Harmonic Control Lyapunov Barrier Functions for Constrained Optimal Control with Reach-Avoid Specifications

Amartya Mukherjee[1], Ruikun Zhou[1], and Jun Liu[1]

Abstract— This paper introduces harmonic control Lyapunov barrier functions (harmonic CLBF) that aid in constrained control problems such as reach-avoid problems. Harmonic CLBFs exploit the maximum principle that harmonic functions satisfy to encode the properties of control Lyapunov barrier functions (CLBFs). As a result, they can be initiated at the start of an experiment rather than trained based on sample trajectories. The control inputs are selected to maximize the inner product of the system dynamics with the steepest descent direction of the harmonic CLBF. Numerical results are presented with four different systems under different reach-avoid environments. Harmonic CLBFs show a significantly low risk of entering unsafe regions and a high probability of entering the goal region.

I. INTRODUCTION

For safety-critical systems, ensuring stability and safety is of paramount importance. One approach to address this issue for such systems is to model it as a constrained optimal control problem. The objective of constrained optimal control is to design a control system that optimizes a notion of performance while satisfying a set of constraints. These constraints are usually designed to prevent catastrophic events, ensure safety, and take into multiple objectives with bounds. Furthermore, with experiments conducted in real life, resetting the environment if the agent goes outside a domain can be expensive [11]. One of the most common scenarios for safety-critical systems is the reach-avoid problem, which can be regarded as a subset of constrained optimal control problems where a trajectory aims to reach a goal state while avoiding a set of unsafe states.

In recent years, there has been a growing interest in reach-avoid problems among the control theory and reinforcement learning community. Such methods involve learning control Lyapunov functions to certify reachability and control barrier functions that satisfy avoidability [9], [3]. In reinforcement learning, agents learn optimal policies through exploration of the environment with unknown dynamics [19]. Constraints are imposed using penalty methods [10] and Lagrangian methods [1]. In these methods, two value functions are computed to estimate cumulative rewards and cumulative costs.

The problem with using two certificates or two value functions is that it leads to conflicts in the control policy whether to satisfy reachability or avoidability at a given point in time. To unify the conflicts, reachability and safety

[1]Amartya Mukherjee, Ruikun Zhou, and Jun Liu are with the Department of Applied Mathematics, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1 (email: `a29mukhe`, `ruikun.zhou`, `j.liu@uwaterloo.ca`).

have been combined into a single certificate called the control Lyapunov barrier function (CLBF). It is shown in [16] that theoretically there exists a single Lyapunov barrier function for such reach-avoid type specifications with stability and safety guarantees, provided that such specifications are robustly satisfiable. Moreover, [5] introduces robust CLBFs and derives a QP-based controller, in which the CLBF is computed using neural networks by encoding the constraints in their loss function. In a similar manner but with reinforcement learning, the Lyapunov barrier-based actor-critic method is proposed in [6] to find a controller that satisfies both certificates with a CLBF that is obtained by solving the Lagrangian relaxation of the constrained optimization problem. Unfortunately, both methods do not provide guarantees that their value function V satisfies all the CLBF constraints, and this is evidenced by their experiments. For example, in 2D quadrotor control problems, the contour plots of the derived CLBF do not completely cover the unsafe regions. Our work addresses this issue by enforcing the CLBF constraints as boundary conditions for harmonic functions. Numerical results demonstrate that this technique indeed improves safety rates, compared with the results reported in [5].

Harmonic functions are solutions to the Laplace equation, which is a second-order linear elliptic partial differential equation (PDE). They satisfy the maximum principle on a compact set, thus making it easier to impose CLBF constraints. They do not have any critical points other than saddle points in the interior of their domain, which makes it easy to derive optimal control strategies. Harmonic functions have been used in the control literature to derive potential fields in past works. In the work of [12], harmonic potential functions with panel methods were used to build potential fields over a task space. The work of [18] used harmonic functions with complex variables and conformal mappings to achieve moving obstacle avoidance. Furthermore, [4] combines harmonic functions using superposition to derive potential fields.

In this paper, we will focus on reach-avoid problems, where an agent must reach a goal region while avoiding unsafe regions. For example, the unsafe regions could be regions on a floor that have holes or pillars. We achieve this by introducing harmonic control Lyapunov barrier functions (harmonic CLBFs) that use Laplace's equation to encode the properties of control Lyapunov barrier functions (CLBFs). Furthermore, Laplace's equation can be solved using numerical methods such as finite element methods (FEMs) and do not be trained using neural networks based on sample

trajectories as done in prior work, thus significantly reducing the computational cost of reach-avoid experiments.

In all, the main contributions of this paper are threefold. Firstly, we propose a novel method to unify harmonic functions with CLBFs to provide both stability and safety guarantees. Secondly, under this framework, we show that optimal controllers for reach-avoid tasks can be derived using gradient-based methods directly with no need for searching such controllers by training a neural network or solving optimal problems. Lastly, we conduct a set of numerical experiments that show a significantly low risk of trajectories entering unsafe regions with the proposed approach.

II. PRELIMINARIES

Throughout this work, we consider a nonlinear control system of the form

$$\dot{x} = f(x, u), \quad x(0) = x_0, \tag{1}$$

where $x \in \mathcal{S} \subset \mathbb{R}^n$ is the state of the system, and $u \in U \subset \mathbb{R}^m$ is the control input.

Let $\mathcal{S}$ be a compact subspace of $\mathbb{R}^n$ denoting the space of all admissible states in a control problem. Let $\mathcal{S}_{goal} \subset \mathcal{S}$ be an open set denoting the space of states that the controller aims to reach, and let $\mathcal{S}_{unsafe}$ be a compact subset of $\mathcal{S}$ denoting the space of states the controller aims to avoid. Define $\mathcal{S}_{safe} = \mathcal{S} \backslash \{\overline{\mathcal{S}_{goal}} \cup \mathcal{S}_{unsafe}\}$.

A. Control Lyapunov Barrier Functions

Definition 1 (Control Lyapunov Barrier Function [5], [6]): A function $V : \mathcal{S} \to \mathbb{R}$ is a control Lyapunov barrier function (CLBF) if, for some constants $c, \lambda > 0$:
1) $V(s) = 0$ for all $s \in \mathcal{S}_{goal}$;
2) $V(s) > 0$ for all $s \in \mathcal{S} \backslash \mathcal{S}_{goal}$;
3) $V(s) \geq c$ for all $s \in \mathcal{S}_{unsafe}$;
4) $V(s) < c$ for all $s \in \mathcal{S} \backslash \mathcal{S}_{unsafe}$;
5) There exists a controller $\pi : \mathcal{S}_{safe} \to U$ such that $\langle \nabla V(s), f(s, \pi(s)) \rangle + \lambda V(s) \leq 0$ for all $s \in \mathcal{S}_{safe}$.

B. Harmonic Functions

Definition 2 (Harmonic Function [8]): A harmonic function u is a C^2-function that satisfies the Laplace equation $\nabla^2 u = \nabla \cdot \nabla u = 0$

In the following theorems, we introduce properties of harmonic functions that are of importance in the field of elliptic PDEs, differential geometry, and complex analysis.

Theorem 1 (Mean Value Theorem [17]): If u is harmonic on a domain $\mathcal{S}$, then u satisfies the mean value property in $\mathcal{S}$. Furthermore, if $x \in \mathcal{S}$ and $r > 0$ are such that $\overline{B_r(x)} \subset \mathcal{S}$, where $B_r(x)$ is a sphere of radius r centered at x, then

$$u(x) = \frac{\int_{B_r(x)} u(y) dy}{\int_{B_r(x)} dy}. \tag{2}$$

This theorem is essential for proving Theorems 2 and 3 that provide bounds on harmonic functions.

Theorem 2 (Strong Maximum Principle) [8]: If u is harmonic on a domain $\mathcal{S}$ and u has its maximum in $\mathcal{S} \backslash \partial \mathcal{S}$, then u is constant.

Theorem 3 (Weak Maximum Principle [8]): If $u \in C^2(\mathcal{S}) \cup C^1(\partial \mathcal{S})$ is harmonic in a bounded domain $\mathcal{S}$, then

$$\max_{\overline{\mathcal{S}}} u = \max_{\partial \mathcal{S}} u. \tag{3}$$

This shows that Theorems 2 and 3 can be exploited to derive a CLBF that has its maximum in unsafe regions and minimum in goal regions.

III. HARMONIC CLBF

In this work, we explore the intersection between CLBF and harmonic functions. By exploiting the maximum principle, we can derive a function $V : S \to \mathbb{R}$ that satisfies properties (1)–(4) for CLBF by imposing boundary conditions on the CLBF.

Definition 3 (Harmonic CLBF): A function $V \in C^2(\mathcal{S}) \cup C^1(\partial \mathcal{S})$ is a harmonic CLBF if it satisfies the following conditions:
1) $\nabla^2 V(s) = 0$ for all $s \in \mathcal{S}_{safe}$
2) $V(s) = 0$ for all $s \in \overline{\mathcal{S}_{goal}}$
3) $V(s) = c$ for all $s \in \partial \mathcal{S} \cup \mathcal{S}_{unsafe}$.

In other words, $V(s)$ is a solution to the boundary value problem given above.

Theorem 4: All harmonic CLBFs V satisfy properties (1)–(4) in Definition 1.

Proof: Properties (1) and (3) in Definition 1 are trivially satisfied as they are set as boundary conditions from properties (2) and (3) in Definition 3.

If $V(s) \leq 0$ for some $s \in \mathcal{S} \backslash \mathcal{S}_{goal}$ ($V(s) \geq c$ for some $s \in \mathcal{S} \backslash \mathcal{S}_{unsafe}$), then $V(s)$ has its minimum (maximum) in the interior of $\mathcal{S}_{safe}$. By Theorem 2, V is then constant in $\mathcal{S}_{safe}$, which contradicts the non-constant boundary conditions imposed on it. Thus, harmonic CLBFs satisfy properties (2) and (4) in Definition 1. ∎

Proposition 1: Every reach-avoid problem admits a unique harmonic CLBF.

Proof: Consider an arbitrary reach-avoid problem characterized by S, S_{goal}, S_{unsafe}. Assume for a contradiction that a reach avoid problem has two distinct harmonic CLBFs, $V_1(s)$ and $V_2(s)$.

Define the function $U(s) = V_1(s) - V_2(s)$. The Laplacian of $U(s)$ is $\nabla^2 U = \nabla^2 V_1 - \nabla^2 V_2 = 0$ on S_{safe}. As a result, U is a function satisfying $\nabla^2 U = 0$ on S_{safe}, $U = 0$ on $\overline{S_{goal}}$, and $U = 0$ on $\partial \mathcal{S} \cup \mathcal{S}_{unsafe}$. By Theorem 3, U has its maximum and minimum of 0 on the boundary, so $U = 0$ on S_{safe}. Thus, the harmonic CLBF is unique. ∎

We now introduce a sufficient condition for verifying property (5) of the CLBF conditions in Definition 1.

Theorem 5: Let V be a harmonic CLBF, and consider the deterministic nonlinear system (1). If there exists a positive constant λ such that

$$\lambda \leq - \sup_{x \in \mathcal{S}_{safe}} \inf_{u \in U} \frac{\langle f(x, u), \nabla V(x) \rangle}{V(x)}, \tag{4}$$

then property (5) in Definition 1 holds.

Proof: Define an optimal controller $\pi^*(x)$ as

$$\pi^*(x) = \arg \inf_{u \in U} \langle f(x, u), \nabla V(x) \rangle.$$

Since $\lambda V(x) \geq 0$ for all $x \in \mathcal{S}_{safe}$, it follows that

$$
\begin{aligned}
\frac{\langle f(x, \pi^*(x)), \nabla V(x) \rangle}{V(x)} &\leq \inf_{u \in U} \frac{\langle f(x, u), \nabla V(x) \rangle}{V(x)} \\
&\leq \sup_{x \in \mathcal{S}_{safe}} \inf_{u \in U} \frac{\langle f(x, u), \nabla V(x) \rangle}{V(x)} \\
&\leq -\lambda,
\end{aligned}
$$

where the last inequality is precisely (4). We have verified property (5) in Definition 1. ∎

Furthermore, a sufficient condition for a system to avoid $\mathcal{S}_{unsafe}$ is:

$$
\sup_{x \in \mathcal{S}_{safe}} \inf_{u \in U} \langle f(x, u), \nabla V(x) \rangle \leq 0 \tag{5}
$$

Since $\mathcal{S}_{unsafe}$ is a closed set, this means $V(x) < c$ for all $x \in \mathcal{S}_{safe}$. Furthermore, since $\frac{\partial}{\partial t} V(x(t)) \leq 0$, this means for all $t \in [0, \infty)$, $x(t)$ will never approach a point where $V(x) = c$, meaning, it will never approach a point in $\mathcal{S}_{unsafe}$.

A. Avoiding the saddle points

As shown in [20], all critical points in $V(x)$ are saddle points. Random noise is added in [7] to parameters while performing gradient descent to guarantee that the algorithm converges to a local minimum. The authors of [13] show that gradient descent with random parameter initialization asymptotically avoids saddle points. So in this paper, we compare deterministic control inputs with stochastic control inputs to see which method shows better convergence to $\mathcal{S}_{goal}$.

At every point $x \in \mathcal{S}_{safe}$, the control input is selected as $u = \operatorname{argmin}_{u \in U} \langle f(x, u), \nabla V(x) \rangle + z$ where z is a small noise sampled from a normal distribution $N(0, \sigma^2)$ to mitigate local optima.

B. Conditions on the noise

In order to preserve the properties of the CLBF, we introduce bounds that must be imposed on z with Taylor series expansion so that the trajectory avoids unsafe regions when the first-order terms dominate.

Let Δt be the time step size used for numerical simulation. Consider the first-order expansion of $x(t + \Delta t)$ as

$$
x(t + \Delta t) = x + f(x, u) \Delta t + O(\Delta t^2). \tag{6}
$$

By adding some noise z to the control input to avoid saddle points, we have

$$
x(t + \Delta t) = x + f(x, u + z) \Delta t + O(\Delta t^2). \tag{7}
$$

Then we apply again the first-order expansion of $f(x, u+z)$, yields

$$
x(t + \Delta t) = x + f(x, u) \Delta t + f_u(x, u)^T z \Delta t + O(\Delta t^2). \tag{8}
$$

Also, we consider the first-order expansion of $V(x(t + \Delta t))$:

$$
\begin{aligned}
V(x(t + \Delta t)) = &V(x) + \langle \nabla V, f(x, u) + \nabla_u f(x, u)^T z \rangle \Delta t \\
&+ O(\Delta t^2).
\end{aligned} \tag{9}
$$

To have the safety guarantees, the condition $V(x(t + \Delta t)) < c$ needs to be satisfied, that is,

$$
\langle \nabla V(x), f(x, u) + \nabla_u f(x, u)^T z \rangle < \frac{c - V(x)}{\Delta t}. \tag{10}
$$

Expanding and simplifying this expression gives the following bounds:

$$
\langle \nabla_u f(x, u) \nabla V(x), z \rangle < \frac{c - V(x)}{\Delta t} - \langle \nabla V(x), f(x, u) \rangle. \tag{11}
$$

Using the Cauchy-Schwarz inequality, a sufficient way to bound the noise z is

$$
||\nabla_u f(x, u) \nabla V(x)||_2 ||z||_2 < \frac{c - V(x)}{\Delta t} - \langle \nabla V(x), f(x, u) \rangle. \tag{12}
$$

If f is a control-affine function $f(x, u) = f_1(x) + f_2(x) u$, then we can bound z as:

$$
||z||_2 < \frac{\frac{c - V(x)}{\Delta t} - \langle \nabla V(x), f(x, u) \rangle}{||f_2(x)||_2 ||\nabla V(x)||_2}. \tag{13}
$$

Since Theorem 5 requires that under the optimal controller $\pi^*(x)$, $\langle \nabla V(x), f(x, \pi^*(x)) \rangle \leq 0$, this shows that the bound imposed on z is positive.

Remark 1: Numerical solutions in section IV show that in harmonic CLBFs, the distinction between safe and unsafe regions is unclear. To make the distinction more transparent, we replace the CLBF as a solution to Laplace's equation with a solution to Poisson's equation $\nabla^2 V = -6$ in this paper, meaning V is a superharmonic function. However, this method also poses a risk that V has local minima in the interior of its domain [2]. This is undesirable as local minima are harder to escape compared to saddle points. We will compare harmonic CLBFs with superharmonic CLBFs with numerical results.

IV. NUMERIAL RESULTS

In this section, we will explore two different reach-avoid environments, one with four small unsafe regions and the other with two big unsafe regions, for three different systems: Roomba, DiffDrive, and CarRobot [15]. Except that, we also perform a reach-avoid task for 2D Quadrotor [5]. The dynamics of each of these systems are provided in Appendix I along with control inputs that minimize the expression in Equation (5) for any $x \in \mathcal{S}_{safe}$. We use $c = 1$ in the CLBF defined in Definition 1 for all numerical results.

For each environment and system, we compute the harmonic ($\nabla^2 V = 0$) and superharmonic ($\nabla^2 V = -6$) CLBF numerically using finite element methods. For both of the CLBFs, we compute the trajectories of the system with 1,000 different randomly initialized initial conditions and count the number of time steps (of size $\Delta t = 0.1$) it takes for the system to reach $\mathcal{S}_{goal}$. We test this setting with both deterministic ($\sigma = 0$) and stochastic ($\sigma = 0.1$, clipped by the upper bound in Equation (13)) controllers and report the mean time taken to reach the goal (μ_T), the standard deviation of the time taken to reach the goal (σ_T), the number of times the system ends in an unsafe region (not included in μ_T, σ_T), and the number of times the system does not reach $\mathcal{S}_{goal}$ after 1,000 time steps (not included in μ_T, σ_T).

A. Problem I

In this problem, we explore an environment that contains a goal region near the origin and four unsafe regions in the interior of the domain. The results with three car-like dynamical systems, