Autonomous Robots
https://doi.org/10.1007/s10514-023-10088-7

Navigation functions with moving destinations and obstacles

Cong Wei[1] · **Chuchu Chen**[1] · **Herbert G. Tanner**[1]

Received: 30 December 2021 / Accepted: 12 January 2023

Abstract
Dynamic environments challenge existing robot navigation methods, and motivate either stringent assumptions on workspace variation or relinquishing of collision avoidance and convergence guarantees. This paper shows that the latter can be preserved even in the absence of knowledge of how the environment evolves, through a navigation function methodology applicable to sphere-worlds with moving obstacles and robot destinations. Assuming bounds on speeds of robot destination and obstacles, and sufficiently higher maximum robot speed, the navigation function gradient can be used produce robot feedback laws that guarantee obstacle avoidance, and theoretical guarantees of bounded tracking errors and asymptotic convergence to the target when the latter eventually stops moving. The efficacy of the gradient-based feedback controller derived from the new navigation function construction is demonstrated both in numerical simulations as well as experimentally.

Keywords Reactive navigation · Dynamic environments · Convergence · Non-point destinations

1 Introduction

Motion planning with obstacle avoidance is one of the oldest problems in robot navigation, with a multitude of available solutions that have been used in a wide range of applications, from typical ones involving mobile robots, manipulators, and self-driving vehicles, to more novel ones such as no-contact disinfection (Pandey et al., 2017; Tanner & Kyriakopoulos, 2000; González et al., 2015; Tiseni et al., 2021). The problem is considered solved given complete knowledge of static robot environments (Minguez et al., 2008); however, *dynamic environments*, where either the robot's target or the obstacles move, bring new unmet challenges (Mohanan & Salgoankar, 2018). The need for robot navigation in dynamic environments arises in many scenarios, including autonomous driving (Fernandes et al., 2012), (UAV) target tracking (Yao et al., 2015; Yadav et al., 2018), and human-robot interaction (Prassler et al., 2001; Li & Tanner, 2019). One major challenge for motion planning and navigation in dynamic environments is that given the temporal coupling between *path planning* and *trajectory generation* (time parameterization), the two subproblems have to be solved simultaneously and in real-time (Mohanan & Salgoankar, 2018). Existing approaches for motion planning in dynamic environments (Alonso-Mora et al., 2017; Wang et al., 2020; Qi et al., 2021) typically iteratively solve the navigation problem over time, but in doing so it is not clear how to establish *global collision avoidance and convergence guarantees*.

Several methods based on (RL) have recently appeared [e.g. Francis et al. (2020)] to address problems of intercepting and tracking moving targets in dynamic environments. Another example is a belief abstraction approach which allows for the incorporation of dynamic obstacles (Warnke et al., 2020), which however needs to assume constant obstacle speed. While more classical RL-based methods (Hasselt et al., 2016) (Q-learning, SARSA algorithm) and their variants (Xu et al., 2017) have been applied to robot navigation, it is known that such methods can suffer from overfitting problems (Vasilopoulos et al., 2020), high computational cost and no theoretical guarantee for convergence to a global optimum (Wijmans et al., 2019; Faust et al., 2018).

Solutions based on iterative graph search-based (Ajanovic et al., 2018; Ferguson & Stentz, 2006; Chen & Hwang, 1998) or sampling methods (Elbanhawi & Simic, 2014; Karaman

Cong Wei and Chuchu Chen contributed equally to this work.

✉ Cong Wei
weicong@udel.edu

Chuchu Chen
ccchu@udel.edu

Herbert G. Tanner
btanner@udel.edu

[1] Department of Mechanical Engineering, University of Delaware, 130 Academy Street, Newark 19716, DE, USA

Published online: 12 February 2023

& Frazzoli, 2011), including (RRT) Zucker et al. (2007); Fulgenzi et al. (2009); Wang et al. (2020); Qi et al. (2021) or (PRM) (Van Den Berg et al., 2005; Jaillet & Simeon, 2004) can suffer from markedly heavy computation cost when applied in high-dimensional dynamic environments (Cai et al., 2018). A significant portion of this computational overhead is associated with the need to repeatedly solve the motion planning problem as the workspace of the robot evolves (Short et al., 2016). Additional computational challenges for sampling-based methods can be traced to a range of different parameters which are not always directly controlled, including poor sampling (Francis et al., 2020). This is where feedback-based methods seem to have an advantage. These methods are often overlooked due to the possible appearance of spurious local minima that could prevent convergence; however this is only an issue in formulations involving superposition of attractive and repulsive vector fields (Olunloyo & Ayomoh, 2009; Montiel et al., 2015). However, feedback-based methods immune to this problem exist. One example is the (VO) approach, where the robot's velocity is selected from admissible sets constructed based on the obstacles' velocities (Alsaab & Bicker, 2014). Naturally, VO efficacy hinges on the accuracy of obstacle motion measurement, which can present challenges for fast loop closures (Chen et al., 2020). Another feedback-based method immune to local minima issues, is the harmonic field approach (Connolly, 1997). This approach is primarily for static environments, since a direct extension to dynamic ones would involve the iterative solution of nontrivial partial differential equations (PDEs) in real time. Still, with some prior information about the kinematics of moving obstacles (Aiushita et al., 1993; Waydo & Murray, 2003) or target (Szulczyński et al., 2011), some interesting results have been reported.

Navigation functions offer yet another option for overcoming the challenge of local minima to guarantee almost global convergence (Koditschek & Rimon, 1990; Rimon & Koditschek, 1992). Their original construction, however, is based on the assumption that the workspace is known and static. Analytical feedback-based methods have attempted to lift the assumption about a known environment by reactively using sensor measurements (Vasilopoulos et al., 2020; Arslan & Koditschek, 2019), yet they have not yet been fully extended to dynamic environments (cf. Paternain et al. (2018)). More evidence is needed to ascertain the potential efficacy of such methods in dynamic environments (Iizuka et al., 2014; Pradhan et al., 2011), and the quest for generalization of navigation function methods to fully dynamic environments reveals unmet technical challenges (Loizou et al., 2003; Sun & Tanner, 2015; Li & Tanner, 2019; Shvalb & Hacohen, 2019).

This paper meets some of the remaining challenges of feedback-based navigation in dynamic environments with moving obstacles and target, for the case of a dynamic sphere world where the robot has sufficient actuation bandwidth to respond to environment variations. The methodology expands the navigation function toolbox with a nontrivial extension of prior work that combines moving destination (Sun & Tanner, 2015) with moving obstacles (Chen et al., 2020), and derives analytic conditions on the geometric parameters of the time-varying workspace, under which the navigation function properties are uniform over time. Based on these properties a feedback law is derived, for which global collision avoidance is theoretically proven, and asymptotic convergence to the target is guaranteed when the latter eventually stops moving. This is achieved without knowledge of obstacle or target kinematics, but rather under the assumption that the robot has sufficient control authority to outmaneuver any moving obstacle. This is a significant advancement over prior work (Yadav & Tanner, 2021) that achieved asymptotic stability on the basis of known moving entity velocities.

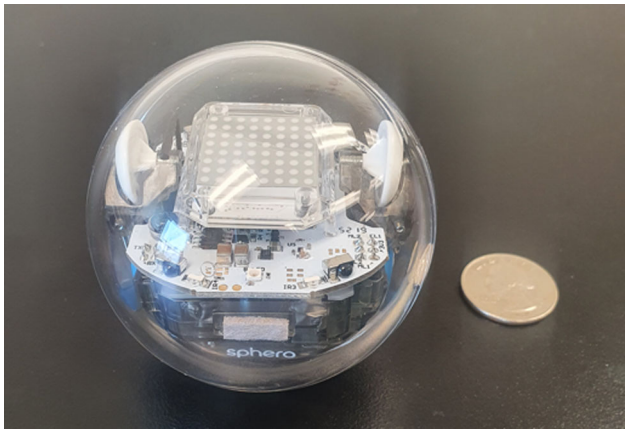


Fig. 1 The SPHERO Bolt robot

The *contributions*, therefore, of this paper are:

(a) Proof that navigation functions built on sphere worlds with time-varying destinations *and* moving internal obstacles can be tuned to be free of local minima; and
(b) Proof of asymptotic convergence to an eventually settling target for a robot steered using a time-varying navigation function and without knowledge of obstacle velocities.

These claims are corroborated in simulations and experiments with spherical mobile robots and obstacles (Fig. 1).

While the aforementioned results apply to idealized sphere world environments, they are still significant for at least two reasons:

(a) There exist methodological pathways to star-world extensions (Li & Tanner, 2019), and
(b) There can still be real-world robot navigation scenarios that conform to this model, as the one motivating this

analysis and which involves spherical robots like the one featuring in Fig. 1 engaging in child-robot play-based interaction.

In these interactions we could have one such robot playing games of chase in cluttered environments containing other toys (possibly static robot balls), exploring the scientific hypothesis that "smart" dynamic and mobile toys can engage with children better than stationary "dump" toys. The dynamic nature of the human subject which the robot should intercept and the possibility of additional moving actors in the scene (other robots or children) motivate the key features of the problem statement presented in Sect. 2. Beyond Sect. 2, which starts with some technical preliminaries, the paper is organized as follows. Section 3 presents a solution roadmap for formalizing the properties of the navigation function and proceeds to refine this roadmap with a sequence of mathematical propositions, while Sect. 4 establishes the convergence properties of a control law based on the gradient of the navigation function. Simulation and experimental results are shown in Sect. 5 validating the property of the navigation function and convergence guaranteed by the gradient control law. Section 7 concludes the paper and hints at directions for future extensions.

2 Problem formulation

2.1 Notation and preliminaries

If $\mathcal{A} \subset \mathbb{R}^n$ is a set, and $\epsilon > 0$ is a small constant, $\mathcal{A}(\epsilon)$ is used to express a *neighborhood* in the exterior $\mathcal{A}$, including its boundary $\partial\mathcal{A}$. The size of this neighborhood will be described later in terms of ϵ and a scalar function that implicitly defines $\mathcal{A}$ as one of its level sets. We denote $\mathcal{A}^{\mathsf{c}}$ the set's *complement* and $\mathring{\mathcal{A}}$ its *interior*. By writing $\bar{\mathcal{A}}$ we express the *closure* of $\mathcal{A}$, i.e., its interior combined with its boundary. The expression $\mathcal{A} \setminus \mathcal{B}$ denotes set difference, i.e., all points of $\mathcal{A}$ that are not in $\mathcal{B}$. The gradient of $f : \mathbb{R}^n \to \mathbb{R}$, denoted ∇f, is treated as a column vector, and if we need to highlight the variable with respect to which we differentiate, say x, we write it as $\nabla_x f$.

2.2 Problem statement

A point robot at configuration x, is moving omnidirectionally in a spherical workspace of radius ρ_0, centered at the origin of $\mathbb{R}^n$ and denoted $\mathcal{W}$. The workspace $\mathcal{W}$ is a spherical subset of the n-dimensional Euclidean space, defined as $\mathcal{W} \triangleq \mathbb{R}^n \setminus \mathcal{B}_0$, where $\mathcal{B}_0 \triangleq \{x : \|x\| \geq \rho_0\}$ is considered the exterior (to the workspace) surrounding obstacle.

Assumption 1 The target's and obstacles' speeds, $\dot{x}_T$ and $\dot{o}_j$, respectively, are bounded, while the (point) robot can produce a speed $\dot{x}$ of magnitude which significantly exceeds those of its target and obstacles.

The *objective* of the robot is to converge to the exterior boundary $\partial\mathcal{B}_T$ of a ball around a moving target centered at x_T and has fixed radius r_T, while avoiding collisions with the outer boundary $\partial\mathcal{B}_0$ as well as with a set of $m \geq 0$ stationary or moving spherical obstacles $\mathcal{B}_j \subset \mathcal{W}$, with fixed radii ρ_j for $j \in \{1, \ldots, m\}$. The *free workspace* of the robot is essentially $\mathcal{W}$ "punctured" by the internal obstacle spheres $\mathcal{B}_j$:

$$\mathcal{F} \triangleq \mathcal{W} \setminus \bigcup_{j\in\{1,\ldots,m\}} \mathcal{B}_j \ .$$

Since all sets of interest (robot, obstacles, workspace boundary) are assumed to be spherical, $\mathcal{F}$ is referred to as a *sphere world*.

The free workspace $\mathcal{F}$ is assumed *valid* in the sense that

(i) All obstacle and target closures are in the interior of the workspace, i.e., $\bar{\mathcal{B}}_T \subset \mathring{\mathcal{W}}$ and $\bar{\mathcal{B}}_j \subset \mathring{\mathcal{W}}$ for $j \in \{1, \ldots, m\}$; and
(ii) None of these closures intersect with one another, i.e., $\forall\, i, j \in \{0, \ldots, m\} \cup \{T\}$, $\bar{\mathcal{B}}_i \cap \bar{\mathcal{B}}_j = \varnothing$.

Assumption 2 [cf Koditschek and Rimon (1990)] Sphere world $\mathcal{F}$ is valid in the sense that the boundaries of any two spheres $\mathcal{B}_\ell$ for $\ell \in \{0, \ldots, m\} \cup \{T\}$ are at least $\delta + \sqrt{\epsilon}$ apart for some arbitrarily small $\delta > 0$.

We will prescribe the minimal necessary separation between workspace objects in Sect. 3 more formally. It is further assumed that the robot at x knows the current location and size of

(i) Every obstacle (o_j and ρ_j, respectively) and
(ii) Its target (x_T and r_T, respectively)

at time t, and that the speeds of the target and every moving obstacle are bounded (Fig. 2).

3 Navigation function properties

3.1 Overview

Partition $\mathcal{F}$ as follows.

- The set near obstacles excluding the target: $\mathcal{F}_0(\epsilon) \triangleq \bigcup_{j=1}^{m} \mathcal{B}_j(\epsilon) \setminus \bar{\mathcal{B}}_T$;

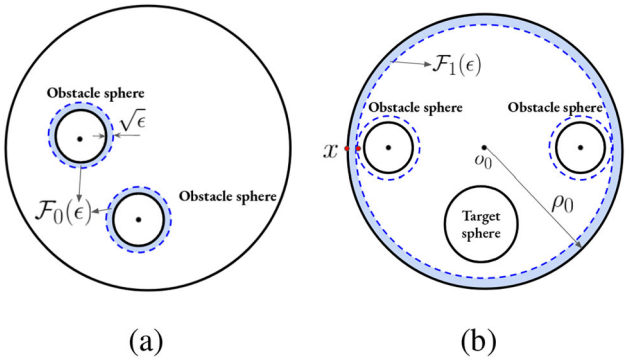


Fig. 2 Elements of workspace decomposition. **a** the set near the obstacles $\mathcal{F}_0(\epsilon)$; **b** the set near the outer workspace boundary $\mathcal{F}_1(\epsilon)$

- The set near the (outer) workspace boundary: $\mathcal{F}_1(\epsilon) \triangleq \mathcal{B}_0(\epsilon) \setminus \left(\bar{\mathcal{B}}_T \bigcup \mathcal{F}_0(\epsilon)\right)$;
- The set away from (any) workspace boundaries: $\mathcal{F}_2(\epsilon) \triangleq \left(\mathring{\mathcal{F}} \setminus \left(\mathcal{F}_0(\epsilon) \bigcup \mathcal{F}_1(\epsilon)\right)\right) \bigcup \bar{\mathcal{B}}_T$;
- The set away from any boundaries and target: $\mathcal{F}_3(\epsilon) \triangleq \mathcal{F}_2(\epsilon) \setminus \mathcal{B}_T(\epsilon)$.

The approach to the problem of Sect. 2.2 involves constructing a sphere-world *navigation function* $\varphi(x)$ (Koditschek & Rimon, 1990) and using its negated gradient to steer the robot to its objective according to a control law of the type $\dot{x} = -\nabla_x f(\varphi)$, where f is some differentiable bijective function (see Fig. 3). Function φ is parameterized by a positive (integer) constant k, which is chosen to give φ its navigation function properties. With a slight—and for the purposes of this work, inconsequential—departure from their original statement (Koditschek & Rimon, 1990) these properties are understood as follows:

Definition 1 [cf Koditschek and Rimon (1990)] A function $\varphi : \mathbb{R}_+ \times \mathcal{F} \to [0, 1]$ is a *navigation function* if it is

(i) Continuously differentiable on $\mathbb{R}_+ \times \mathcal{F}$,
(ii) Attains its minimum on $\partial\mathcal{B}_T$,
(iii) Attains its maximum on $\partial\mathcal{F}$, and
(iv) Is a Morse-Bott function on $\mathring{\mathcal{F}}$.

The mathematical roadmap for establishing the navigation function properties for φ is as follows.

1. Identify the target boundary $\partial\mathcal{B}_T$ as a non-degenerate critical submanifold of φ;
2. Establish that no critical points of φ are on $\partial\mathcal{F}$;
3. Demonstrate that with appropriate parameter selection, there can be no critical points in $\mathcal{F}_3(\epsilon)$;
4. Show that there exist an upper bound on ϵ, below which no *local minima* of φ exist in $\mathcal{F}_0(\epsilon)$;

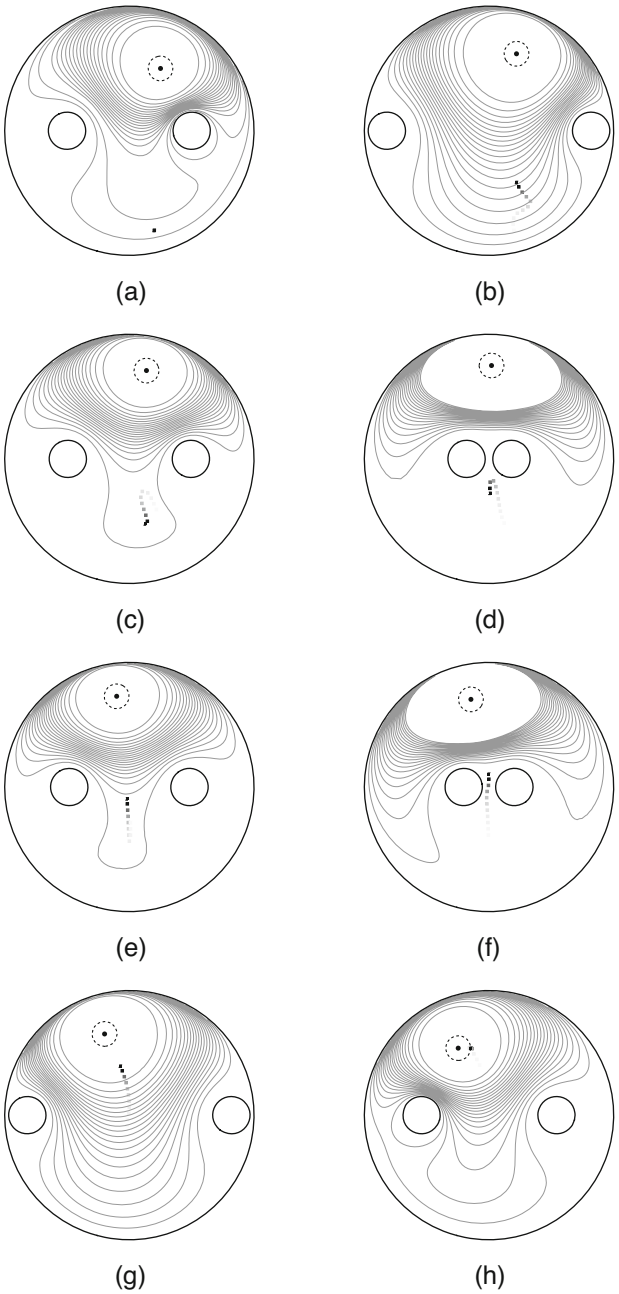


Fig. 3 A series of snapshots from a simulation study where the robot navigates using the gradient of a navigation function within a dynamic environment featuring two obstacles and a target moving on fixed, periodic trajectories

5. Prove that there exists a lower bound on k above which no critical point exist in $\mathcal{F}_1(\epsilon)$; and finally
6. Determine an appropriate choice of k for which all critical points in $\mathring{\mathcal{F}}_0(\epsilon)$ are non-degenerate.

It is known that in two dimensions any potential field defined on a manifold like $\mathring{\mathcal{F}} \setminus \bar{\mathcal{B}}_T$ will have at least as many stationary points other than the motion planning destination

as the number of interior obstacles (Koditschek & Rimon, 1990). A judicious choice of φ, however, can ensure that these stationary points are saddles with have attraction sets of measure zero. Denote $\mathcal{S}(t)$ the union of these regions of attraction of those saddles, keeping in mind that in the case considered here this set is *time-varying*. With the navigation function properties in place, and assuming that $x(0) \notin \mathcal{S}(0)$ and that the robot has sufficient control authority relative to its moving target and obstacles, we proceed to show that a control law of the form $u = -\nabla_x \varphi$, which essentially renders $\mathring{\mathcal{F}} \setminus (\bar{\mathcal{B}}_T \cup \mathcal{S}(t))$ positively invariant, also ensures that the robot tracks its target boundary with bounded error, and that if the target stops moving it will eventually asymptotically converge to it. The end result of the methodology is the emergence of navigation behaviors like the one featured in Fig. 3, where the robot reactively avoids collisions and deliberately seeks paths to its destination. More details on the scenario of Fig. 3 is found in Sect. 5. The proofs of the technical statements on the construction and properties of the navigation function φ can be found in the Appendix.

3.2 Refinement of the solution roadmap

For each spherical obstacle $\mathcal{B}_j$ define a smooth scalar function β_j that attains negative values in $\mathring{\mathcal{B}}_j$, is positive in $\mathcal{B}_j^c$ and zero $\partial\mathcal{B}_j$. For the workspace boundary $\mathcal{B}_0$ the situation is reversed: $\mathring{\mathcal{B}}_0$ maps to positive values (indicating free space) and the exterior maps to negative values. One choice for such functions is

$$\beta_j(t,x) \triangleq \|x(t) - o_j(t)\|^2 - \rho_j^2 \qquad j \in \{1,\dots,m\}$$
$$\beta_0(x) \triangleq \rho_0^2 - \|x\|^2 \ .$$

Now we are in position to define concretely the neighborhoods of workspace boundaries as follows:

$$\mathcal{B}_j(\epsilon) \triangleq \{x \in \mathcal{F} \mid 0 < \beta_j(x) < \epsilon\} \quad j \in \{0,\dots,m\} \ .$$

It can be shown (using a triangle inequality for $j \in \{1,\dots,m\}$ and a reverse triangle inequality for $j = 0$) that when $x \in \mathcal{F}_3(\epsilon)$, the robot is at least $\sqrt{\epsilon}$ away from the workspace boundary. Thus the workspace validity property "scales" with ϵ and regions near obstacle boundaries remain always within the valid workspace, i.e., $\forall\, j \in \{0,\dots,m\}$, $\mathcal{B}_j(\epsilon) \subset \mathcal{F} \setminus \mathcal{F}_3(\epsilon)$.

The surface on the boundary of the ball around the moving target, $\partial\mathcal{B}_T$, which is the destination surface of the navigation function, can be formally captured as $\{x \in \mathbb{R}^n \,|\, \|x - x_T(t)\|^2 - r_T^2 = 0\}$. As a metric of distance between the robot and its destination surface we use the goal function

$$J(t,x) \triangleq \left(\|x - x_T(t)\|^2 - r_T^2\right)^2$$

whereas a surrogate of the distance between the robot and boundary of the free space can be

$$\beta(t,x) \triangleq \beta_0(x) \prod_{j=1}^{m} \beta_j(t,x) \ .$$

The crux of the technical approach is to show first that for the function

$$\varphi(t,x) \triangleq \frac{J(t,x)}{\left[J(t,x)^k + \beta(t,x)\right]^{1/k}} \tag{1}$$

there exist a *fixed* positive real $N(\epsilon) > 0$ such that for every integer $k > N(\epsilon)$, (1) gives rise to a navigation function in the sense of Definition 1.

The following sequence of propositions codify the roadmap of Sect. 3.1. Their proofs are in the Appendix.

Proposition 1 *The target boundary, $\partial\mathcal{B}_T(t)$, is a non-degenerate critical submanifold for φ.*

Proposition 2 *All critical points of φ are in $\mathring{\mathcal{F}}$.*

Note now that φ as defined in (1), and $\hat{\varphi} \triangleq J^k/\beta$ share the same critical points, and their type is identical (Koditschek & Rimon, 1990). This fact is exploited to simplify the analysis of the critical points of φ, using $\hat{\varphi}$ in a surrogate role, as in the following proposition.

Proposition 3 *For every $\epsilon > 0$ there exists an $N(\epsilon) > 0$ such that if integer $k \geq N(\epsilon)$ there are no critical points of $\hat{\varphi} = J^k/\beta$ in $\mathcal{F}_3(\epsilon)$.*

Proposition 4 *In any valid workspace, $\exists\, \epsilon_0$ such that $\hat{\varphi} = J^k/\beta$ has no local minima in $\mathcal{F}_0(\epsilon)$, as long as $\epsilon < \epsilon_0$.*

Proposition 5 *In a valid workspace, and for any $\delta > 0$, there exist a $k_1 > 0$ such that if integer*

$$k > k_1 \triangleq \frac{2m(\rho_0 - \delta)^2}{\delta^2} \ ,$$

$\hat{\varphi}$ has no critical points in $\mathcal{F}_1(\epsilon)$.

Proposition 6 *With an appropriate choice of k, critical points x_c in the interior of $\mathcal{F}_0(\epsilon)$ are non-degenerate.*

3.3 Summary

We can summarize the bounds derived within the proof of each of the above propositions for the proximity to workspace boundary parameter ϵ and the tuning parameter k, in Tables 1 and 2, respectively. The design process that guarantees the construction of a navigation function selects ϵ in a way that respects all inequalities in Table 1, and then based on this value of ϵ, k is selected to satisfy all inequalities in Table 2.

Table 1 Summary of bounds on ϵ in different propositions. An admissible value for ϵ should satisfy the conjunction of the above conditions

Proposition	Upper bound on ϵ
Proposition 4	$\epsilon < 2\delta^4(\rho_0-\delta)^{m-1}\left[\left(\frac{r_d+\delta}{2\rho_0-r_d-\delta}\right)^2+1\right]\cdot$ $\left\{\frac{6m(\rho_0-\delta)^{m-1}(2\rho_0-r_d-\delta)^5}{\delta+r_d}+\frac{32m(\rho_0-\delta)^5}{\delta^2}\right.$ $+2(2\rho_0-r_d-\delta)^4(\rho_0-\delta)^{m-1}\left[(\rho_0-\delta)^{m-1}+(m-1)(\rho_0-\delta)^{m-2}\right.$ $\left.\left.+m(m^2-2m+2)(\rho_0-\delta)^{m-3}+(m-1)^2\right\}^{-1}\right.$
Proposition 6	$\epsilon<\frac{1-\sqrt{\frac{1+\xi^2}{2}}}{2^{2m-3}(m-1)\rho_0^{m-3}}$

Table 2 Summary of bounds on k in different propositions. For a choice of ϵ consistent with Table 1, a value for k that satisfies the conjunction of the above conditions is guaranteed to produce a navigation function

Propositions	Lower bounds on k
Proposition 3	$k \geq \frac{2(2m+1)(\rho_0-\delta)^3}{r_d(\sqrt{\epsilon}+\delta)^2}$
Proposition 5	$k \geq \frac{2m(\rho_0-\delta)^2}{\delta^2}$

4 Proof of convergence

It can be shown that with prior knowledge of the target's [cf. Sun and Tanner (2015); Yadav and Tanner (2021)] and obstacles' trajectories, an appropriately constructed control law can formally establish collision avoidance and convergence of the robot to its destination. This paper shows that even if these trajectories are unknown, with sufficient control authority the robot can track its target while avoiding collisions, and if the target stops moving, it will eventually converge to it.

Assume that the robot has the kinematics of a single integrator, or that its dynamics can be feedback linearized in this form:

$$\dot{x} = u \qquad x(0) = x_0 \in \mathcal{F} \setminus \mathcal{S}(0) \ . \tag{2}$$

Now define the control law as

$$u = -c\,\nabla\varphi(t,x) \ , \tag{3}$$

where $c > 0$ is a constant control gain.

Proposition 7 *The dynamics of the navigation function*

$$\varphi(t,x) = \frac{J(t,x)}{[J(t,x)^\kappa + \beta(t,x)]^{1/\kappa}} \ ,$$

induced by robot control law (3) away from the zero-measure attraction sets of its stationary points in a sphere world $\mathcal{F}$, is input-to-state-stable (ISS) with respect to the speeds of robot destination and obstacles.

Proof Consider the closed loop scalar dynamical system

$$\begin{aligned}\dot{\varphi}(t,x) &= \frac{\partial\varphi}{\partial t} + \left(\frac{\partial\varphi}{\partial x}\right)^\intercal u \\ &= \left(\frac{\partial\varphi}{\partial x}\right)^\intercal u + \frac{\partial\varphi}{\partial\beta}\dot{\beta} + \frac{\partial\varphi}{\partial J}\frac{\partial J}{\partial t} \ ,\end{aligned}$$

the dynamics of which, given $\frac{\partial J}{\partial t} = \frac{\partial J^\intercal}{\partial x_T}\dot{x}_T$, can be expanded in the form

$$\begin{aligned}\dot{\varphi}(t,x) = &\left\{\frac{\beta(t,x)}{[J(t,x)^\kappa+\beta(t,x)]^{1+1/\kappa}}\frac{\partial J(t,x)}{\partial x}\right. \\ &\left.-\frac{2J(t,x)\left[\sum_{j=1}^m \bar{\beta}(t,x)[x-o_j(t)]^\intercal - \prod_{j=1}^m \beta_j(t,x)\,x^\intercal\right]}{\kappa[J(t,x)^\kappa+\beta(t,x)]^{1+1/\kappa}}\right\}u \\ &+\frac{2J(t,x)\sum_{j=1}^m \bar{\beta}_j(t,x)[x-o_j(t)]^\intercal}{\kappa[J(t,x)^\kappa+\beta(t,x)]^{1+1/\kappa}}\dot{o}_j \\ &+\frac{4\beta(t,x)\sqrt{J(t,x)}}{[J(t,x)^\kappa+\beta(t,x)]^{1+1/\kappa}}[x_T(t)-x]^\intercal\,\dot{x}_T \ ,\end{aligned} \tag{4}$$

where one can note that terms multiplying $[x-o_j(t)]^\intercal\dot{o}_j$ and $[x_T-x]^\intercal\dot{x}_T$ are nonnegative in $\bar{\mathcal{F}}$. The denominator term in (4) can be recognized as

$$\left[J(t,x)^\kappa+\beta(t,x)\right]^{1+1/\kappa} = \left[\frac{J(t,x)}{\varphi(t,x)}\right]^{\kappa+1} > 0$$

for all $(t,x) \in \mathbb{R}_+ \times \bar{\mathcal{F}}$. With some algebraic manipulation, (4) can be brought to the form

$$\begin{aligned}&\left[\frac{\kappa\,[J(t,x)/\varphi(t,x)]^{\kappa+1}}{\sqrt{J(t,x)}}\right]\dot{\varphi} = \left[\frac{\kappa[J(t,x)/\varphi(t,x)]^{\kappa+1}}{\sqrt{J(t,x)}}\right]\left(\frac{\partial\varphi}{\partial x}\right)^\intercal u \\ &+2\sqrt{J(t,x)}\sum_{j=1}^m \bar{\beta}_j(t,x)[x-o_j(t)]^\intercal\dot{o}_j \\ &+4\kappa\beta(t,x)[x_T(t)-x]^\intercal\dot{x}_T \ .\end{aligned} \tag{5}$$

Based on the properties of φ as established in the proofs of Propositions 1 through 6 (see Appendices A and B), the sum of the last two terms in the right hand side of (5) is upper bounded:

$$
\begin{aligned}
&2\sqrt{J(t,x)}\sum_{j=1}^{m}\bar{\beta}_j(t,x)[x-o_j(t)]^{\intercal}\dot{o}_j\\
&\quad+4\kappa\beta(t,x)[x_T(t)-x]^{\intercal}\dot{x}_T\\
&<2\sqrt{J(t,x)}\sum_{j=1}^{m}\bar{\beta}_j(t,x)\|x-o_j(t)\|\sup_{t\geq 0}\|\dot{o}_j\|\\
&\quad+4\kappa\beta(t,x)\|x_T(t)-x\|\sup_{t\geq 0}\|\dot{x}_T\|\\
&<8\kappa\beta(t,x)\rho_0\sup_{t\geq 0}\|\dot{x}_T\|+4\sqrt{J(t,x)}\sum_{j=1}^{m}\bar{\beta}_j(t,x)\rho_0\sup_{t\geq 0}\|\dot{o}_j\|\\
&<\kappa(2\rho_0)^{2m+3}\sup_{t\geq 0}\|\dot{x}_T\|+\frac{m(4\rho_0^2-r_T^2)(2\rho_0)^{2m+2}}{2}\sup_{t\geq 0}\|\dot{o}_j\|\\
&=(2\rho_0)^{2m+2}\left[2\kappa\rho_0\sup_{t\geq 0}\|\dot{x}_T\|+\frac{m(4\rho_0^2-r_T^2)}{2}\sup_{t\geq 0}\|\dot{o}_j\|\right]\\
&<(2\rho_0)^{2m+3}\left[\kappa\sup_{t\geq 0}\|\dot{x}_T\|+m\rho_0\sup_{t\geq 0}\|\dot{o}_j\|\right]. \qquad (6)
\end{aligned}
$$

Therefore, once (3) is plugged into (5), it leads to

$$
\begin{aligned}
\dot{\varphi}&\overset{(6)}{<}-c\,\|\nabla\varphi(t,x)\|^2+\left[\frac{\varphi(t,x)}{J(t,x)}\right]^{\kappa+1}(2\rho_0)^{2m+5}\\
&\cdot\left[\kappa\sup_{t\geq 0}\|\dot{x}_T\|+m\rho_0\sup_{t\geq 0}\|\dot{o}_j\|\right]. \qquad (7)
\end{aligned}
$$

The ratio $\frac{\varphi(t,x)}{J(t,x)}=[J(t,x)^{\kappa}+\beta(t,x)]^{-1/\kappa}$ is upper and lower bounded in $\bar{\mathcal{F}}$, while $\nabla\varphi$ will always be nonzero $\forall x\notin\mathcal{S}(t)$, from which point an application of the ultimate boundedness theorem (Khalil, 2002) on the dynamics of φ establishes the existence of a $\mathcal{KL}$ class function ξ and a $\mathcal{K}$ class function γ such that

$$\varphi(t,x)\leq\xi\big(t,\varphi(0,x_0)\big)+\gamma\big(\sup_{t\geq 0}\{k\|\dot{x}_T\|+m\rho_0\|\dot{o}_j\|\}\big).$$

□

Some remarks on (7) may be in order.

Remark 1 [Convergence] The boundedness of the $\left[\frac{\varphi(t,x)}{J(t,x)}\right]^{\kappa+1}=[J(t,x)^{\kappa}+\beta(t,x)]^{-1-1/\kappa}$ term allows the selection of a sufficiently large control gain c to overcome the disturbing influence of the moving obstacles and target, which is limited via the finite values of $\sup_{t\geq 0}\|\dot{o}_j\|$ and $\sup_{t\geq 0}\|\dot{x}_T\|$, respectively.

Remark 2 [Collision avoidance] Similarly, collision avoidance under (3) is not unconditional; it relies on selecting a gain c large enough to enable the robot to overcome the agility of dynamic obstacles and target.

Remark 3 [Saddle points] While it can be readily verified a priori whether the robot's position is in $\mathcal{S}(t)$ at initial time, the theoretical *possibility* that $x(t)$ intersects with $\mathcal{S}(t)$ as the latter moves inside the workspace over time, cannot be eliminated. That said, given that $\mathcal{S}(t)$ is of zero measure and time-varying, the *probability* that $x(t)\in\mathcal{S}(t)\ \forall\, t>\tau>0$ is practically zero.

In light of these observations and armed with Proposition 7, we move to the following claim:

Proposition 8 *If the target eventually stops ($\dot{x}_T=0$) and with the conditions of Proposition 7 in force, there is a sufficiently large control gain $c>0$ to ensure the point robot (2) under (3) asymptotically converges to its target.*

Proof Assuming $\dot{x}_T=0$ after some time $\tau>0$ (and given that there is no memory in the system which means that $x(\tau)$ can be considered a new initial condition), (7) can be restated without its $\dot{x}_T$ term, establishing ISS of $\varphi(t,x)$ with respect to $\sup_{t\geq\tau}\|\dot{o}_j\|$. At this point, ultimate boundedness arguments can establish that for a sufficiently large $c>0$ (the existence of which is predicated on Assumption 1), $\varphi(x,t)$ can reach in finite time an arbitrarily small value $\varepsilon>0$, after which time the state $x(t)$ will stay within the sublevel set $\{x\in\mathcal{F}:\varphi(x)<\varepsilon\}$. Continuity now suggests that if this ε is sufficiently small, the sublevel set $\{x\in\mathcal{F}:\varphi(x)<\varepsilon\}$ will be contained in a small neighborhood of $\mathcal{B}_T$. Notice, however, that (validity) Assumption 2 now forces $\mathcal{B}_T$ to always be at least some $\delta+\sqrt{\epsilon}$ away from any obstacle $\mathcal{B}_j$ for $j\in\{0,\ldots,m\}$. As a result, for sufficiently small ε and for $c>0$ adequately large to ensure finite-time positive invariance of the sublevel set $\{x\in\mathcal{F}:\varphi(x)<\varepsilon\}$, the $\mathcal{B}_T$ neighborhood that contains the latter will be disjoint from any $\mathcal{B}_j$ for $j\in\{0,\ldots,m\}$.

The significance of this fact is the realization that using a sufficiently large control gain, one can send the robot where the obstacles cannot follow: once inside this sublevel set $\{x\in\mathcal{F}:\varphi(x)<\varepsilon\}$, the robot has a straight shot to its target which cannot be disrupted by the motion of the obstacles. While not evident from (7), this view can be justified by (4) when $\dot{x}_T=0$ (as assumed). With the robot out of the obstacles' reach (β_j lower bounded in the sublevel set) and capable of moving faster than them, there is only a finite time during which the obstacles can maintain $[x-o_j(t)]^{\intercal}\dot{o}_j>0$. After that time, the second term in (4) can also be dropped:

$$\dot{\varphi}=-c\,\|\nabla\varphi\|^2\ , \qquad (8)$$

establishing asymptotic convergence for φ. □

5 Numerical validation

Figure 3 illustrates in the form of snapshots how the time-varying navigation function can steer the (point mass) system to its moving destination among two spherical obstacles that oscillate thus varying the width of the allowable pathways between them.

The figure is read from top left to bottom right; the initial position for the robot (see Fig. 3a) is marked as a point in the bottom region of the spherical outer boundary of the environment, whereas the moving destination is depicted as the point at the upper portion, surrounded by a small (target) sphere that the robot needs to converge to.

The target transcribes a circular motion within the top region of the work space (above the two moving obstacles). These two obstacles, arranged along the horizontal diameter of the outer boundary, move back and forth in an oscillatory motion on this line, with the same frequency and speed and varying distance between them. The sequence of snapshots in the figure depict the robot starting to move upward toward its destination, attempting at first to pass in between the two oscillating obstacles (Fig. 3b). Then the gap closes as the two moving obstacles approach each other and the robot fails to pass through in two consecutive attempts (Fig. 3c, d) and backtracks. Then, being closer now and with the right timing a gap opens (Fig. 3e), and the robot "sees its chance" to make another run between them (Fig. 3f). Once through (Fig. 3g) it has a clear shot to its goal (Fig. 3h).

In another scenario, depicted in Fig. 4, the obstacles are coordinated deliberately to exhibit a "malicious" behavior with respect to the robot, swarming around it in an attempt to prevent it from reaching its target. In this scenario, therefore, the obstacles do not follow fixed trajectories, but adapt their motion reactively in response to the motion of the robot and the target. Specifically, the obstacles move under the effect of swarming cohesion & separation artificial forces, generated as a negated gradient of an inter-agent potential function of the relative distance d_{ij} between agents i and j having the form $V(d_{ij}) = 1/(\nu_1 d_{ij})^2 + \log^2(\nu_2 d_{ij})$ (Tanner et al., 2003, 2005). The arrows marked on the obstacles in Fig. 4 denote the direction of these artificial swarming forces. The swarm interaction network between obstacles, target, and robot is fully connected, but target and robot are indifferent to the swarming interaction. In this scenario, the robot's maximum speed surpasses that of the obstacles by a factor of 3. As the sequence of snapshots in Fig. 4 indicate, despite their intend, the obstacles are not fast enough to block the robot which outmaneuvers their cluster from the right.

Things can become more challenging, however, when the relative actuation capacity between robot and obstacles approaches unity. In an otherwise identical scenario to that of Fig. 4, where the ratio of maximum speeds between robot and obstacles falls to 2, the obstacles are able to move in a coordinated fashion to block the robot from reaching its target (Fig. 5). (Although this scenario was not simulated for longer time horizons than those of Figs. 3 and 4 to explore if the robot eventually finds a successful circumnavigation strategy.) Videos of the aforementioned scenarios are available at https://udspace.udel.edu/handle/19716/31417.

6 Experimental validation

The experimental testbed mirrors the simulation setup and consists of a (virtual) outer workspace boundary marked by black tape in Fig. 6a with $\rho_0 = 150$ cm, one static obstacle, one moving obstacle, and a moving target.

The Sphero™ robots used in this experiment are basically differential drive vehicles with ability to turn in place. They are controlled at a kinematic level using an API that prescribes speed and bearing directives, accompanied by underlying pid controllers that steer the vehicle to track these references. In this sense, the kinematics of Sphero™ can be assumed to adhere to the equations of a unicycle:

$$\dot{x} = v\cos\theta \qquad \dot{y} = v\sin\theta \qquad \dot{\theta} = \omega\ .$$

Given that only the (x, y) position is of interest here, we can reasonably consider an input-output feedback linearization process that would render the above kinematics equivalent to those of a single integrator.

The obstacles and the destination are spherical robots; the moving obstacle and the target are realized by Sphero™ BOLT robots, while the static obstacle and the interceptor robot are realized by a Sphero™ SPRK+ and a Sphero™ 2, respectively; all these spheres have equal radii, and the volume of the robot was taken into account when modelled as a point by doubling the radius the control algorithm uses for the other spheres to $\rho_1 = \rho_2 = 20$ cm. The target sphere around the destination has radius $r_d = 25$ cm. The objects are distinguished by their color LED signature; the target emits blue light, while the navigating robot emits red in Fig. 6a. The moving obstacle and the destination follow circular trajectories, which are unknown to the robot. The location of every object is determined in real time via color detection using a light-tracking overhead RealSense D415 camera.

The latter's limitations is the reason for the low (ambient light) exposure in Fig. 6a. Control and communication is facilitated through the Sphero Multi-Agent Robotic Testbed, while the robots' kinematic commands are computed in MATLAB and relayed through bluetooth. Figure 6a depicts the initial workspace and robot configuration, with the robot (red light) on the bottom right area of the workspace—the outer boundary of which is marked with a black tape, and the two moving objects (blue target, green obstacle) tracking

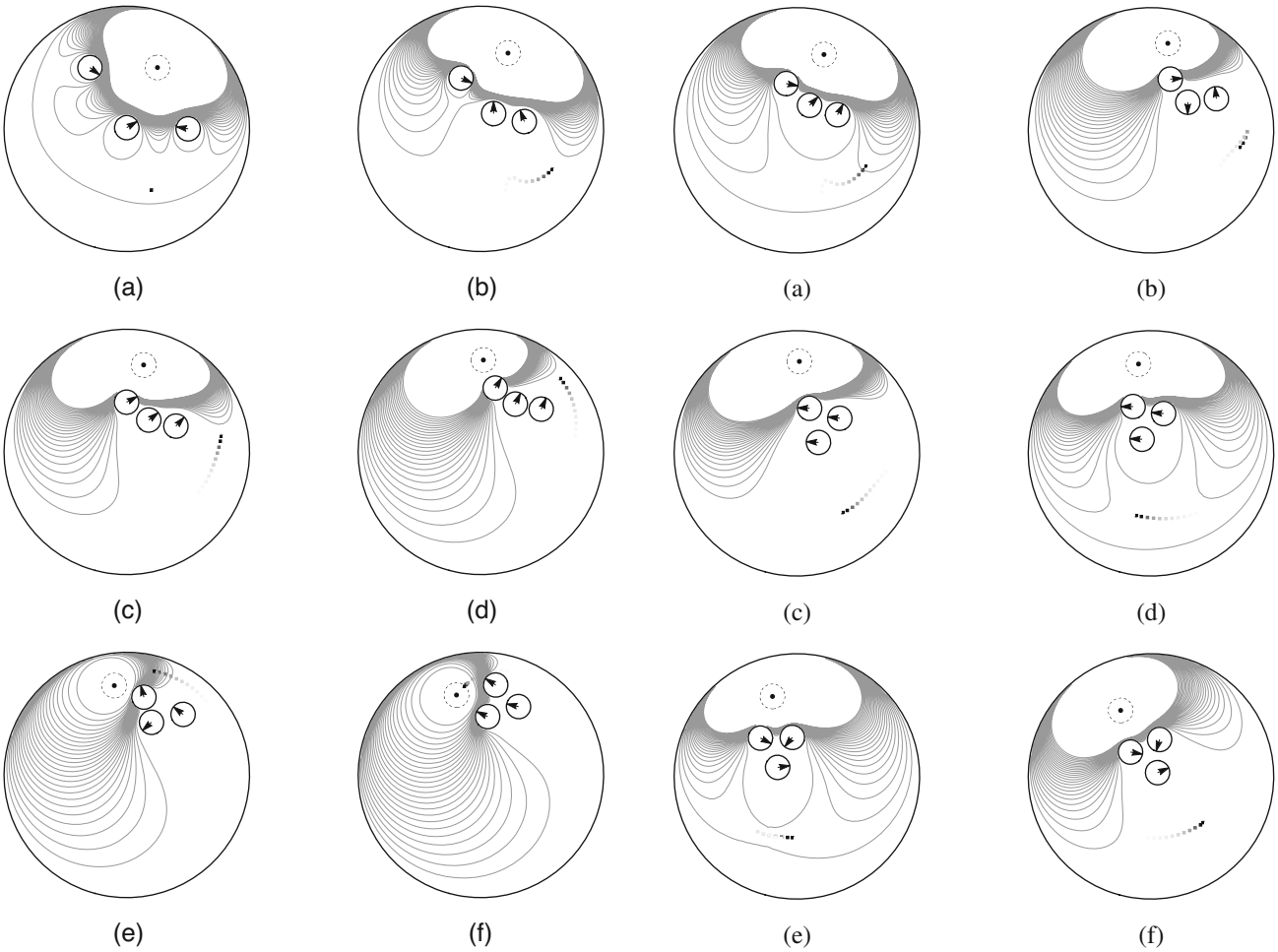


Fig. 4 A series of snapshots from a simulation study featuring two obstacles attempting to swarm around robot and target. Despite the adversarial action, the robot still reaches its destination

Fig. 5 A series of snapshots from a simulation study featuring two obstacles attempting to swarm around robot and target. In this case, because the robot cannot move fast enough relative to the obstacles, the latter succeed in blocking its path to its destination

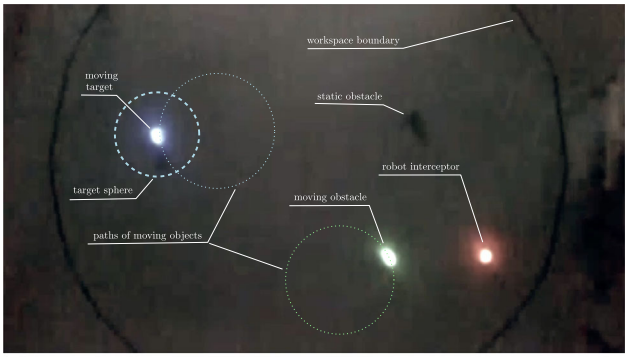
circular trajectories outlined by thin dotted curves. None of these trajectories is known to the robot.

The thicker circle around the target marks the surface of $\mathcal{B}_T$ around it, while a static obstacle is visible on the top right. Figure 6b gives a snapshot of the steady state, where the robot has navigated between the green moving and black static obstacles, reached the surface of the target sphere and tracks it as the target moves along its circular trajectory. A video of the experimental run described is available at https://udspace.udel.edu/handle/19716/31417.

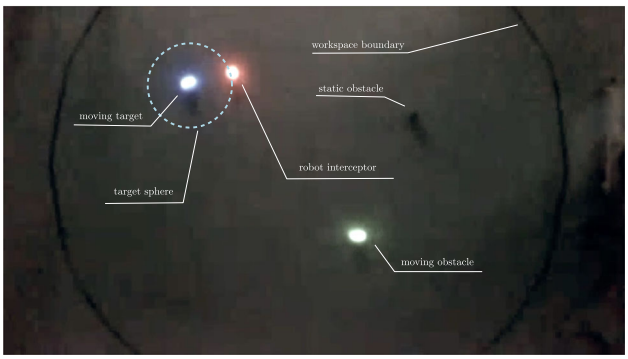
Figure 7 shows the evolution of the value of the navigation function as the experiment of Fig. 6 progresses over a time window of one minute. The robot is initially close —taking into account its volume, and given the parameter tuning applied— to the outer workspace boundary, which is why value for φ starts very close to its maximum, and then very quickly decreases to its minimum of zero. The temporary intermittent small increases evident in the graph of φ are due to a combination of motion noise and use of raw and unfiltered measurements of the objects' positions directly from the overhead color tracking system. Steady state is practically reached within 8 seconds, and in the remaining time the robot tracks the target as the latter goes around its circular path, remaining close to $\partial\mathcal{B}_T$.

7 Conclusions and future work

There has been anecdotal evidence that the navigation function methodology can also be effective in time-varying environments. This paper mathematically establishes the truth of this conjecture through a series of propositions, the proof of which offer explicit (albeit conservative) uniform bounds on the function's parameters. The paper accompanies the proof of correctness of the time-varying sphere world navigation function with a proof of convergence for a negated gradient-based control law. While the paper addresses the



(a) Initial configuration



(b) Steady state

Fig. 6 Snapshots from experimental implementation

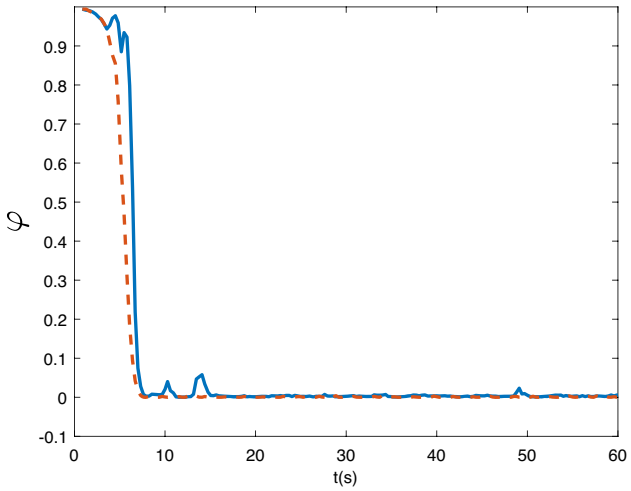


Fig. 7 Evolution of the value of the navigation function during the experiment with the boundary conditions of Fig. 6. Blue solid curve gives the value of φ based on (unfiltered) position measurements; red dashed curve indicates the expected evolution of φ in controlled simulation conditions

key but idealized case of navigation in time-varying sphere worlds where both destination and obstacles may be moving, there has already been work that paves the road toward extensions to star worlds (Li & Tanner, 2019) and multi-robot systems (Yadav & Tanner, 2021).

Acknowledgements This work was supported in part by NSF through award no 2014264.

Declarations

Conflict of interest We certify that there is no actual or potential conflict of interest in relation to this article.

