State-Feedback Optimal Motion Planning in the Presence of Obstacles

Panagiotis Rousseas, *Graduate Student Member, IEEE*, Charalampos P. Bechlioulis, *Member, IEEE*, and Kostas J. Kyriakopoulos, *Fellow, IEEE*

***Abstract*—In this letter, a solution to the kinematic optimal motion planning problem is presented, where a previous nearly globally optimal approach is extended to workspaces with internal obstacles. The method is inspired by fundamental properties of velocity fields in the presence of obstacles, where topological restrictions inhibit naive approaches. The topological perplexity problem presents itself as a challenging issue for optimal control, even for low-dimensional cases with simple dynamics. Our scheme is formulated such that a locally optimal workspace decomposition enables extracting a close-to-optimal solution. Several synthetic workspace examples are demonstrated, along with comparisons against existing optimal approaches, where our scheme is superior w.r.t. both cost value and execution time.**

***Index Terms*—Motion and path planning, optimization and optimal control.**

I. Introduction

THE motion planning problem is one of the most fundamental problems in Robotics. The "embodied intelligence" aspect of robotic systems [1] at the core of the field necessitates operation in the real world and thus, safe planning is a prerequisite in almost all applications. Hence, significant efforts have been allocated to solving the path and motion planning problems since the infancy of the field. Noteably, the difference between path-planning and motion-planning is crucial; path planners concentrate on the problem of finding a safe geometric path between pairs of starting-ending configurations on the physical space, while motion planning concerns planning over the state space (e.g., joint angles and velocities) in order to achieve safe navigation. In many approaches these two aspects are understandably decoupled, although an integrated approach is theoretically possible. As a result, there exist a plethora of approaches to address the aforementioned problems, with two main classes emerging: Sampling-Based Methods (SBMs) and Continuous Methods (CMs). SBMs have been widely celebrated over the last few decades, owing to their relative simplicity and -in some cases- optimality guarantees. CMs on the other hand have not been thoroughly developed w.r.t. optimality; The positive traits of CMs, namely smoothness, global properties and provable guarantees, motivate us in filling this gap. This is in contrast to SBMs, which are most efficient in providing a single, non-smooth path (barring post-processing through other methods).

In this work, a CM based on position state-feedback is proposed. The herein employed state space coincides with the physical geometric space, thus our method can also be contextualized as a path planner. Furthermore, this work belongs to a class of methods aiming at formally extending optimality to the CM paradigm [2], [3], [4], [5], [6], providing the crucial extension to environments with obstacles. The goal is to overcome identified pathologies of continuous methods related to obstacles [7], while at the same time providing safety, convergence and optimality guarantees. Any obstacles induce topological issues that render any optimal control problem difficult to solve [8], even in simple cases for robot dynamics and workspace geometries. Therefore, even in the traditional motion planning problem, obstacles require careful consideration.

II. Related Work

One of the most celebrated class of methods for path planning consist of the Sampling-Based Methods (SBMs) [9] and discrete/graph-based ones. Prominent examples of such approaches include Djikstra's algorithm [10], A* [11], D* and D*-lite [12], Random Rapidly exploring Trees (RRT) [13] and its variant RRT* [14], Probabilistic Road Maps (PRMs) [15], Fast Marching Trees (FMT*) [16] and Visibility Graphs (VGs) [17]. The wide adoption of such methods is accompanied by extensive development; for example, RRT* variants include smart re-planning [18], continuous cost formulations [19], dynamic workspaces [20], control barrier functions formulations [21] and extensions such as informed RRT* [22], RRT*-SMART [23], etc. Interestingly, the FMT approach has gone through developments where a continuous, wave-front propagation method produces a funnel-shaped potential, akin to AHPFs [24] This method proves to be both fast and effective in complex maps. Concerning optimality, Djikstra's Algorithm, A* and D* and RRT* (and its variants) produce minimum length quasi-linear

Manuscript received 21 July 2023; accepted 28 October 2023. Date of publication 2 November 2023; date of current version 9 November 2023. This letter was recommended for publication by Associate Editor J. Capitan and Editor A. Bera upon evaluation of the reviewers' comments. This work was supported by the Hellenic Foundation for Research and Innovation (HFRI) under the 4th Call for HFRI PhD Fellowships [Fellowship Number: 9110]. *(Corresponding author: Panagiotis Rousseas.)*

Panagiotis Rousseas is with the School of Mechanical Engineering, Control Systems Laboratory, National Technical University of Athens, 10682 Athens, Greece (e-mail: prousseas@mail.ntua.gr).

Charalampos P. Bechlioulis is with the Department of Electrical and Computer Engineering, University of Patras, 26504 Patras, Greece (e-mail: chmpechl@upatras.gr).

Kostas J. Kyriakopoulos is with the Center of AI & Robotics (CAIR), New York University, Abu Dhabi 129188, UAE (e-mail: kkyria@nyu.edu).

Digital Object Identifier 10.1109/LRA.2023.3329624

paths. In case of RRT* methods, asymptotic optimality guarantees are also proven.

Prior to such SBMs, continuous methods were developed to tackle Motion Planning (MP) through continuous vector fields. Such methods usually design real-valued potentials over the workspace. Notable examples include Navigation Functions (NFs) [7] and Artificial Potential Fields (APFs), where convergence is guaranteed through the presence of a single local minima, while safety is maintained through boundary conditions. Posteriorly, Artificial Harmonic Potential Fields (AHPFs) [25] were developed in order to overcome tunability issues of the former methods, and are used for navigating within disk worlds, where the properties of harmonic functions enable extracting "tune-free" approaches. Crucially, extending AHPFs to arbitrary workspaces, necessitates transformations of arbitrary workspaces into disk worlds [26], [27], [28]. Alternatively, more recently the construction of harmonic vector basis' directly on the physical workspace [2], [4], [29] has been investigated.

Nevertheless, solutions to *continuous* OMP, have not been extensively researched with some exceptions, [30], [31], [32] which, however necessitate the solution of a hard Partial Differential Equation (PDE), or extensive parameter tuning. The authors' efforts have thus far concentrated on the parameter-tuning paradigm [2], [3], [5], where the properties of parameterized AHPFs were employed, which however limits the space of possible policies. Our most recent work [6], concentrates on the parameter-free paradigm, where nearly globally optimal solutions were extracted. Thus, a promising research direction for reactive OMP was discovered, as the limitations stemming from parametrizing the policy were overcome.

Finally, learning methods have also been employed [33], however most such methods are platform-specific and are thus of limited relevance to this work. For instance, solutions to robotic locomotion [34], [35], [36] necessitate high-level velocity commands [34], [35]. **Hence, our class of methods is pertinent as it can serve as such a high-level controller.**

III. Problem Formulation

Consider a point robot,[1] operating within a two-dimensional, connected and bounded set $\mathcal{Q} \subset \mathbb{R}^2$, with $M \in \mathbb{N}_+$ inner distinct obstacle $\mathcal{O}_i \subset \mathcal{Q}$, $i \in \mathcal{M}$, where $\mathcal{M}$ denotes the index set $\mathcal{M} = \{1, \ldots, M\}$. The robot's workspace is thus denoted by $\mathcal{W} = \mathcal{Q} - \bigcup_{i=1}^{M} \mathcal{O}_i$ and its boundary is denoted by $\partial\mathcal{W}$. Further consider a desired final position within the workspace denoted by $p_d \in \mathcal{W}$. In this work, we treat the single integrator dynamics:

$$\dot{p} = u, p(0) = \bar{p}, \tag{1}$$

where $p(t) : \mathbb{R}_+ \mapsto \mathcal{W}$ denotes the robot's position, $\bar{p} \in \mathcal{W}$ denotes the robot's position at time $t = 0$ and $u(t)$ denotes the input signal, i.e., the robot's velocity.

The aim is to design a **state-feedback velocity input**[2] $u(p) : \mathcal{W} \mapsto \mathbb{R}^2$ that minimizes the quadratic infinite-horizon cost function:

$$V_u(\bar{p}) = \int_0^\infty Q(p_u(\tau; \bar{p}); p_d) + R(u(\tau))\, \mathrm{d}\tau, \tag{2}$$

where $p_u(t; \bar{p}) : \mathbb{R}_+ \mapsto \mathcal{W}$ denotes the trajectory of System (1) under the control input u, from the initial position $p(0) = \bar{p} \in \mathcal{W}$. Furthermore, the **state-related cost term** Q and the **input-related cost term** R are defined as:

$$Q(p; p_d) = \alpha \|p - p_d\|^2, \tag{3a}$$

$$R(u) = \beta \|u\|^2, \tag{3b}$$

where $\alpha, \beta \in \mathbb{R}_+$ are design parameters and $\|\cdot\|$ is the Euclidean norm. The metric (2) along with (3) stem from [37] and the term (3a) reflects the minimization of the *settling time* of the system, while (3b) penalizes the *energy expenditure* of System (1).

A. Optimal Motion Planning in Simply-Connected Workspaces

In a previous work [6], we have proposed a method that computes the optimal velocity field in simply-connected workspaces, based on a Policy Iteration (PI) scheme. Based on the results in [38], along with the technical results in [6], the near globally optimal policy is indeed extracted in the absence obstacles. This method will be employed herein as well, and the reader is directed to this work for further details which will not be elaborated upon here. In the following subsection, the reasons why the aforementioned method can not be applied in the presence of obstacles are outlined.

B. Topological Perplexity in Multiply-Connected Workspaces

A fundamental aspect of feedback stabilization in multiply-connected workspaces (i.e., workspaces with "holes") is the notion of Topological Perplexity (TP) [39]. The aforementioned "holes" impose restrictions on the design of feedback laws on such workspaces. Namely, *strict* feedback stabilization is impossible in multiply-connected workspaces. This is a well-documented feature of CMs for motion planning [7], where local maxima and saddles are imparted in the presence of obstacles. While these subsets do not pose limitations in the context of motion planning, since they are of lesser measure than the workspace itself, this is not the case when treating optimality. The aforementioned manifolds contain forward invariant sets which evidently render the integral (2) infinite (see Fig. 1). Hence the method in [6] can not be employed in this case due to the following three effects: 1) the approximation of the infinite cost function, 2) the resulting velocity for the proposed scheme will inherit the singularities of any initial velocity field – thus it is not optimal, 3) the forward-invariant manifolds will persist in the successive scheme, due to 2), essentially subdividing the

[1] A disk robot can also be considered by applying a workspace transformation that inflates the workspace boundaries: $\partial\mathcal{W}^I = T(\partial\mathcal{W})$ where $T(z) = z + Rn(z)$, $R \in \mathbb{R}_+$ denoting the robot's radius and n denoting the inwards-pointing vector that is normal to the boundary at the point $z \in \partial\mathcal{W}$.

[2] To avoid any ambiguity, we note that a reactive field $u(p)$ can be expressed as a function of time, if it is evaluated along trajectories of System (1), i.e, $u(t) \triangleq u(p_u(t; \bar{p}))$, therefore our definitions of time series vs velocity fields are consistent throughout the manuscript.

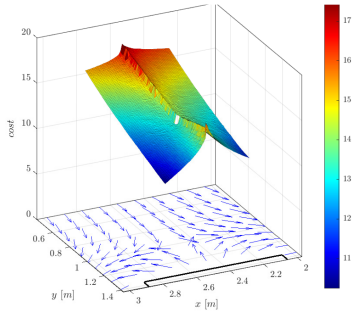

Fig. 1. Example of the cost function near a saddle. The infinity cannot be appropriately demonstrated owing to numerically extracting the cost function values, but note the steep rise of the cost near the saddle's stable manifold, which indicates the latter.

workspace in a sub-optimal manner. Nevertheless, this work aims at overcoming these issues through a step of transforming the workspace into a simply-connected version, where the globally optimal solution can be provably extracted. The crucial aspect of the proposed scheme rests on the workspace transformation, through two-dimensional subsets, which significantly impacts the optimality of the scheme. This is implemented through theory-inspired heuristics, such that a complete method is established. Finally, we note that this method produces a **local** solution, in the sense that the workspace transformation depends on the initialization of the method, as it will become apparent.

IV. Proposed Method

We begin by highlighting the steps of the proposed method, which are discussed in the following subsections: 1) Transform the workspace into a simply-connected version, 2) Find the locally optimal version of the simply-connected version of step 1), 3) Solve the optimal motion planning problem in the transformed workspace of step 2) through [6].

1) Transformation to Simply-Connected Workspace: In this subsection a complete approach for transforming multiply to simply-connected workspaces is presented. This transformation consists of the difference between the initial workspace and a set of two-dimensional regions that link obstacles to the outer boundary or other obstacles, and is denoted as: $\mathcal{W}_s = \mathcal{W} \setminus (\bigcup_{i=1}^{M} \mathcal{S}_i)$, where $\mathcal{S}_i \subset \mathcal{W},\ i \in \mathcal{M}$. The main focus of the proposed scheme rests on the computation of the aforementioned subsets. To this end, the properties of AHPFs are employed. Consider an initial AHPF-based velocity vector field as in [6] defined over the **initial** workspace $\mathcal{W}$ with M ostacles. Then, according to the following proposition, this field exhibits **exactly** M saddles:

Proposition 1: Let $\mathcal{W} = \mathcal{Q} - \bigcup_{i=1}^{M} \mathcal{O}_i$ denote a Jordan (the boundary of $\mathcal{W}$ consists of disjointed Jordan curves) multiply connected workspace (manifold) embedded in the Euclidean plane ($\mathcal{W} \subset \mathbb{R}^2$), with M internal obstacles. A Harmonic Navigation Field [6], defined over $\mathcal{W}$ exhibits exactly M saddles in $\text{int}(\mathcal{W})$.

Proof: See Appendix. ■

The aforementioned saddles consist of the critical point, along with two sub-manifolds, the stable and unstable one [40]. The stable manifold is exactly the forward invariant set that induces the infinity on the related cost function and can be used to construct the two dimensional sets $\mathcal{S}_i \subset \mathcal{W},\ i \in \mathcal{M}$ (henceforth referred to as "TP regions"). It is easy to see that the stable manifolds connect the obstacles to the boundary or other obstacles. Assuming otherwise would imply that there exist internal local minima, which is prohibited by the min-max principle of AHPFs [41]. Therefore, the stable manifolds of the saddles, denoted by $\mathcal{S}_i^s \in \mathcal{W},\ i \subset \mathcal{M}$ can be used to create the sets $\mathcal{S}_i\ i \in \mathcal{M}$ as follows:

$$\mathcal{S}_i(r) = \{z \in \mathcal{W} \text{ s.t. } \|z - \omega\| \leq r,\ \forall \omega \in \mathcal{S}_i^s\},\ i \in \mathcal{M}, \quad (4)$$

where $r \in \mathbb{R}_+$ denotes a small buffer radius around the stable manifold for forming the two-dimensional region. This is furthermore a valid transformation to a simply-connected version of the workspace, in the sense that there exists a AHPF velocity field free of saddles. This is evidenced if one considers the initial velocity field that was employed to extract the aforementioned regions; since its saddles are omitted in the transformed workspace $\mathcal{W}_s$, there exists a small enough radius $r > 0$ such that the former can be used to navigate from everywhere to the goal position.

2) Locally Optimal Simply-Connected Version:: Having extracted an initial, simply-connected version of the multiply-connected workspace, in this section a theory-inspired heuristic for finding the locally optimal position for the sets $\mathcal{S}_i,\ i \in \mathcal{M}$ is presented. A metric for assessing the optimality of the choice of regions is the difference of the **optimal** cost function on the two disjoint free boundaries[3] of the sets $\mathcal{S}_i$. If there is a difference in cost between the former, then this implies that the respective region should be displaced towards the larger cost values. This intuitively follows from considering that the trajectories for initial points that exhibit the larger cost value could exhibit decreased cost values, if they were to follow the (optimal) policy that leads them to crossing the respective region $\mathcal{S}_i^s$. Put simply, the latter regions dictate whether trajectories should "go to the left or to the right" of the respective obstacle. Hence, a sub-optimal choice for the regions leads to a sub-optimal choice between the former two choices. This is outlined in the following theorem, where we prove how an ameliorated choice for the region placement exists. Theorem 1 is based upon the limit where the buffer radius r approaches zero, hence the regions $\mathcal{S}_i$ become one-dimensional and are aptly referred to as "split regions".

Theorem 1: The optimal policy for a multiply-connected case exhibits equal cost over the split regions, i.e., the optimal cost function is continuous.

Proof: See Appendix. ■

Theorem 1 motivates the heuristic of displacing the regions $\mathcal{S}_i,\ i \in \mathcal{M}$ even when the buffer radius is greater than zero. Therefore, starting from a sub-optimal initial choice for the regions $\mathcal{S}_i^{(0)},\ i \in \mathcal{M}$, a sequence of regions can be extracted $\{\mathcal{S}_i^{(0)}, \mathcal{S}_i^{(1)}, \ldots, \mathcal{S}_i^{(J)}\},\ i \in \mathcal{M}$, by successively displacing each region based on the difference in the *optimal cost* on the free boundaries of each region. Note that, extracting the optimal cost

[3]The term "free boundary" is employed to refer to the parts of the boundary $\partial\mathcal{S}_i$ of the respective sets that lie on the **interior** of the **initial** workspace $\mathcal{W}$.

Algorithm 1: Proposed Algorithm.

- **Inputs**: Workspace $\mathcal{W}$, TP region buffer radius $r > 0$, scaling constant $\delta \in \mathbb{R}_+$, maximum TP displacement iterations $J \in \mathbb{N}$.
- Compute an initial safe and convergent AHPF [6].
- Compute the stable manifolds of each saddle (5) and the respective initial TP regions $\mathcal{S}_i^{(0)},\ i \in \mathcal{M}$ through (4).

for $j = 0 : J$ **do**

- Compute an initial policy $u^{(j)} : \mathcal{W}_s^{(j)} \mapsto \mathbb{R}^2$ for the transformed workspace $\mathcal{W}_s^{(j)} = \mathcal{W} \setminus (\bigcup_{i=1}^{M} \mathcal{S}_i^{(j)})$.
- Compute the cost function values $\overline{V_{u^{(j)}}},\ \underline{V_{u^{(j)}}}$ through integrating the resperive trajectories.
- Displace the manifolds through (7).

end for

- Compute the optimal velocity field through the method in [6], applied for the final simply-connected version of the workspace $\mathcal{W}_s^{(J)} = \mathcal{W} \setminus (\bigcup_{i=1}^{M} \mathcal{S}_i^{(J)})$.

value is not limited to the method in [6], SBMs such as RRT* can also be employed, resulting in faster convergence in certain cases (as it requires a small number of trajectories to be evaluated).

V. Implementation Details

In this section, we discuss several details over the implementation of the method. The details that are discussed in the following subsections are also presented in the form of an algorithm in Algorithm 1.

1) Initial Workspace Decomposition: The proposed method necessitates a complete method for transforming the workspace into a simply connected version. Owing to Proposition 1, an AHPF defined over $\mathcal{W}$ yields a set of M saddle-points $\bar{s}_i \in \mathcal{W},\ i \in \mathcal{M}$. In order to extract the sets $\mathcal{S}_i,\ i \in \mathcal{M}$, the stable manifolds $\mathcal{S}_i^s,\ i \in \mathcal{M}$ are computed as follows: Consider the stationary point of a saddle $\bar{s}_i \in \mathcal{W},\ i \in \mathcal{M}$. Then, in case of non-degenerate critical points, the Hessian of the field $\mathcal{H}(\bar{s}_i) \in \mathbb{R}^{2\times 2}$ can be computed, and exhibits one strictly positive and one negative eigenvalue and two corresponding eigenvectors, denoted by $\lambda^+, \lambda^- \in \mathbb{R}$ and $v^+, v^- \in \mathbb{R}^2$ respectively. The negative eigenvalue-eigenvector pair corresponds to the stable manifold [40]. Therefore, the stable manifolds $\mathcal{S}_i^s,\ i \in \mathcal{M}$ can be extracted by solving the following system of ODEs from two initial positions:

$$\dot{p} = -u^{(0)}(p),\ p(0) = \bar{s}_i \pm \epsilon v^-, \tag{5}$$

where $u^{(0)}(p) : \mathcal{W} \mapsto \mathbb{R}^2$ denotes the AHPF vector field, $\epsilon > 0$ is a small positive constant and the $\pm$ sign corresponds to two initial positions yielding the two branches of the stable manifold. In case of degenerate critical points, these are also isolated saddles, owing to the properties of AHPFs [42]. However, the stable direction of the saddles cannot be extracted through the Hessian. To amend this, the integration in (5) can be computed for a set of points that are distributed uniformly around the critical point at a distance equal to ϵ, which will result in extracting both sections of the saddle's manifolds, due to the negated unstable saddle manifold's attractiveness. The solution of (5) is carried out until the trajectories reach the boundary $\partial\mathcal{W}$. The integration is carried out backwards in time -note the negative sign in (5)- in order to render the scheme stable, as the trajectories converge to the negated stable manifold, while the small constant ϵ places the initial point away from the stationary saddle point (which would halt the integration indefinitely). The eigenvector is employed to get "as close" to the stable set as possible, owing to the linearizable nature of the field close to the saddle. An example of the above is depicted in Fig. 2. Finally, the TP regions $\mathcal{S}_i(r),\ i \in \mathcal{M}$ can be easily computed.

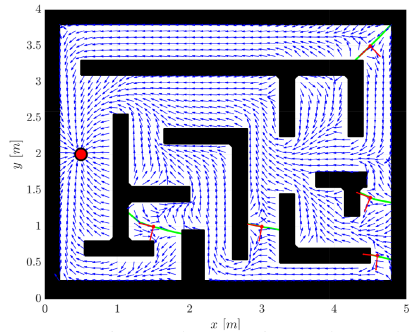

Fig. 2. Example of the result of the stable manifolds. The workspace (black region), the normalized AHPF vector field (blue arrows), the stable manifolds (green curves) and the corresponding eigenvectors (red arrows) are depicted for a goal position (red disk).

2) TP Regions Displacement: In order to displace the TP regions $\mathcal{S}_i,\ i \in \mathcal{M}$, the optimal policy for a given transformation should be computed on the boundaries $\partial\mathcal{S}_i,\ i \in \mathcal{M}$, in order to asses the equality criterion established in Theorem 1. Notably, this necessitates solving the OMP in [6] for each evaluation. In order to render the scheme more efficient, we note how, owing to the kinematic problem treated herein, the most crucial aspect for optimality is the velocity norm, which can be computed in a closed form [6]. Hence, the initial policy, imbued with the latter norm is employed to extract the cost difference for each control point. While this approach does not necessarily yield the optimal position for the TP regions, we nevertheless demonstrate in results' section the optimality of the final results. We stress that other methods such as RRT* can be employed to assess the positions of the TP regions, however, owing to their statistical nature they have not been employed herein.

Consider now the TP regions and their corresponding stable manifolds $\mathcal{S}_i$ and $\mathcal{S}_i^s,\ i \in \mathcal{M}$. The regions $\mathcal{S}_i$ necessarily contain two, disconnected "free" boundaries, denoted by $\overline{\partial\mathcal{S}_i},\ \underline{\partial\mathcal{S}_i} \subset \mathcal{W} \setminus \partial\mathcal{W}$. A safe and convergent velocity field for the **transformed** workspace $u(p) : \mathcal{W}_s \mapsto \mathbb{R}^2$ admits a continuous and bounded cost function $V_u(p)$: over the former sets, i.e.:

$$\begin{aligned} \overline{V_u} &= V_u\left(\overline{\partial\mathcal{S}_i}\right) : \overline{\partial\mathcal{S}_i} \mapsto \mathbb{R}_+ \\ \underline{V_u} &= V_u\left(\underline{\partial\mathcal{S}_i}\right) : \underline{\partial\mathcal{S}_i} \mapsto \mathbb{R}_+. \end{aligned} \tag{6}$$

Hence, starting from a stable manifold $\mathcal{S}_i^{s(j)},\ i \in \mathcal{M},\ j = 1, \ldots, J$, a subsequent manifold can be extracted by

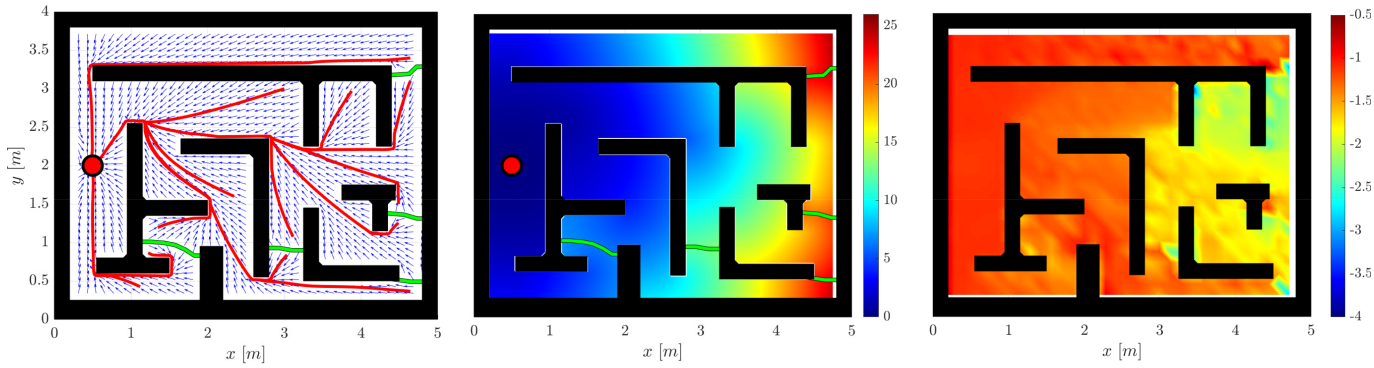

Fig. 3. Results for the proposed method. The workspace is depicted through the black-shaded regions, the goal position is depicted through a red disk and the TP regions are depicted through the green-shaded regions. From left to right, the normalized velocity field is depicted with blue arrows, exemplary trajectories are depicted with red lines the final (optimal) cost function is depicted through the colormap. In the center-most figure, the cost of the proposed method is presented while in the right-most figure the cost difference between our method and the RRT* is presented.

displacing it:

$$\mathcal{S}_i^{s(j+1)}(\gamma) = \mathcal{S}_i^{s(j)}(\gamma) + \delta\left(\overline{V_u}(\gamma) - \underline{V_u}(\gamma)\right)\hat{n}_i^j(\gamma),$$
$$i \in \mathcal{M},\ j = 1, \ldots, J \tag{7}$$

where $\gamma \in [0,1]$ is a variable that parameterizes each one-dimensional manifold $\mathcal{S}_i^{s(j)},\ i \in \mathcal{M},\ j = 1, \ldots, J$, $\hat{n}_i^j(\gamma) : [0,1] \mapsto \mathbb{R}^2$ is the unit vector normal to the curve $\mathcal{S}_i^{s(j)}$ for each $\gamma \in [0,1]$ and $\delta \in \mathbb{R}^+$ is a scaling constant. Concurrently, the cost function values are defined as follows:

$$\overline{V_u}(\gamma) = \overline{V_u}\left(p(\gamma)\right) \text{ where } p(\gamma) = \arg\min_{z \in \overline{\partial \mathcal{S}_i}} \{\|\mathcal{S}_i^{s(j)}(\gamma) - z\|\},$$
$$\underline{V_u}(\gamma) = \underline{V_u}\left(p(\gamma)\right) \text{ where } p(\gamma) = \arg\min_{z \in \underline{\partial \mathcal{S}_i}} \{\|\mathcal{S}_i^{s(j)}(\gamma) - z\|\}, \tag{8}$$

which essentially denotes the closest cost function values for each sector of the free boundaries of $\mathcal{S}_i^{s(j)}$ to each point in the stable manifold. The cost function values can be computed through numerically integrating the trajectories of (1) along with the cost (2). Note that the normal vector in (7) should be oriented such that the curve is displaced towards the maximal value $\max\{\overline{V_u}(\gamma), \underline{V_u}(\gamma)\}$. Hence, through (7) and (8) a sequence of manifolds is constructed $\{\mathcal{S}_i^{s(0)}, \mathcal{S}_i^{s(1)}, \ldots, \mathcal{S}_i^{s(J)}\},\ i \in \mathcal{M}$, and hence the sequence of regions $\{\mathcal{S}_i^{(0)}, \mathcal{S}_i^{(1)}, \ldots, \mathcal{S}_i^{(J)}\},\ i \in \mathcal{M}$ is extracted through (4).

Remark 1: Any additional dynamic obstacles -to the ones included during the herein presented OMP solution- can be accounted for through appropriate local controllers to ensure safety via local collision avoidance. However, this would result in a sub-optimal solution, which is to be expected. In order to provide a provably optimal solution, the motion profile for each obstacle needs to be available, in order to enable extracting the time dependent optimal cost function. This is left as part of future works.

VI. Results

In this section, we provide results for our method over two workspaces with internal obstacles. Furthermore, we provide comparative studies with several SBMs. The proposed method, as well as the RRT* one, were implemented in MATLAB version 2022a, while the rest of the SBMs are python implementations. In order to compare the cost function (2) for both methods, we imbue the SBMs with the optimal norm [6] and calculate the respective on-trajectory cost numerically. Finally, in order to provide a fair comparison of execution times for all methods, -since the proposed method provides a solution form everywhere towards a final goal position- the SBMs' initial positions for the cost computation were sampled at a sufficiently dense grid.

Our method's results over two workspaces with obstacles are presented in Figs. 3 and 4, where the normalized final vector fields as well as the final cost functions are presented, along with exemplary trajectories. The parameters for each case respectively were set as: $r = 0.025,\ \epsilon = 0.075,\ \delta = 0.25$ and $r = 0.015,\ \epsilon = 0.075,\ \delta = 0.25$. In order to quantify and asses the effect of the workspace transformation, we employ the ratio of the area that is discarded over the area of the initial workspace. This quantifies the probability of a starting position landing on the former sets, where navigation is not complete through our method. This could however be easily amended through heuristics, especially since the TP regions' size is small, but is left out of the scope of this work. The aforementioned ratio for the workspaces of Figs. 3 and 4 are equal to 0.94% and 1.14% respectively. Thus, the small ratio values are deemed satisfactory.

Finally, while our method does not produce the true minimum-path solution -compared to e.g., VG methods, this is owing to the former being extracted as integral curves of the optimal velocity field. Hence, owing to ODE theory, trajectories should not overlap and smoothness is conserved. Thus, our method trades truly minimum-length paths for smoothness of the solution and the velocity input, which we believe is a valuable trade-off.

1) Comparative Studies: Comparative results between our method and several SBMs are depicted in Figs. 3 and 5, in the form of the difference: $\text{Diff} = \text{V}_{\text{proposed}} - \text{V}_{\text{SBM}}$. In Fig. 3 the comparison is carried out against an RRT* method, while in Fig. 5 an RRT*, a Smart RRT*, an FMT* and an Informed RRT* methods are presented. Hence, negative values correspond to our method outperforming the former ones (note that each figure has its own colomap range and colors). It is evident that all

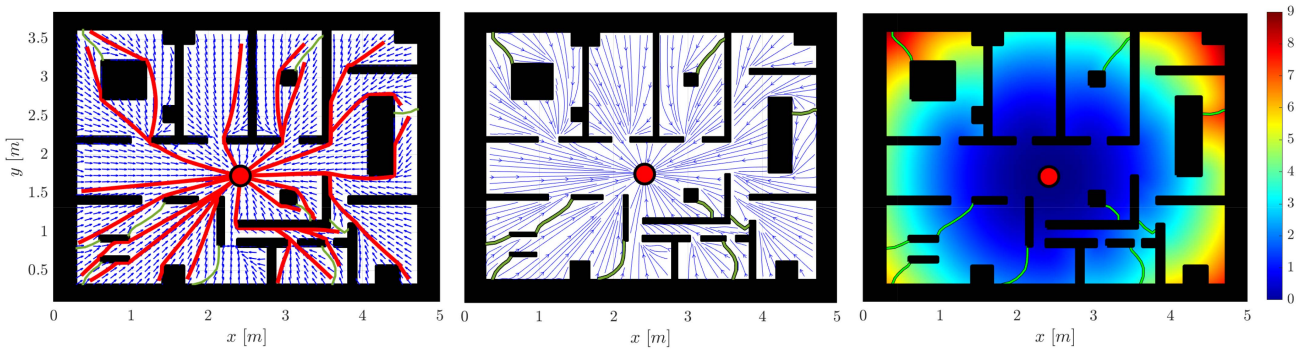

Fig. 4. Results for the proposed method. The workspace is depicted through the black-shaded regions, the goal position is depicted through a red disk and the TP regions are depicted through the green-shaded regions. From left to right, the normalized velocity field is depicted with blue arrows, exemplary trajectories are depicted with red lines, streamlines are depicted with blue lines and finally the optimal cost function is depicted through the colormap.

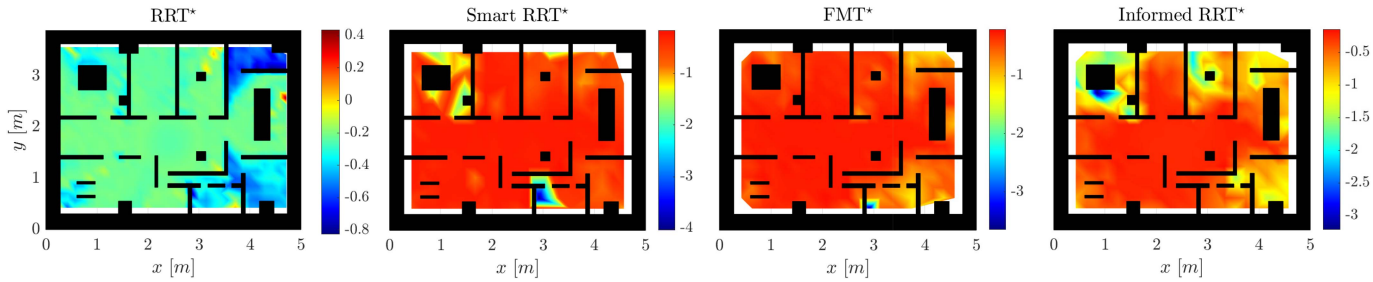

Fig. 5. Comparative plots of the difference of the proposed method with the SBM costs. Negative values indicate that our method outperforms the compared methods.

TABLE I
COMPARATIVE EXECUTION TIMES

Workspace of Fig. 4					
	Number of Points	# Iterations	Mean Time per Traj. $[s]$	Total Time $[mins]$	
				13.9	
Ours	-	-	-	Time for TP Regions	Time for OMP
				3.9	10
RRT⋆	656	2500	3.9791	43.5	
Smart RRT⋆	289	1000	25.9389	124.9	
Informed RRT⋆	288	2000	26.9195	129.2	
FMT⋆	288	2000	9.0642	43.5	
Workspace of Fig. 3					
				6.2	
Ours	-	-	-	Time for TP Regions	Time for OMP
				0.9	5.3
RRT⋆	701	2500	4.17	48.8	

approaches except for the RRT⋆ produce inferior results to our method. The only case where the RRT⋆ method outperforms ours in some instances is for the workspace of Fig. 4, but even then, for the vast majority of points our method proves superior, with only very few exceptions near the TP regions. We underline that for statisctical significance, 10 RRT⋆ trajectories were run for each starting point, and the best results were employed to extract the aforementioned comparisons.

Finally, we demonstrate qualitative results over a workspace with geometrical characteristics which have proven to be problematic for the RRT⋆ method in Fig. 6; namely, with narrow corridors. Our method provides smoother trajectories with lesser length, while the SBM struggles owing to the small probability of sampling optimal waypoints inside the slim regions. Notably, 100 RRT⋆ trials were carried out for each starting position of Fig. 6, with a success rate of 27%. Therefore, our method is considered far superior for the aforementioned type of workspace, since it is able to provide a solution from anywhere within the workspace, with no failures. Finally, as the corridor's width is decreased, the proposed method's run-time remains mostly unaffected, whereas the RRT⋆'s success rate deteriorates.

2) Computational Complexity: In order to asses the computational complexity of our method, all of the methods were run on the same machine -a PC running Ubuntu LTS 18.04, i7 CPU and 50 Gb RAM- and were timed accordingly. In Table I, details about the SBMs' parameters as well as execution time data are compared against the execution time of our method. It is evident that even for a conservative sampling of initial positions, our method is far superior in execution time. While these results do not reflect the execution times for a single trajectory, our method is especially advantageous in cases where a large number of trajectories might be required.

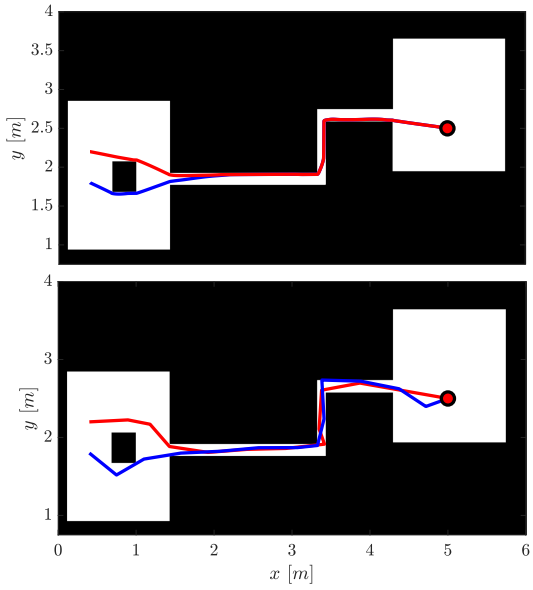

Fig. 6. Comparative trajectories between our method (top) and RRT* (bottom) for two initial positions.

Fig. 7. Figure for the proof of Theorem 1. The boundaries of the workspace (black regions), the split-region (red line) and the defined sets of the proof (green and blue). In the right figure the two sides of the assumed discontinuous optimal cost function are depicted (isometric view). The lesser part of the cost function can be smoothly extended to provide a "better" input.

VII. Conclusions and Future Directions

In this work, an approach to extend the provably asymptotically globally optimal solution to OMP [6] is presented. The proposed method is effective in tackling the problem of topological perplexity in complex workspaces, without sacrificing the applicability of previous results. At the same time, the approach proves to be superior to several modern, and most crucially also provably asymptotically optimal, SBMs while inheriting the advantages of continuous method. One limitation of the method rests on the locally optimal nature of the workspace decomposition, which depends on the initial saddles' position and which we aim to address in the future. Furthermore, in future works, we aim at extending the presented framework to higher order systems (e.g., double integrator, mechanical systems) as well as three-dimensional workspaces.

Appendix

1) Proof of Proposition 1: Proof: Let $\mathcal{D} \subset \mathbb{R}^2$ denote the unitary, planar disk. In [43], it has been proven that the Euler characteristic χ of any n-dimensional sphere world homeomorph $\mathcal{F}^n$ with M obstacles is: $\chi(\mathcal{F}^n) = 1-(-1)^n M$. However it is trivial to show that any planar sphere world $\mathcal{F}^2 \subset \mathbb{R}^2$ is homeomorphic to the *punctured* disk $\mathcal{D}' = \mathcal{D} - \bigcup_{i=1}^{M} d_i$, where $d_i = \mathcal{T}(\partial\mathcal{O}_i) \in \mathcal{D}, i = 1,\dots,M$ and $\mathcal{T}$ is the respective diffeomorphism, while $\mathcal{W}$ is also homeomorphic to $\mathcal{D}'$ [26]. Therefore, since the Euler characteristic is topologically invariant: $\chi(\mathcal{W}) = \chi(\mathcal{D}') = \chi(\mathcal{F}^2) = 1-(-1)^2 M = 1-M$. Consider the Poincaré-Hopf Theorem: $\sum_{i=1}^{I} \text{index}_{p_i}(v) = \chi(\mathcal{W}) = 1-M$, where $p_i,\ i = 1,\dots,I$ denotes the I isolated zeros of the vector field v. Let $\mathcal{I}_s \subset \{1,2,\dots,I\}$ denote the index set of the saddles of v that lie in int$(\mathcal{W})$. Owing to the form of v in [6], along with the minimum-maximum principle of Harmonic Functions, the isolated singularities of v consist only of the saddles $p_s, s \in \mathcal{I}_s$, and the single stable equilibrium at p_d. Therefore, we get: $\sum_{i=1}^{I} \text{index}_{p_i}(v) = \text{index}_{p_d}(v) + \sum_{s\in\mathcal{I}_s} \text{index}_{p_s}(v)$. However, for any saddle in the two-dimensional plane $\text{index}_{p_s}(v) = -1,\ s \in \mathcal{I}_s$,[4] while for the sink $\text{index}_{p_d}(v) = +1$. Therefore, we finally get: $1 + \sum_{s\in\mathcal{I}_s}(-1) = 1-M \Rightarrow = \#(\mathcal{I}_s) = M$ where $\#(\cdot)$ denotes the cardinality of a set, concluding the proof since the cardinality of $\mathcal{I}_s$ is equal to the number of saddles in int$(\mathcal{W})$. ■

2) Proof of Theorem 1: Proof: We will prove this by contradiction. Assume that the optimal cost function is discontinuous. For the sake of simplicity we assume that for each point on the split region the inequality relationship between the cost values is constant, i.e.:

$$V^\star(\overline{z}) > V^\star(\underline{z}),\ \forall \overline{z} = \lim_{z \to {}^{+}\mathcal{S}_i(0)} (z) \tag{9}$$

where $\underline{z} = \lim_{z\to{}^{+}\mathcal{S}_i(0)}(z),\ i \in \mathcal{M}$, $\overline{z}, \underline{z}$ denote the points approaching the split region from the two different "sides" of the region. Consider now the optimal policy $u^\star(p),\ p \in \underline{z}$. Assuming that the boundary of the workspace is locally smooth, then further consider a smooth extension of $u^\star$, denoted by $\underline{u}^\star$ over a set $\mathcal{S}(p) \subset \mathcal{W}$ that contains only the side of the split region that contains $\overline{z}$. Owing to smoothness, if the region $\mathcal{S}$ is sufficiently small, the input $\underline{u}^\star$ is safe. Hence, the respective cost function $V^\star_{\mathcal{S}}(p),\ p \in \mathcal{S}$ obeys the following PDE:

$$\left(\nabla V^\star_{\mathcal{S}}\right)^{\mathrm{T}} \underline{u}^\star = -Q\left(p; p_d\right) - R\left(\underline{u}^\star(p)\right),\ p \in \mathcal{S}, \tag{10}$$

which can be solved through the method of characteristics [45] through the following three ordinary differential equations:

$$\frac{\mathrm{d}x}{\underline{u}^\star_x(p)} = \frac{\mathrm{d}y}{\underline{u}^\star_y(p)} = -\frac{\mathrm{d}V^\star_{\mathcal{S}}}{Q\left(p; p_d\right) + R\left(u(p)\right)} = \mathrm{d}t, \tag{11}$$

where $[x,y]^{\mathrm{T}} = p \in \mathcal{S}$ and $[\underline{u}^\star_x, \underline{u}^\star_y] = \underline{u}^\star(p)$. Finally, the initial condition for the cost function in (10) is $V^\star_{\mathcal{S}}(t=0) = V^\star(p(0)),\ p(0) \in \overline{z}$. Through the method of characteristics, the first two terms in (10), starting from the split region, yield iso-chronous curves of the form $g(x,y) = t$. Computed in backwards-time, the cost function $V^\star_{\mathcal{S}}(x,y)$ is strictly increasing owing to $Q,\ R$ being positive definite away from the goal position. Therefore, through the intermediate value theorem, continuity of the solution of (10) as well as (11), we conclude that there exists a subset $\underline{\mathcal{S}} \subseteq \mathcal{S}$ such that (see Fig. 7) -through an

[4]We note that, even in case of degenerate critical points, they can be shown to be isolated saddles for AHPFs, as well as attain an index of (-1) [44].

abuse of notation-: $V^\star(\underline{z}) \leq V^\star_{\mathcal{S}}(p) \leq V^\star(p),\ \forall p \in \underline{\mathcal{S}}$. Hence, the cost function $V^\star(p)$ is not optimal on the set $\underline{\mathcal{S}}$, since we have constructed a safe policy with lesser cost on the latter set, thus leading to a contradiction and concluding the proof. ■

REFERENCES

[1] N. Roy et al., "From machine learning to robotics: Challenges and opportunities for embodied intelligence,"2021, *arXiv:2110.15245*.

[2] P. Rousseas, C. Bechlioulis, and K. J. Kyriakopoulos, "Harmonic-based optimal motion planning in constrained workspaces using reinforcement learning," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2005–2011, Apr. 2021.

[3] P. Rousseas, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Optimal robot motion planning in constrained workspaces using reinforcement learning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 6917–6922.

[4] P. Rousseas, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Trajectory planning in unknown 2D workspaces: A smooth, reactive, harmonics-based approach," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 1992–1999, Apr. 2022.

[5] P. Rousseas, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Optimal motion planning in unknown workspaces using integral reinforcement learning," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 6926–6933, Jul. 2022.

[6] P. Rousseas, C. P. Bechlioulis, and K. J. Kyriakopoulos, "A continuous off-policy reinforcement learning scheme for optimal motion planning in simply-connected workspaces," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 10247–10253.

[7] E. Rimon and D. E. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Trans. Robot. Automat.*, vol. 8, no. 5, pp. 501–518, Oct. 1992 .

[8] M. D. Kvalheim and D. E. Koditschek, "Necessary conditions for feedback stabilization and safety," 2022 [Online]. Available: /article/id/62aabc922d80b75d2b6d88c4

[9] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, 2011.

[10] N. Anastopoulos, K. Nikas, G. Goumas, and N. Koziris, "Early experiences on accelerating Dijkstra's algorithm using transactional memory," in *Proc. IEEE Int. Parallel Distrib. Process. Symp.*, 2009, pp. 1–8.

[11] X. Liu and D. Gong, "A comparative study of A-star algorithms for search and rescue in perfect maze," in *Proc. IEEE Int. Conf. Electric Inf. Control Eng.*, 2011, pp. 24–27.

[12] S. Koenig and M. Likhachev, "D*lite," in *Proc. Nat. Conf. Artif. Intell.*, 2002, pp. 476–483.

[13] S. M. LaValle and J. James, and J. Kuffner, "Randomized kinodynamic planning," *Int. J. Robot. Res.*, vol. 20, no. 5, pp. 378–400, 2001.

[14] I. Noreen, A. Khan, and Z. Habib, "Optimal path planning using RRT* based approaches: A survey and future directions," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 11, pp. 1–16, 2016.

[15] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robot. Automat.*, vol. 12, no. 4, pp. 566–580, Aug. 1996.

[16] L. Janson, E. Schmerling, A. Clark, and M. Pavone, "Fast marching tree: A fast marching sampling-based method for optimal motion planning in many dimensions," *Int. J. Robot. Res.*, vol. 34, no. 7, pp. 883–921, 2015, doi: 10.1177/0278364915577958.

[17] H. Niu, A. Savvaris, A. Tsourdos, and Z. Ji, "Voronoi-visibility roadmap-based path planning algorithm for unmanned surface vehicles," *J. Navigation*, vol. 72, no. 4, pp. 850–874, 2019, doi: 10.1017/S0373463318001005.

[18] Z. Wang, Y. Li, H. Zhang, C. Liu, and Q. Chen, "Sampling-based optimal motion planning with smart exploration and exploitation," *IEEE/ASME Trans. Mechatron.*, vol. 25, no. 5, pp. 2376–2386, Oct. 2020.

[19] L. Jaillet, J. Cortes, and T. Simeon, "Transition-based RRT for path planning in continuous cost spaces," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 2145–2150.

[20] J. Wang, M. Q. -H. Meng, and O. Khatib, "EB-RRT: Optimal motion planning for mobile robots," *IEEE Trans. Automat. Sci. Eng.*, vol. 17, no. 4, pp. 2063–2073, Oct. 2020.

[21] G. Yang, M. Cai, A. Ahmad, C. Belta, and R. Tron, "LQR-CBF-RRT*: Safe and optimal motion planning," *arXiv*, 2023. [Online]. Available: https://arxiv.org/abs/2304.00790

[22] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, "Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 2997–3004.

[23] J. Nasir et al., "RRT*-smart: A rapid convergence implementation of RRT*," *Int. J. Adv. Robot. Syst.*, vol. 10, no. 7, 2013, Art. no. 299, doi: 10.5772/56718.

[24] S. Garrido, L. Moreno, M. Abderrahim, and D. Blanco, "FM2: A real-time sensor-based feedback controller for mobile robots," *Int. J. Robot. Automat.*, vol. 24, pp. 3169–3192, 10 2009.

[25] S. G. Loizou, "Closed form navigation functions based on harmonic potentials," in *Proc. IEEE 50th Conf. Decis. Control Eur. Control Conf.*, 2011, pp. 6361–6366.

[26] P. Vlantis, C. Vrohidis, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Robot navigation in complex workspaces using harmonic maps," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 1726–1731 .

[27] L. Fan, J. Liu, W. Zhang, and P. Xu, "Conformal navigation transformations with application to robot navigation in complex workspaces," 2022. [Online]. Available: https://arxiv.org/abs/2208.06876

[28] L. Fan and J. Liu, "Mobile robot navigation in complex polygonal workspaces using conformal navigation transformations," 2022. [Online]. Available: https://arxiv.org/abs/2208.09635

[29] J. Kim and P. Khosla, "Real-time obstacle avoidance using harmonic potential functions," in *Proc. IEEE Int. Conf. Robot. Automat.*, 1991, pp. 790–796.

[30] M. B. Horowitz and J. W. Burdick, "Optimal navigation functions for nonlinear stochastic systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 224–231.

[31] J. Amiryan and M. Jamzad, "Adaptive motion planning with artificial potential fields using a prior path," in *Proc. IEEE 3rd RSI Int. Conf. Robot. Mechatron.*, 2015, pp. 731–736.

[32] P. Vadakkepat, K. C. Tan, and W. Ming-Liang, "Evolutionary artificial potential fields and their application in real time robot path planning," in *Proc. IEEE Congr. Evol. Comput.*, 2000, pp. 256–263.

[33] C. Zhou, B. Huang, and P. Fränti, "A review of motion planning algorithms for intelligent robots," *J. Intell. Manuf.*, vol. 33, pp. 387–424, 2022.

[34] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Sci. Robot.*, vol. 5, no. 47, 2020, Art. no. eabc5986. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abc5986

[35] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Sci. Robot.*, vol. 7, no. 62, 2022, Art. no. eabk2822. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abk2822

[36] N. Rudin, D. Hoeller, M. Bjelonic, and M. Hutter, "Advanced skills by learning locomotion and local navigation End-to-End," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 2497–2503.

[37] R. E. Kalman, "Contributions to the theory of optimal control," *Bolén de la Sociedad MatematicaMexicana*, vol. 5, pp. 102–119, 1960.

[38] G. N. Saridis and C. S. G. Lee, "An approximation theory of optimal control for trainable manipulators," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, no. 3, pp. 152–159, Mar. 1979.

[39] Y. Baryshnikov, "Topological perplexity of feedback stabilization," *J. Appl. Comput. Topol.*, vol. 7, pp. 75–87, 2022.

[40] S.-B. Hsu, *Ordinary Differential Equations With Applications*. Berlin, Germany: Springer, 2013.

[41] M. Yesuf and B. Mebrate, "Properties of harmonic functions," M.Sc. Thesis Wollo Univ., Wollo Math., ETHIOPIA, 2019.

[42] S. G. Loizou and E. D. Rimon, "Correct-by-construction navigation functions with application to sensor based robot navigation," 2021. [Online]. Available: https://arxiv.org/abs/2103.04445

[43] D. E. Koditschek and E. Rimon, "Robot navigation functions on manifolds with boundary," *Adv. Appl. Math.*, vol. 11, no. 4, pp. 412–442, 1990.

[44] D. Bozkurt, A. Deliceoğlu, and T. Şengül, "Interior structural bifurcation of 2D symmetric incompressible flows," 2020. [Online]. Available: https://www.aimsciences.org/article/id/bcf91177-6baa-405d-bcd5-03e267bec32f

[45] A. K. Nandakumaran and P. S. Datti, *First-Order Partial Differential Equations: Method of Characteristics, Ser. Cambridge IISc Series*. Cambridge, U.K.: Cambridge Univ. Press, 2020, pp. 48–86.