal Motion Planning Algorithm for Mobile Robots. *IEEE Trans. Autom. Sci. Eng.* **2019**, *16*, 1271–1288. [[CrossRef](#)]
5. Kavraki, L.E.; Svestka, P.; Latombe, J.; Overmars, M.H. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robot. Autom.* **1996**, *12*, 566–580. [[CrossRef](#)]

6. Chen, G.; Luo, N.; Liu, D.; Zhao, Z.; Liang, C. Path planning for manipulators based on an improved probabilistic roadmap method. *Robot. Comput.-Integr. Manuf.* **2021**, *72*, 102196. [[CrossRef](#)]
7. Salzman, O.; Hemmer, M.; Halperin, D. On the Power of Manifold Samples in Exploring Configuration Spaces and the Dimensionality of Narrow Passages. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 529–538. [[CrossRef](#)]
8. Kingston, Z.; Moll, M.; Kavraki, L.E. Sampling-Based Methods for Motion Planning with Constraints. *Annu. Rev. Control Robot. Auton. Syst.* **2018**, *1*, 159–185. [[CrossRef](#)]
9. Ogren, P.; Leonard, N.E. A convergent dynamic window approach to obstacle avoidance. *IEEE Trans. Robot.* **2005**, *21*, 188–195. [[CrossRef](#)]
10. Wen, J.; Zhang, X.; Gao, H.; Yuan, J.; Fang, Y. E³MoP: Efficient Motion Planning Based on Heuristic-Guided Motion Primitives Pruning and Path Optimization With Sparse-Banded Structure. *IEEE Trans. Autom. Sci. Eng.* **2021**, *19*, 2762–2775. [[CrossRef](#)]
11. Bhattacharya, S.; Ghrist, R. Path Homotopy Invariants and their Application to Optimal Trajectory Planning. *Ann. Math. Artif. Intell.* **2018**, *84*, 139–160. [[CrossRef](#)]
12. Diaz-Arango, G.; Vázquez-Leal, H.; Hernandez-Martinez, L.; Pascual, M.T.S.; Sandoval-Hernandez, M. Homotopy Path Planning for Terrestrial Robots Using Spherical Algorithm. *IEEE Trans. Autom. Sci. Eng.* **2018**, *15*, 567–585. [[CrossRef](#)]
13. Khatib, O. Real-time obstacle avoidance for manipulators and mobile robots. In Proceedings of the IEEE International Conference on Robotics and Automation, St. Louis, MO, USA, 25–28 March 1985; pp. 500–505.
14. Koditschek, D. Exact robot navigation by means of potential functions: Some topological considerations. In Proceedings of the IEEE International Conference on Robotics and Automation, Raleigh, NC, USA, 31 March–3 April 1987; pp. 1–6.
15. Rimon, E.; Koditschek, D. Exact Robot Navigation Using Artificial Potential Functions. *IEEE Trans. Robot. Autom.* **1992**, *8*, 501–518. [[CrossRef](#)]
16. Loizou, S.G.; Jadbabaie, A. Density Functions for Navigation-Function-Based Systems. *IEEE Trans. Autom. Control* **2008**, *53*, 612–617. [[CrossRef](#)]
17. Filippidis, I.; Kyriakopoulos, K.J. Adjustable navigation functions for unknown sphere worlds. In Proceedings of the IEEE Conference on Decision and Control and European Control Conference, Orlando, FL, USA, 12–15 December 2011; pp. 4276–4281.
18. Filippidis, I.F.; Kyriakopoulos, K.J. Navigation Functions for everywhere partially sufficiently curved worlds. In Proceedings of the IEEE International Conference on Robotics and Automation, St Paul, MN, USA, 14–18 May 2012; pp. 2115–2120.
19. Paternain, S.; Koditschek, D.E.; Ribeiro, A. Navigation Functions for Convex Potentials in a Space With Convex Obstacles. *IEEE Trans. Autom. Control* **2018**, *63*, 2944–2959. [[CrossRef](#)]
20. Arslan, O.; Koditschek, D.E. Sensor-based reactive navigation in unknown convex sphere worlds. *Int. J. Robot. Res.* **2019**, *38*, 196–223. [[CrossRef](#)]
21. Connolly, C.; Burns, J.; Weiss, R. Path Planning Using Laplace’s Equation. In Proceedings of the IEEE International Conference on Robotics and Automation, Cincinnati, OH, USA, 13–18 May 1990; pp. 2102–2106.
22. Mantegh, I.; Jenkin, M.R.M.; Goldenberg, A.A. Path Planning for Autonomous Mobile Robots Using the Boundary Integral Equation Method. *J. Intell. Robot. Syst.* **2010**, *59*, 191–220. [[CrossRef](#)]
23. Golan, Y.; Edelman, S.; Shapiro, A.; Rimon, E. Online Robot Navigation Using Continuously Updated Artificial Temperature Gradients. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1280–1287. [[CrossRef](#)]
24. Kim, J.O.; Khosla, P.K. Real-time obstacle avoidance using harmonic potential functions. *IEEE Trans. Robot. Autom.* **1992**, *8*, 338–349. [[CrossRef](#)]
25. Merheb, A.; Gazi, V.; Sezer-Uzol, N. Implementation Studies of Robot Swarm Navigation Using Potential Functions and Panel Methods. *IEEE/ASME Trans. Mechatron.* **2016**, *21*, 2556–2567. [[CrossRef](#)]
26. Velagić, J.; Vuković, L.; Ibrahimović, B. Mobile Robot Motion Framework Based on Enhanced Robust Panel Method. *Int. J. Control Autom. Syst.* **2020**, *18*, 1264–1276. [[CrossRef](#)]
27. Feder, H.J.S.; Slotine, J.J.E. Real-time path planning using harmonic potentials in dynamic environments. In Proceedings of the IEEE International Conference on Robotics and Automation, Albuquerque, NM, USA, 20–25 April 1997; pp. 874–881.
28. Marchidan, A.; Bakolas, E. A local reactive steering law for 2D collision avoidance with curvature constraints and constant speed. *Robot. Auton. Syst.* **2022**, *155*, 104156. [[CrossRef](#)]
29. e Silva, E.P.; Engel, P.M.; Trevisan, M.; Idiart, M.A. Exploration Method Using Harmonic Functions. *Robot. Auton. Syst.* **2002**, *40*, 25–42. [[CrossRef](#)]
30. Maffei, R.; Souza, M.P.; Mantelli, M.; Pittol, D.; Kolberg, M.; Jorge, V.A.M. Exploration of 3D terrains using potential fields with elevation-based local distortions. In Proceedings of the IEEE International Conference on Robotics and Automation, Paris, France, 31 May–31 August 2020; pp. 4239–4244.
31. Voruganti, H.; Dasgupta, B.; Hommel, U. A novel potential field based domain mapping method. In Proceedings of the 10th WSEAS Conference on Computers (ICCOMP06), Athens, Greece, 13 July 2006; pp. 655–661.
32. Gautam Pradeepkumar, B.; Gondegaon, S.; Voruganti, H.K. Domain Mapping for Path Planning and Mesh Generation. In Proceedings of the International Conference on Theoretical, Applied, Computational and Experimental Mechanics, Kharagpur, India, 29–31 December 2014.
33. Loizou, S.G. Closed form Navigation Functions based on harmonic potentials. In Proceedings of the IEEE Conference on Decision and Control and European Control Conference, Orlando, FL, USA, 12–15 December 2011; pp. 6361–6366.
34. Loizou, S.G. The Navigation Transformation. *IEEE Trans. Robot.* **2017**, *33*, 1516–1523. [[CrossRef](#)]

35. Loizou, S.G. Navigation functions in topologically complex 3-D workspaces. In Proceedings of the American Control Conference, Montreal, QC, Canada, 27–29 June 2012; pp. 4861–4866.
36. Loizou, S.G. The Multi-Agent Navigation Transformation: Tuning-Free Multi-Robot Navigation. In Proceedings of the Robotics: Science and Systems, Berkeley, CA, USA, 14–16 July 2014; pp. 1–9.
37. Vlantis, P.; Vrohidis, C.; Bechlioulis, C.P.; Kyriakopoulos, K.J. Robot Navigation in Complex Workspaces Using Harmonic Maps. In Proceedings of the IEEE International Conference on Robotics and Automation, Brisbane, Australia, 21–25 May 2018; pp. 1726–1731.
38. Duren, P.; Hengartner, W. Harmonic mappings of multiply connected domains. *Pac. J. Math.* **1997**, *180*, 201–220. [[CrossRef](#)]
39. Biringen, S.; Chow, C.Y. *An Introduction to Computational Fluid Mechanics by Example*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2011.
40. Kuethe, A.M.; Chow, C.Y.; Fung, Y.C. *Foundations of Aerodynamics: Bases of Aerodynamics Design*, 5th ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 1997.
41. Khalil, H.K. *Nonlinear Systems*; Prentice-Hall: Hoboken, NJ, USA, 1996.
42. Panageas, I.; Piliouras, G. Gradient Descent Only Converges to Minimizers: Non-Isolated Critical Points and Invariant Regions. *arXiv* **2016**, arXiv:1605.00405.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.