Appendix A Useful expressions

A.1 Gradients and Hessians for the obstacle function β

For $\beta_0(x)$ the gradient is found directly as:

$$\beta_0 = \rho_0^2 - \|x\|^2 \implies \nabla\beta_0 = -2x \tag{A1}$$

For $\beta_j(t,x)$ with $j \in \{1 \dots m\}$ it is

$$\begin{aligned}\beta_j(t,x) &= \|x - o_j(t)\|^2 - \rho_j^2 \\ &\implies \nabla\beta_j = 2[x - o_j(t)]\end{aligned} \tag{A2}$$

The *omitted product* for an obstacle function $\beta_j(t,x)$ is defined as Koditschek and Rimon (1990):

$$\bar{\beta}_j \triangleq \textstyle\prod_{l=0, l\neq j}^{m} \beta_l$$

allowing the decomposition $\beta = \beta_j\bar{\beta}_j$. Thus the gradient of β is found directly as:

$$\begin{aligned}\nabla\beta(t,x) &\overset{(A2,A1)}{=} \\ &-2x\bar{\beta}_0(x) + 2\sum_{j=1}^{m}[x - o_j(t)]\bar{\beta}_j(t,x)\ .\end{aligned} \tag{A3}$$

Based on the above, the fact that no obstacle radius is larger than ρ_0, and given the workspace validity assumption, one can derive the bound

$$\|\nabla\beta(t,x)\| \leq 2(m+1)(\rho_0 - \delta)^m\ . \tag{A4}$$

Noting that $\nabla^2\beta_j(t,x) = 2I$ where I is the identity matrix, for any $j \in \{0, m\}$ the Hessian of β expands as

$$\nabla^2\beta(t,x) = \nabla\beta_j\nabla\bar{\beta}_j^{\mathsf{T}} + 2\bar{\beta}_j I + \nabla\bar{\beta}_j\nabla\bar{\beta}_j^{\mathsf{T}} + \beta_j\nabla^2\bar{\beta}_j\ .$$

Based on the above expression, after taking norms, expanding, and using recursion on $\|\nabla^2\bar{\beta}_j\|$, one can arrive at

$$\|\nabla^2\beta\| \leq 2(\rho_0-\delta)^m\big[(\rho_0-\delta)^m + m(\rho_0-\delta)^{m-1} + (m+1)(m^2+1)(\rho_0-\delta)^{m-2} + m^2\big] , \tag{A5}$$

which holds everywhere in $\mathcal{F}_2(\epsilon)$.

A.2 Gradient and Hessian of $\hat{\varphi}$

The gradient of $\hat{\varphi} = J^k/\beta$ can be written as

$$\nabla\hat{\varphi} = \frac{kJ^{k-1}}{\beta^2}\left(\beta\nabla J - \frac{1}{k}J\nabla\beta\right) . \tag{A6}$$

At a critical point x_c, $\hat{\varphi}$ needs to satisfy $\nabla\hat{\varphi}|_{x_c} = 0$, meaning

$$\beta\,\nabla J|_{x_c} - \tfrac{1}{k}J\,\nabla\beta|_{x_c} = 0 . \tag{A7}$$

It is noteworthy here that at a critical point of $\hat{\varphi}$, the gradient of J is in the same direction as that of β:

$$\nabla J|_{x_c} \overset{(A7,A3)}{=} -2\bar{\beta}_0 x_c + \frac{2J}{k\beta}\sum_{j=1}^{m}(x_c - o_j)\bar{\beta}_j . \tag{A8}$$

On the other hand, the Hessian

$$\nabla^2\hat{\varphi} = (k\nabla J\beta - J\nabla\beta)\,\nabla\left(\frac{J^{k-1}}{\beta^2}\right)^{\intercal} + \frac{J^{k-1}\left(k\nabla J\nabla\beta^{\intercal} + k\beta\nabla^2 J - \nabla\beta\nabla J^{\intercal} - J\nabla^2\beta\right)}{\beta^2} ,$$

when evaluated at x_c given (A7), reduces to

$$\nabla^2\hat{\varphi}|_{x_c} = \frac{J^{k-1}}{\beta^2}\Big(k\nabla J\,\nabla\beta^{\intercal} + k\beta\,\nabla^2 J - \nabla\beta\,\nabla J^{\intercal} - J\,\nabla^2\beta\Big)|_{x_c} . \tag{A9}$$

A.3 Gradient and Hessian of the goal function J

At an arbitrary $x \in \mathcal{F}$, the gradient and Hessian of the goal function J, are expressed, respectively, as

$$\nabla J(t,x) = \begin{cases} 4\sqrt{J}\,[x - x_T(t)] & x \notin \mathcal{B}_T \\ -4\sqrt{J}\,[x - x_T(t)] & x \in \mathcal{B}_T \end{cases} \tag{A10}$$

and given the workspace validity assumption, for $x \notin \mathcal{B}_T$

$$\|\nabla J(t,x)\| \leq 4(2\rho_0 - r_T - \delta)^3 . \tag{A11}$$

For the Hessian of the goal function we have

$$\nabla^2 J(t,x) = \begin{cases} 4\sqrt{J}\,I + 8[x - x_T(t)][x - x_T(t)]^{\intercal} & x \notin \mathcal{B}_T \\ -4\sqrt{J}\,I - 8[x - x_T(t)][x - x_T(t)]^{\intercal} & x \in \mathcal{B}_T \end{cases} . \tag{A12}$$

Similarly, for $x \notin \mathcal{B}_T$ the norm of this matrix is uniformly upper bounded as

$$\|\nabla^2 J(t,x)\| \leq 12(2\rho_0 - r_T - \delta)^2 . \tag{A13}$$

Appendix B Proofs

B.1 Proposition 1

Proof We first show that a vector $\boldsymbol{v}$ satisfying $\boldsymbol{v}^{\intercal}\nabla^2\varphi|_{x_D}\boldsymbol{v} = 0$ is tangent to $\partial\mathcal{B}_T$. For $x_D(t) \in \partial\mathcal{B}_T$, it holds that $\|x_D(t) - x_T(t)\|^2 - r_T^2 = 0$, implying that J and ∇J both vanish. Therefore, the gradient of φ at x_D, written as (explicit dependence of terms on x and t is dropped for brevity),

$$\nabla\varphi|_{x_D} = \left.\frac{(J^k+\beta)^{1/k}\nabla J - J\nabla(J^k+\beta)^{1/k}}{(J^k+\beta)^{2/k}}\right|_{x_D} = 0$$

indicating that x_D is a critical point for φ; given that $\varphi(t,x) \geq 0$ and $\varphi(t,x_D) = 0$, x_D is a minimum. The Hessian of $\varphi(t,x)$ evaluated at x_D is expressed as

$$\begin{aligned}\nabla^2\varphi|_{x_D} &= \left.\frac{\nabla\left(\nabla J(J^k+\beta)^{1/k} - J\nabla(J+\beta)^{1/k}\right)}{(J^k+\beta)^{2/k}}\right|_{x_D} \\ &\quad + \left.\left((J^k+\beta)^{1/k}\nabla J - J\,\nabla(J+\beta)^{1/k}\right)\left(\nabla(J^k+\beta)^{-2/k}\right)^{\intercal}\right|_{x_D} \\ &= 8\beta^{-1/k}(x_D - x_T)(x_D - x_T)^{\intercal}\end{aligned}$$

from which it follows that

(a) $\nabla^2\varphi|_{x_D}$ is singular, and
(b) The quadratic form $v^{\intercal}\nabla^2\varphi|_{x_D}v$ is zero for any $v \perp (x_D - x_T)$, where in $x_D - x_T(t)$ we identify the radial vector from the surface to the center of $\mathcal{B}_T(t)$. Now $x_D - x_T(t)$ is in fact along the normal direction to $\partial\mathcal{B}_T(t)$ given that $\mathcal{B}_T$ is a sphere; for any $\boldsymbol{w} = \lambda[x_D - x_T(t)]$, $\lambda \in \mathbb{R}$, notice that $\boldsymbol{w}^{\intercal}\nabla^2\varphi|_{x_D}\boldsymbol{w} = 8\lambda^2\beta^{-1/k} \neq 0$ and thus the Hessian is nondegenerate in a direction normal to $\partial\mathcal{B}_T(t)$.

□

B.2 Proposition 2

Proof The proposition is falsified only when a critical point exists in $\partial\mathcal{F}$. We can show that this cannot happen. Since $\mathcal{B}_j$ for $j \in \{0, \dots, m\}$ do not intersect in a valid $\mathcal{F}$, any $x_0 \in \partial\mathcal{F}$ will necessarily be on a single $\partial\mathcal{B}_\ell$, for some $\ell \in \{0, \dots, m\}$. Then $\beta_\ell(t, x_0) = 0$ but $\nabla\beta_\ell|_{x_0} = 2[x_0 - o_l(t)] \neq 0$, while $\forall j \in \{0, \dots, m\} \setminus \{\ell\}$, $\beta_j(t, x_0) > 0$. Then, $\nabla\varphi|_{x_0}$ reduces to

$$\begin{aligned}\nabla\varphi|_{x_0} &= \left.\frac{(J^k+\beta)^{1/k}\nabla J - J\,\nabla(J^k+\beta)^{1/k}}{(J^k+\beta)^{2/k}}\right|_{x_0} \\ &= \left.\frac{(J^k+\beta)\nabla J - \frac{J}{k}\left(kJ^{k-1}\nabla J + \nabla\beta\right)}{(J^k+\beta)^{\frac{1}{k}+1}}\right|_{x_0} \overset{\beta=0}{=} \\ &\quad -\frac{J^{-k}\nabla\beta|_{x_0}}{k} \overset{\beta_\ell=0}{=} -\frac{J^{-k}}{k}\prod_{j=0, j\neq\ell}^{m}\beta_j(t,x_0)\nabla\beta_\ell|_{x_0} \neq 0\,.\end{aligned}$$

□

B.3 Proposition 3

Proof Before we begin, it may be worthwhile to recall some implications of Assumption 2:

(i) The robot is in the interior of $\mathcal{B}_0^{\mathsf{c}}$ (whose center is the default origin of $\mathcal{F}$) and $\sqrt{\epsilon}+\delta$ away from its boundary, i.e., $\|x - o_0\| = \|x\| \leq \rho_0 - \sqrt{\epsilon} - \delta$;
(ii) The distance of any obstacle center from the origin of the workspace is upper bounded $\forall\, t \geq 0$, i.e., $\|o_j(t)\| \leq \rho_0 - \rho_j - \sqrt{\epsilon} - \delta$
(iii) The distance between the robot and $\mathcal{B}_T$ is upper and lower bounded $\forall\, t \geq 0$, i.e., $r_T \leq \|x - x_T(t)\| \leq 2\rho_0 - \sqrt{\epsilon} - \delta - r_T$; and
(iv) The distance between the robot and any interior obstacle center is upper and lower bounded, i.e., $\rho_j < \|x - o_j(t)\| \leq 2(\rho_0 - \sqrt{\epsilon} - \delta) - \rho_j$.

At a critical point $x \in \mathcal{W}(\epsilon)$ of $\hat{\varphi}$ it is necessary that $\beta\nabla J - \frac{1}{k}J\nabla\beta = 0$, meaning that if $k > \frac{J\|\nabla\beta\|}{\beta\|\nabla J\|}$, x cannot be a critical point. The strategy therefore is to set k sufficiently high so as to preclude the possibility of critical points in $\mathcal{W}(\epsilon)$. For this, it suffices to have

$$\mathrm{k} \geq \sup_{\mathcal{W}(\epsilon), t\geq 0}\frac{J}{\|\nabla J\|}\sup_{\mathcal{W}(\epsilon), t\geq 0}\frac{\|\nabla\beta\|}{\beta}\,.$$

Let $x \in \mathcal{W}(\epsilon)$, and suppose $\mathcal{B}_j$ is the obstacle closest to x. For $x \in \mathcal{F}_2(\epsilon) \subset \mathcal{W}(\epsilon)$, it holds that $\beta_j(t, x) \geq \epsilon$, so

$$\begin{aligned}&\sup_{\mathcal{F}_2(\epsilon)}\frac{J(t,x)}{\|\nabla J(t,x)\|} \overset{(A10)}{=} \sup_{\mathcal{F}_2(\epsilon)}\frac{\|x - x_T(t)\|^2 - r_T^2}{4\|x - x_T(t)\|} \\ &\leq \quad \frac{(\rho_0-\delta)^2}{r_T}\,,\end{aligned} \tag{B14}$$

and similarly

$$\begin{aligned}&\sup_{\mathcal{F}_2(\epsilon)}\frac{\|\nabla\beta\|}{\beta} \leq \sup_{\mathcal{F}_2(\epsilon)}\sum_{j=0}^{m}\frac{\|\nabla\beta_j\|}{\beta_j} = \sup_{\mathcal{F}_2(\epsilon)}\frac{2\|x\|}{\beta_0(x)} \\ &+ \sup_{\mathcal{F}_2(\epsilon)}\sum_{j=1}^{m}\frac{2\|x - o_j(t)\|}{\beta_j(t,x)} \leq \frac{2}{(\sqrt{\epsilon}+\delta)^2}\sup_{\mathcal{F}_2(\epsilon)}\Big\{\|x\| \\ &+ \sum_{j=1}^{m}\|x - o_j(t)\|\Big\} \leq \frac{2(2m+1)(\rho_0-\delta)}{(\sqrt{\epsilon}+\delta)^2}\,.\end{aligned} \tag{B15}$$

Denote $N(\epsilon)$ the product of the suprema in (B14) and (B15):

$$N(\epsilon) \triangleq \frac{2(2m+1)(\rho_0-\delta)^3}{r_T(\sqrt{\epsilon}+\delta)^2}\,.$$

Now if $k \geq N(\epsilon)$, x cannot be a critical point. □

B.4 Proposition 4

Proof A sufficient condition for a critical point of $\hat{\varphi}$ *not* to be a local minimum, is for the Hessian of $\hat{\varphi}$ evaluated there to have at least one negative eigenvalue. Demonstrating the existence of a negative eigenvalue essentially amounts to showing that there exists a vector $\boldsymbol{v} \in \mathbb{R}^n$ such that $\boldsymbol{v}^\intercal\nabla^2\hat{\varphi}\boldsymbol{v} < 0$.

Toward this end, denote $x_c \in \mathcal{F}_0(\epsilon)$ the critical point of $\hat{\varphi}$ in question, and expand the gradient of the obstacle function $\beta(t, x)$ evaluated at x_c, using omitted products:

$$\begin{aligned}\nabla\beta|_{x_c} &= \sum_{l=1}^{m}2[x_c - o_l(t)]\bar{\beta}_l(t,x_c) - 2\bar{\beta}_0(x_c)\,x_c \\ &= 2(x_c - o_j)\bar{\beta}_j + \beta_j\underbrace{\left[2\sum_{l=1,l\neq j}^{m}(x_c - o_l)\frac{\bar{\beta}_l}{\beta_j} - 2\frac{\bar{\beta}_0}{\beta_j}x_c\right]}_{\alpha_j} \\ &= 2(x_c - o_j)\bar{\beta}_j + \beta_j\alpha_j\,,\end{aligned} \tag{B16}$$

where we dropped the dependence of terms on t for brevity. Now, given that $x_c \notin \mathcal{B}_T$ is a critical point, and that $k\,\beta\,\nabla J = J\,\nabla\beta$, it follows

$$\begin{aligned}\nabla J &\overset{(A10)}{=} 4\left(\|x_c - x_T\|^2 - r_T^2\right)[x_c - x_T] \\ &\overset{(A8)}{=} \frac{J}{k\beta}\nabla\beta \overset{(B16)}{=} \frac{J[2(x_c - o_j)\bar{\beta}_j + \beta_j\alpha_j]}{k\beta}\,,\end{aligned}$$

which one can manipulate to arrive at

$$x_c - x_T = \frac{\sqrt{J}}{2k}\left(\frac{x_c - o_j}{\beta_j} + \frac{\alpha_j}{2\bar{\beta}_j}\right)\,. \tag{B17}$$

Then, multiplying both sides of (A7) with $\nabla J^\intercal$ and expanding β using the omitted product of β_j, yields

$$k\beta \,\|\nabla J\|^2|_{x_c} = J\,\nabla\beta^\intercal\,\nabla J|_{x_c} \implies$$
$$k\beta = \frac{\bar\beta_j \nabla\beta_j^\intercal \nabla J + \beta_j \nabla\bar\beta_j^\intercal \nabla J}{16\|x_c - x_T(t)\|^2}\ . \tag{B18}$$

Now in order for x_c *not* to be a local minimum, it suffices to show that there exists some vector $\boldsymbol{v}$ so that $\boldsymbol{v}^\intercal\nabla^2\varphi|_{x_c}\boldsymbol{v} < 0$ when ϵ is set sufficiently small. Indeed, take $\boldsymbol{v}$ to be a unit vector orthogonal to $\nabla\beta_j$. Then,

$$\begin{aligned}
&\frac{\beta^2}{J^{k-1}}\boldsymbol{v}^\intercal\nabla^2\varphi|_{x_c}\boldsymbol{v}\\
&\overset{(A7)(A9)}{=} J\beta_j\tfrac{1-\frac{1}{k}}{\bar\beta_j}\boldsymbol{v}^\intercal\nabla\bar\beta_j\nabla\bar\beta_j^\intercal\boldsymbol{v} + k\beta\boldsymbol{v}^\intercal\nabla^2 J\boldsymbol{v}\\
&-2J\bar\beta_j - J\beta_j\boldsymbol{v}^\intercal\nabla^2\bar\beta_j\boldsymbol{v}\\
&\overset{(B18)}{=} \frac{\bar\beta_j\nabla\beta_j^\intercal\nabla J + \beta_j\nabla\bar\beta_j^\intercal\nabla J}{16\|x_c - x_T(t)\|^2}\boldsymbol{v}^\intercal\nabla^2 J\boldsymbol{v} - 2J\bar\beta_j +\\
&\boldsymbol{v}^\intercal J\beta_j\left[\tfrac{1-\frac{1}{k}}{\bar\beta_j}\nabla\bar\beta_j\nabla\bar\beta_j^\intercal - \nabla^2\bar\beta_j\right]\boldsymbol{v}\\
&= \beta_j\underbrace{\left[\tfrac{\nabla\bar\beta_j^\intercal\nabla J\,\boldsymbol{v}^\intercal\nabla^2 J\boldsymbol{v}}{16\|x_c - x_T(t)\|^2} + J\boldsymbol{v}^\intercal\left(\tfrac{1-\frac{1}{k}}{\bar\beta_j}\nabla\bar\beta_j\nabla\bar\beta_j^\intercal - \nabla^2\bar\beta_j\right)\boldsymbol{v}\right]}_{B}\\
&+\bar\beta_j\underbrace{\left[\frac{\boldsymbol{v}^\intercal\nabla^2 J\boldsymbol{v}}{16\|x_c - x_T(t)\|^2}\nabla\beta_j^\intercal\nabla J - 2J\right]}_{A}\ .
\end{aligned}\tag{B19}$$

Now expand the term $\boldsymbol{v}^\intercal\nabla^2 J\boldsymbol{v}$ in A into

$$\begin{aligned}
&\boldsymbol{v}^\intercal\nabla^2 J\boldsymbol{v} \overset{(A12)}{=} 4\left(\|x_c - x_T(t)\|^2 - r_T^2\right)\\
&+8\left(\boldsymbol{v}^\intercal[x_c - x_T(t)]\right)^2 \overset{(B17)}{=} 4\sqrt{J} + \frac{J|\boldsymbol{v}^\intercal\alpha_j|^2}{2k^2\bar\beta_j{}^2}\ ,
\end{aligned}$$

then substitute back

$$\begin{aligned}
A &= \frac{4\sqrt{J} + \frac{J|\boldsymbol{v}^\intercal\alpha_j|^2}{2k^2\bar\beta_j{}^2}}{16\|x_c - x_T\|^2}[8\sqrt{J}(x_c - o_j)^\intercal(x_c - x_T)] - 2J\\
&= \frac{2J + \frac{J^{\frac{3}{2}}|\boldsymbol{v}^\intercal\alpha_j|^2}{(2k\bar\beta_j)^2}}{\|x_c - x_T\|^2}(x_c - o_j)^\intercal(x_c - x_T) - 2J\ .
\end{aligned}$$

It is known (Koditschek and Rimon (1990), Lemma 3.5) that for *any* x_T and o_j in a valid workspace, the critical point x_c satisfies

$$(x_c - x_T)^\intercal(x_T - o_j) \le \|x_T - o_j\|\left[\sqrt{\epsilon + \rho_j^2} - \|x_T - o_j\|\right]\ ,$$

which leads to upper bounding A as by the quantity

$$2J\left[\frac{\|x_T - o_j\|\left(\sqrt{\epsilon+\rho_j^2} - \|x_T - o_j\|\right)}{\|x_c - x_T\|^2}\left(1 + \frac{\sqrt{J}|\boldsymbol{v}^\intercal\alpha_j|^2}{2(k\bar\beta_j)^2}\right) - 1\right]$$

which can be shown to be negative. Indeed, given the workspace validity assumption, $\sqrt{\epsilon + \rho_j^2} - \|x_T - o_j\| < \sqrt{\epsilon + \rho_j^2} - (\rho_j + r_T + \sqrt{\epsilon} + \delta) < 0$. On the other hand, term B is multiplied by $\beta_j < \epsilon$ for in $x_c \in \mathcal{B}_j(\epsilon)$. Therefore, by making ϵ small enough, one can guarantee that $\boldsymbol{v}^\intercal\nabla^2\varphi|_{x_c}\boldsymbol{v} < 0$. Specifically how small, can be determined as follows, first by further upper bounding A for $x_c \in \mathcal{B}_j(\epsilon)$:

$$\begin{aligned}
A &\le -2J\left[\tfrac{r_T\|x_T - o_j\|}{\|x_c - x_T\|^2} + 1\right]\\
&\le -2J\left[\tfrac{r_T(r_T + \sqrt{\epsilon} + \delta)}{(2\rho_0 - r_T - \sqrt{\epsilon} - \delta)^2} + 1\right]\\
&\le -2(\delta + \sqrt{\epsilon})^4\left[\tfrac{r_T(r_T + \delta + \sqrt{\epsilon})}{(2\rho_0 - r_T - \delta - \sqrt{\epsilon})^2} + 1\right]\\
&\le -2\delta^4\left[\left(\tfrac{r_T + \delta}{2\rho_0 - r_T - \delta}\right)^2 + 1\right]\ .
\end{aligned}\tag{B20}$$

Similarly,

$$\begin{aligned}
B &\le \frac{|\nabla\bar\beta_j^\intercal\nabla J|\|\nabla^2 J\|}{16\|x_c - x_T\|^2} + J\|\tfrac{1-\frac{1}{k}}{\bar\beta_j}\nabla\bar\beta_j\nabla\bar\beta_j^\intercal - \nabla^2\bar\beta_j\|\\
&\le \frac{\|\nabla\bar\beta_j\|\|\nabla J\|\|\nabla^2 J\|}{16\|x_c - x_T\|^2} + \tfrac{k-1}{k\bar\beta_j}J\|\nabla\bar\beta_j\|^2 + J\|\nabla^2\bar\beta_j\|\\
&\le \tfrac{\|\nabla\bar\beta_j\|\|\nabla J\|\|\nabla^2 J\|}{16(\delta + r_T)} + \tfrac{(k-1)J\|\nabla\bar\beta_j\|^2}{k\bar\beta_j} + J\|\nabla^2\bar\beta_j\|\\
&\overset{(A4,A11,A13)}{\le} \tfrac{6m(\rho_0 - \delta)^{m-1}(2\rho_0 - r_T - \delta)^5}{\delta + r_T} + \tfrac{32m(\rho_0 - \delta)^5}{\delta^2}\\
&+2(2\rho_0 - r_T - \delta)^4(\rho_0 - \delta)^{m-1}\left[(\rho_0 - \delta)^{m-1} + (m-1)(\rho_0 - \delta)^{m-2}\right.\\
&\left.+m(m^2 - 2m + 2)(\rho_0 - \delta)^{m-3} + (m-1)^2\right]\ .
\end{aligned}\tag{B21}$$

Using (B20) and (B21), (B19) allows for the derivation of a lower bound on ϵ that would make $\boldsymbol{v}^\intercal\nabla^2\varphi|_{x_c}\boldsymbol{v}$ negative:

$$\begin{aligned}
&\tfrac{\beta^2\boldsymbol{v}^\intercal\nabla^2\varphi|_{x_c}\boldsymbol{v}}{J^{k-1}} \le \beta_j B + \bar\beta_j A \le 0\\
&\impliedby \beta_j \le -\tfrac{(\rho_0 - \delta)^{m-1}A}{B}\\
&\impliedby \epsilon \le 2\delta^4(\rho_0 - \delta)^{m-1}\left[\left(\tfrac{r_T + \delta}{2\rho_0 - r_T - \delta}\right)^2 + 1\right]\\
&\times\left\{\tfrac{6m(\rho_0 - \delta)^{m-1}(2\rho_0 - r_T - \delta)^5}{\delta + r_T} + \tfrac{32m(\rho_0 - \delta)^5}{\delta^2}\right.\\
&+2(2\rho_0 - r_T - \delta)^4(\rho_0 - \delta)^{m-1}\left[(\rho_0 - \delta)^{m-1} + (m-1)(\rho_0 - \delta)^{m-2}\right.\\
&\left.\left.+m(m^2 - 2m + 2)(\rho_0 - \delta)^{m-3} + (m-1)^2\right\}^{-1}\right.\ .
\end{aligned}$$

□

B.5 Proof of Proposition 5

Proof By contradiction: assume $x_c \in \mathcal{F}_1(\epsilon)$ is a critical point. At a critical point, we know that $\nabla\hat{\varphi}(x_c) \equiv \nabla\frac{J^k}{\beta} = 0$. We will show that an appropriate choice of k forces $\nabla\hat{\varphi}^\intercal\nabla J > 0$ instead.

Since $x_c \in \mathcal{F}_1(\epsilon) \subset \mathcal{B}_0(\epsilon)$, the workspace validity assumption forces a minimal separation between x_c and x_T:

$$\|x_c\| - \|x_T(t)\| > \delta \ . \tag{B22}$$

Given that,

$$\begin{aligned}
&\nabla J^\intercal\nabla\beta_0 \overset{(A10),(A1)}{=} \\
&-8\sqrt{J}\,(x_c - x_T(t))^\intercal\, x_c = 8\sqrt{J}\left(x_T(t)^\intercal x_c - \|x_c\|^2\right) \\
&\leq 8\underbrace{\sqrt{J}}_{>0}\underbrace{\|x_c\|}_{>0}\underbrace{(\|x_T(t)\| - \|x_c\|)}_{<0} \overset{(B22)}{<} 0 \ .
\end{aligned} \tag{B23}$$

Now expand $\nabla\hat{\varphi}^\intercal\nabla J$ and bound it as follows:

$$\begin{aligned}
\nabla\hat{\varphi}^\intercal\nabla J &\overset{(A6)}{=} \frac{k J^{k-1}}{\beta^2}\left(\beta\,\nabla J - \frac{J\nabla\beta}{k}\right)^\intercal \nabla J \\
&= \frac{J^k}{\beta^2}\left(\frac{k\beta}{J}\nabla J^\intercal\nabla J - \nabla\beta^\intercal\nabla J\right) \\
&= \frac{J^k}{\beta^2}\left(16k\beta\|x - x_T(t)\|^2 - \nabla\beta^\intercal\nabla J\right) \\
&= \frac{J^k\beta_0\left[16k\bar{\beta}_0\|x - x_T(t)\|^2 - \left(\nabla\bar{\beta}_0^\intercal + \frac{\bar{\beta}_0}{\beta_0}\nabla\beta_0^\intercal\right)\nabla J\right]}{\beta^2} \\
&\overset{(B23)}{>} \frac{J^k\beta_0}{\beta^2}\left(16k\bar{\beta}_0\|x - x_T(t)\|^2 - \nabla\bar{\beta}_0^\intercal\nabla J\right) \ .
\end{aligned}$$

To ensure $\nabla\hat{\varphi}^\intercal\nabla J > 0$, one needs $k > \frac{\nabla\bar{\beta}_0^\intercal\nabla J}{16\bar{\beta}_0\|x - x_T(t)\|^2}$ and toward this end we obtain a supremum of that expression in $\mathcal{F}_1(\epsilon)$ as follows:

$$\begin{aligned}
\frac{\nabla\bar{\beta}_0^\intercal\nabla J}{16\bar{\beta}_0\|x - x_T(t)\|^2} &\leq \frac{\|\nabla\bar{\beta}_0\|\|\nabla J\|}{16\bar{\beta}_0\|x - x_T(t)\|^2} \\
&= \frac{\sqrt{J}}{4\|x - x_T(t)\|}\frac{\|\nabla\bar{\beta}_0\|}{\bar{\beta}_0} \\
&\leq \sup_{\mathcal{F}_1}\left(\frac{\sqrt{J}}{4\|x - x_T(t)\|}\right)\sup_{\mathcal{F}_1}\left(\frac{\|\nabla\bar{\beta}_0\|}{\bar{\beta}_0}\right) \\
&\leq \sup_{\mathcal{F}_1}\left(\frac{\|x - x_T(t)\|^2 - r_T^2}{4\,\|x - x_T(t)\|}\right)\sum_{l=1}^{m}\left(\sup_{\mathcal{F}_1}\frac{2\,\|x - o_l(t)\|}{\beta_l}\right) \\
&< \sup_{\mathcal{F}_1}\left(\frac{\|x - x_T(t)\|}{2} - \frac{r_T^2}{2\,\|x - x_T(t)\|}\right)\sum_{l=1}^{m}\left(\sup_{\mathcal{F}_1}\frac{\|x - o_l(t)\|}{\delta^2}\right) \\
&\leq \frac{2m(\rho_0 - \delta)^2}{\delta^2} \triangleq k_1 \ ,
\end{aligned}$$

and $k_1 \leq k$, no critical points exist in $\mathcal{F}_1(\epsilon)$. □

B.6 Proof of Proposition 6

Proof One way to establish non-degeneracy for a critical point is to partition the tangent space on which $\hat{\varphi}$ lies into two subspaces, and ensure that a quadratic form $v^\intercal\nabla^2\hat{\varphi}v$ is positive for all vectors v in one subspace, and negative for all vectors v on the other (Koditschek and Rimon (1990),Lemma 3.8).

If β_j is the implicit function for the obstacle closest to critical point $x_c \in \mathcal{F}_0(\epsilon)$, the proof of Proposition 4 established that for v in the subspace that is orthogonal to $\nabla\beta_j/\|\nabla\beta_j\|$, $v^\intercal\nabla^2\hat{\varphi}\,v < 0$. So now consider the complement of the aforementioned subspace, which is naturally spanned by $\bar{v} \triangleq \nabla\beta_j/\|\nabla\beta_j\|$. We want to verify that $\bar{v}^\intercal\nabla^2\varphi\bar{v} > 0$.

Combining (A8) with (A9) yields

$$\nabla^2\hat{\varphi}|_{x_c} = \frac{J^{k-1}}{\beta^2}\left(k\beta\nabla^2 J + \frac{J(1-1/k)}{\beta}\nabla\beta\nabla\beta^\intercal - J\nabla^2\beta\right) \ .$$

Let us now expand the expression

$$\begin{aligned}
&\frac{\beta^2}{J^{k-1}}\bar{v}^\intercal\ \nabla^2\hat{\varphi}\ \bar{v} = \\
&\bar{v}^\intercal\left[\frac{J}{\beta}\left(1 - \frac{1}{k}\right)\nabla\beta\nabla\beta^\intercal - J\,\nabla^2\beta + k\beta\nabla^2 J\right]\bar{v} = \\
&k\beta\ \bar{v}^\intercal\nabla^2 J\ \bar{v} + \frac{J\left(1-\frac{1}{k}\right)}{\beta}\left(\nabla\beta^\intercal\ \bar{v}\right)^2 - J\ \bar{v}^\intercal\nabla^2\beta\ \bar{v} \ ,
\end{aligned} \tag{B24}$$

and note that for small enough ϵ (Koditschek and Rimon (1990),p. 435)

$$\frac{J\|\nabla\beta\|^2}{2k\beta} + \frac{J(1-1/k)}{\beta}\left(\nabla\beta^\intercal\ \bar{v}\right)^2 - J\bar{v}^\intercal\nabla^2\beta\ \bar{v} \geq 0 \ .$$

Then to set the sign of (B24), it suffices to make

$$\bar{v}^\intercal\ k\beta\nabla^2 J\ \bar{v} \geq \frac{J\|\nabla\beta\|^2}{2k\beta} \ . \tag{B25}$$

To ensure that, first focus on the left hand side of (B25) and recall (A12) for the Hessian of J at a critical point x_c:

$$\begin{aligned}
&\bar{v}^\intercal k\beta\nabla^2 J\bar{v} \\
&\overset{(A10)}{=} k\beta\bar{v}^\intercal\left(4\sqrt{J}I + 8\,(x_c - x_T(t))\,(x_c - x_T(t))^\intercal\right)\bar{v} \\
&= 4k\beta\sqrt{J} + 8k\beta|\bar{v}^\intercal\,(x_c - x_T(t))\,|^2 \ .
\end{aligned} \tag{B26}$$

At the critical point x_c we have $J\nabla\beta = k\beta\nabla J$ and by taking the squared norms of both sides we arrive at

$$4k\beta \overset{(A10)}{=} \frac{J\|\nabla\beta\|^2}{4k\beta\|x_c - x_T(t)\|^2} \ . \tag{B27}$$

Now we plug (B27) back to (B26) and obtain the left hand side of (B25) in the form:

$$k\beta\,\bar{v}^{\intercal}\nabla^2 J\,\bar{v} = 4k\beta\sqrt{J} + 8k\beta|\bar{v}^{\intercal}\,(x_c - x_T(t))\,|^2 = \frac{\|\nabla\beta\|^2 J^{3/2} + 2J\|\nabla\beta\|^2|\bar{v}^{\intercal}\,(x_c - x_T(t))\,|^2}{4k\beta\|x_c - x_T(t)\|^2}\ . \quad \text{(B28)}$$

Given (B28), (B25) reduces to

$$\frac{\sqrt{J} + 2|\bar{v}^{\intercal}\,(x_c - x_T(t))\,|^2}{2\|x_c - x_T(t)\|^2} \geq 1 \iff \|x_c - x_T(t)\|^2 - r_T^2 + 2|\bar{v}^{\intercal}\,(x_c - x_T(t))\,|^2 \geq 2\|x_c - x_T(t)\|^2 \iff 2|\bar{v}^{\intercal}\,(x_c - x_T(t))\,|^2 \geq \|x_c - x_T(t)\|^2 + r_T^2\ . \quad \text{(B29)}$$

Assuming, without loss of generality, that for some j, $x_c \in \mathcal{B}_j(\epsilon)$, means that $\bar{v}^{\intercal} = \big(x_c - o_j(t)\big)^{\intercal}/\|x_c - o_j(t)\|$ in which case (B29) becomes

$$2|\tfrac{(x_c - o_j(t))^{\intercal}}{\|x_c - o_j(t)\|}\,(x_c - x_T(t))|^2 \geq \|x_c - x_T(t)\|^2 + r_T^2\ . \quad \text{(B30)}$$

Any critical point x_c inside $\mathcal{F}_0(\epsilon)$ will be by definition away from the target, i.e., $\|x_c - x_T(t)\| > r_T$. There is therefore a $\zeta < 1$ such that

$$r_T = \zeta \inf_{\mathcal{B}_j(\epsilon)} \|x_c - x_T(t)\|\ .$$

Now (B30) is implied if

$$\frac{\big(x_c - o_j(t)\big)^{\intercal}\,(x_c - x_T(t))}{\|x_c - o_j(t)\|\|x_c - x_T(t)\|} > 1 \geq \sqrt{\frac{1+\zeta^2}{2}}\ . \quad \text{(B31)}$$

Let us see, therefore, how the left hand side of (B31) can be lower-bounded.

First, let us leverage (B17) to substitute for $x_c - x_T(t)$ in the left hand side of (B31) and lower bound it (dropping indicators of time dependence for brevity) as

$$\begin{aligned}
&\frac{\big(x_c - o_j\big)^{\intercal}\,(x_c - x_T)}{\|x_c - o_j\|\|x_c - x_T\|} \\
&\geq \frac{\frac{\sqrt{J}(x_c - o_j)^{\intercal}}{4k}\left[\frac{2(x_c - o_j)}{\beta_j} + \frac{\alpha_j}{\bar{\beta}_j}\right]}{\frac{\sqrt{J}\,\|x_c - o_j\|}{4k}\left[\frac{2\|x_c - o_j\|}{\beta_j} + \frac{\|\alpha_j\|}{\bar{\beta}_j}\right]} \\
&\geq \frac{2\|x_c - o_j\|^2/\beta_j - \|x_c - o_j\|\|\alpha_j\|/\bar{\beta}_j}{2\|x_c - o_j\|^2/\beta_j + \|x_c - o_j\|\|\alpha_j\|/\bar{\beta}_j} \\
&= \frac{1 - \beta_j\|\alpha_j\|/2\bar{\beta}_j\|x_c - o_j\|}{1 + \beta_j\|\alpha_j\|/2\bar{\beta}_j\|x_c - o_j\|} \\
&= 1 - \frac{\beta_j\|\alpha_j\|/\bar{\beta}_j\|x_c - o_j\|}{1 + \beta_j\|\alpha_j\|/2\bar{\beta}_j\|x_c - o_j\|} \\
&\geq 1 - \frac{\beta_j\|\alpha_j\|}{\bar{\beta}_j\|x_c - o_j\|} \geq 1 - \frac{\epsilon\|\alpha_j\|}{\bar{\beta}_j\|x_c - o_j\|}\ .
\end{aligned} \quad \text{(B32)}$$

Using the lower bound of (B32) in place of the left hand side of (B31) yields a suitable ϵ that essentially enforces (B25) through (B29):

$$1 - \frac{\epsilon\|\alpha_j\|}{\bar{\beta}_j\|x_c - o_j\|} > \sqrt{\frac{1+\zeta^2}{2}} \iff \frac{\epsilon\|\alpha_j\|}{\bar{\beta}_j\|x_c - o_j\|} < 1 - \sqrt{\frac{1+\zeta^2}{2}} \iff \epsilon < \left(1 - \sqrt{\frac{1+\zeta^2}{2}}\right)\frac{\bar{\beta}_j\|x_c - o_j\|}{\|\alpha_j\|} < \left(1 - \sqrt{\frac{1+\zeta^2}{2}}\right)\frac{\rho_j^m}{\|\alpha_j\|}\ .$$

To fix the bound on ϵ we produce an upper bound for α_j as follows (dropping temporarily again time dependency indicators for brevity):

$$\begin{aligned}
\|\alpha_j\| &= \Big\|2\sum_{l=1,l\neq j}^{m}(x_c - o_l)\tfrac{\bar{\beta}_l}{\bar{\beta}_j} - 2\tfrac{\bar{\beta}_0}{\bar{\beta}_j}x_c\Big\| \\
&= \Big\|2\sum_{l=1,l\neq j}^{m}(x_c - o_l)\prod_{\substack{a=1\\ l\neq a\neq j}}^{m}\beta_a - 2x_c\prod_{\substack{b=1\\ 0\neq b\neq j}}^{m}\beta_b\Big\| \\
&< 2\sum_{\substack{l=1\\ l\neq j}}^{m}\sup_{x_c\in\mathcal{F}_0}\|(x_c - o_l)\|\prod_{\substack{a=1\\ j\neq a\neq l}}^{m}\beta_a - 2\|x_c\|\prod_{\substack{b=1\\ 0\neq b\neq j}}^{m}\beta_b \xrightarrow{\text{ess inf }\beta_b \to 0} \\
&\leq 2^{2m-3}(\rho_0 - \sqrt{\epsilon} - \delta)^{2m-4}\sum_{\substack{l=1\\ l\neq j}}^{m}(\rho_l + \sqrt{\epsilon}) \\
&\leq (m-1)(2\rho_0)^{2m-3}\ .
\end{aligned}$$

Given this bound on $\|\alpha_j\|$, an upper bound ϵ_3' on ϵ is set:

$$\epsilon_3{}' \triangleq \frac{1 - \sqrt{\frac{1+\zeta^2}{2}}}{2^{2m-3}(m-1)\rho_0^{m-3}}\ .$$

Tracing the thought trail back,

$$\epsilon < \epsilon_3{}' \implies 1 - \frac{\epsilon\|\alpha_j\|}{\bar{\beta}_j\|x_c - o_j\|} > \sqrt{\frac{1+\zeta^2}{2}} \implies \bar{v}^{\intercal}\,k\beta\nabla^2 J\,\bar{v} \geq \frac{J\|\nabla\beta\|^2}{2k\beta} \implies \bar{v}^{\intercal}\nabla^2\varphi\,\bar{v} > 0\ ,$$

and the critical point $x_c \in \mathcal{F}_0(\epsilon)$ cannot be degenerate. □

References

Aiushita, S., Hisanobu, T., & Kawamura, S. (1993) Fast path planning available for moving obstacle avoidance by use of laplace potential.

In *Proceedings of the IEEE international conference on robotics and automation* (pp. 673–678).

Ajanovic, Z., Lacevic, B., Shyrokau, B., Stolz, M., Horn, M. (2018). Search-based optimal motion planning for automated driving. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems* (pp. 4523–4530).

Alonso-Mora, J., DeCastro, J. A., Raman, V., Rus, D., & Kress-Gazit, H. (2017). Reactive mission and motion planning with deadlock resolution avoiding dynamic obstacles. *Autonomous Robots, 42*(4), 801–824.

Alsaab, A., & Bicker, R. (2014) Improving velocity obstacle approach for obstacle avoidance in indoor environments. In *Proceedings of the UKACC international conference on control* (pp. 325–330).

Arslan, O., & Koditschek, D. E. (2019). Sensor-based reactive navigation in unknown convex sphere worlds. *The International Journal of Robotics Research, 38*(2–3), 196–223.

Cai, K., Wang, C., Cheng, J., De Silva, C. W., Meng, M. Q.-H. (2018) Mobile robot path planning in dynamic environments: A survey. arXiv preprint arXiv:2006.14195

Chen, C., Li, C., & Tanner, H. G. (2020) Navigation functions with non-point destinations and moving obstacles. In *Proceedings of the IEEE American control conference* (pp. 2532–2537).

Chen, P. C., & Hwang, Y. K. (1998). SANDROS: A dynamic graph search algorithm for motion planning. *IEEE Transactions on Robotics and Automation, 14*(3), 390–403.

Connolly, C. (1997). Harmonic functions and collision probabilities. *The International Journal of Robotics Research, 16*(4), 497–507.

Elbanhawi, M., & Simic, M. (2014). Sampling-based robot motion planning: A review. *IEEE Access, 2*, 56–77.

Faust, A., Oslund, K., Ramirez, O., Francis, A., Tapia, L., Fiser, M., Davidson, J. (2018). PRM-RL: Long-range robotic navigation tasks by combining reinforcement learning and sampling-based planning. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 5113–5120)

Ferguson, D., & Stentz, A. (2006). Using interpolation to improve path planning: The field D* algorithm. *Journal of Field Robotics, 23*(2), 79–101.

Fernandes, L. C., Souza, J. R., Shinzato, P. Y., Pessin, G., Mendes, C. C. T., Osorio, F. S., Wolf, D. F. (2012) Intelligent robotic car for autonomous navigation: Platform and system architecture. In *Proceedings of the second Brazilian conference on critical embedded systems* (pp. 12–17).

Francis, A., Faust, A., Chiang, H.-T.L., Hsu, J., Kew, J. C., Fiser, M., & Lee, T.-W.E. (2020). Long-range indoor navigation with PRM-RL. *IEEE Transactions on Robotics, 36*(4), 1115–1134.

Fulgenzi, C., Spalanzani, A., & Laugier, C. (2009) Probabilistic rapidly-exploring random trees for autonomous navigation among moving obstacles. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 4027–4033).

González, D., Pérez, J., Milanés, V., & Nashashibi, F. (2015). A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems, 17*(4), 1135–1145.

Hasselt, H. v., Guez, A., Silver, D. (2016). Deep reinforcement learning with double Q-learning. In *Proceedings of the thirtieth AAAI conference on artificial intelligence* (pp. 2094–2100).

Iizuka, S., Nakamura, T., & Suzuki, S. (2014). Robot navigation in dynamic environment using navigation function APF with SLAM. In *Proceedings of the IEEE 8th Europe-Asia congress on mecatronics* (pp. 89–92).

Jaillet, L., Siméon, T. (2004) A PRM-based motion planner for dynamically changing environments. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems* (Vol. 2, pp. 1606–1611).

Karaman, S., & Frazzoli, E. (2011). Sampling-based algorithms for optimal motion planning. *The International Journal of Robotics Research, 30*(7), 846–894.

Khalil, H. K. (2002). *Nonlinear systems*. Prentice Hall.

Koditschek, D. E., & Rimon, E. (1990). Robot navigation functions on manifolds with boundary. *Advances in Applied Mathematics, 11*(4), 412–442.

Li, C., & Tanner, H. G. (2019). Navigation functions with time-varying destination manifolds in star worlds. *IEEE Transactions on Robotics, 35*(1), 35–48.

Loizou, S. G., Tanner, H. G., Kumar, V., Kyriakopoulos, K. (2003) Closed loop navigation for mobile robots in dynamic environments. In *Proceedings of the IEEE/RSJ international conference on robots and systems* (pp. 3769–3774).

Minguez, J., Lamiraux, F., & Laumond, J.-P. (2008). Motion planning and obstacle avoidance. In B. Siciliano & O. Khatib (Eds.), *Springer Handbook of Robotics* (pp. 827–852). Berlin, Heidelberg: Springer.

Mohanan, M. G., & Salgoankar, A. (2018). A survey of robotic motion planning in dynamic environments. *Robotics and Autonomous Systems, 100*, 171–185.

Montiel, O., Orozco-Rosas, U., & Sepúlveda, R. (2015). Path planning for mobile robots using bacterial potential field for avoiding static and dynamic obstacles. *Expert Systems with Applications, 42*(12), 5177–5191.

Olunloyo, V. O. S., & Ayomoh, M. K. O. (2009). Autonomous mobile robot navigation using hybrid virtual force field concept. *European Journal of Scientific Research, 31*(2), 204–228.

Pandey, A., Pandey, S., & Parhi, D. R. (2017). Mobile robot navigation and obstacle avoidance techniques: A review. *International Robotics & Automation Journal, 2*(3), 96–105.

Paternain, S., Koditschek, D. E., & Ribeiro, A. (2018). Navigation functions for convex potentials in a space with convex obstacles. *IEEE Transactions on Automatic Control, 63*(9), 2944–2959.

Pradhan, N., Burg, T., & Birchfield, S. (2011). Robot crowd navigation using predictive position fields in the potential function framework. In *Proceedings of American control conference* (pp. 4628–4633).

Prassler, E., Scholz, J., & Fiorini, P. (2001). A robotics wheelchair for crowded public environment. *IEEE Robotics & Automation Magazine, 8*(1), 38–45.

Qi, J., Yang, H., & Sun, H. (2021). MOD-RRT*: A sampling-based algorithm for robot path planning in dynamic environment. *IEEE Transactions on Robotics, 68*(8), 7244–7251.

Rimon, E., & Koditschek, D. E. (1992). Exact robot navigation using artificial potential functions. *Transactions on Robotics and Automation, 8*(5), 501–518.

Short, A., Pan, Z., Larkin, N., & van Duin, S. (2016) Recent progress on sampling based dynamic motion planning algorithms. In *Proceedings of the IEEE international conference on advanced intelligent mechatronics* (pp. 1305–1311).

Shvalb, N., & Hacohen, S. (2019). Motion in potential field and navigation function. In E. Kagan, N. Shvalb, & I. Ben-Gal (Eds.), *Autonomous mobile robots and multi-robot systems* (pp. 87–107). John Wiley & Sons.

Sun, J., & Tanner, H. G. (2015). Constrained decision-making for low-count radiation detection by mobile sensors. *Autonomous Robots, 39*(4), 519–536.

Szulczyński, P., Pazderski, D., & Kozłowski, K. (2011). Harmonic functions and collision probabilities. *Journal of Automation Mobile Robotics and Intelligent Systems, 5*(3), 497–507.

Tanner, H. G., & Kyriakopoulos, K. J. (2000). Nonholonomic motion planning for mobile manipulators. In *Proceedings of the IEEE international conference on robotics and automation* (Vol. 2, pp. 1233–1238).

Tanner, H. G., Jadbabaie, A., & Pappas, G. J. (2003). Stable flocking of mobile agents, Part I: Fixed topology. In *Proceedings of the IEEE conference on decision and control* (pp. 2010–2015).

Tanner, H. G., Jadbabaie, A., & Pappas, G. J. (2005). Flocking in teams of nonholonomic agents. In S. Morse, N. Leonard, & V. Kumar (Eds.), *Cooperative control. Lecture notes in control and information sciences* (pp. 229–239). Springer.

Tiseni, L., Chiaradia, D., Gabardi, M., Solazzi, M., Leonardis, D., & Frisoli, A. (2021). UV-C mobile robots with optimized path planning: Algorithm design and on-field measurements to improve surface disinfection against SARS-CoV-2. *IEEE Robotics Automation Magazine, 28*(1), 59–70.

Van Den Berg, J.P., Nieuwenhuisen, D., Jaillet, L., & Overmars, M. H. (2005). Creating robust roadmaps for motion planning in changing environments. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems* (pp. 1053–1059).

Vasilopoulos, V., Pavlakos, G., Bowman, S. L., Caporale, J. D., Danilidis, K., Pappas, G. J., & Koditschek, D. E. (2020). Reactive semantic planning in unexplored semantic environments using deep perceptual feedback. *IEEE Robotics and Automation Letters, 5*(3), 4455–4462.

Wang, J., Meng, M.Q.-H., & Khatib, O. (2020). EB-RRT: Optimal motion planning for mobile robots. *IEEE Transactions on Automation Science and Engineering, 17*(4), 2063–2073.

Warnke, J., Shamsah, A., Li, Y., Zhao, Y. (2020). Towards safe locomotion navigation in partially observable environments with uneven terrain. In *Proceedings of IEEE conference on decision and control* (pp. 958–965).

Waydo, S., & Murray, R. M. (2003) Vehicle motion planning using stream functions. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 2484–2491).

Wijmans, E., Kadian, A., Morcos, A., Lee, S., Essa, I., Parikh, D., Savva, M., Batra, D. (2019). DD-PPO: Learning near-perfect pointgoal navigators from 2.5 billion frames. arXiv preprint arxiv:1911.00357

Xu, D., Fang, Y., Zhang, Z., Meng, Y. (2017). Path planning method combining depth learning and Sarsa algorithm. In *Proceedings of the international symposium on computational intelligence and design* (Vol. 2, pp. 77–82).

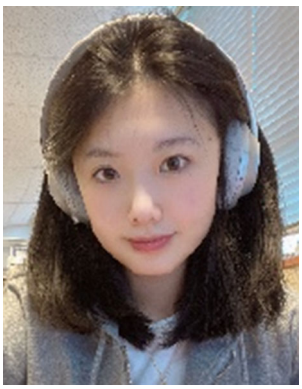
Yadav, I., Eckenhoff, K., Huang, G., Tanner, H.G. (2018). Visual-inertial target tracking and motion planning for UAV-based radiation detection. arXiv preprint arXiv:1805.09061

Yadav, I., Tanner, H. G. (2021) Exact decentralized receding horizon planning for multiple aerial vehicles. In *Proceedings of the IEEE conference on decision and control* (pp. 5747–5752).

Yao, P., Wang, H., & Su, Z. (2015). Real-time path planning of unmanned aerial vehicle for target tracking and obstacle avoidance in complex dynamic environment. *Aerospace Science and Technology, 47*, 269–279.

Zucker, M., Kuffner, J.,& Branicky, M. (2007). Multipartite RRTs for rapid replanning in dynamic environments. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 1603–1609).

Cong Wei received the Ph.D degree in Mechanical Engineering from the University of Delaware in 2021. He got the M.S. degree in ocean engineering from Harbin Engineering University, Harbin, China, in 2014 and the B.S. degree in marine engineering from Ningbo University, Ningbo, China, in 2012. From 2014 to 2016, he was a research assistant at Institute of Underwater Engineering, Shanghai Jiao Tong University, Shanghai, China. Cong Wei is currently a postdoctoral associate from Maryland Robotics Center (MRC) of the Institute for Systems Research (ISR) at the University of Maryland, College Park. His research interest includes multi-agent coordination, nonlinear control, marine robotics, and robotic motion planning.



Chuchu Chen is a Ph.D candidate at the University of Delaware pursuing degree in Mechanical Engineering in the College of Engineering. She is currently working in the Robot Perception and Navigation Group (RPNG) which is a part of the UD Center for Autonomous and Robotic Systems (CARS). Her research focuses on nonlinear state estimation, robot navigation, visual-inertial navigation systems (VINS) and simultaneous localization and mapping (SLAM). She received a Bachelor of Engineering Degree in Mechanical Engineering from Harbin Engineering University, Harbin, China, in 2017.

Bert Tanner received his Ph.D. in mechanical engineering from the NTUA, Athens, Greece, in 2001. He was a postdoctoral researcher at the University of Pennsylvania from 2001 to 2003, and subsequently took a position as an assistant professor at the University of New Mexico. In 2008 he joined the Department of Mechanical Engineering at the University of Delaware, where he is currently a professor. In 2019 he was appointed Director of the Center for Autonomous and Robotics Systems. Tanner's research interests are in the areas of multi-robot system planning and control, with emphasis on flocking and swarming, constrained navigation, heterogeneous coordination, and hybrid systems. He received NSF's Career award in 2005, he is a fellow of the ASME, and a senior member of IEEE. He has served in the editorial boards of the IEEE Transactions on Automatic Control, Nonlinear Analysis - Hybrid Systems, the IEEE Robotics and Automation Magazine and the IEEE Transactions on Automation Science and Engineering. He is currently an associate editor for Automatica, and he is a chief specialty editor for Frontiers in Robotics and AI: multi-robot systems. He has also been serving in several conference editorial boards of both IEEE Control Systems and IEEE Robotics and Automation Societies.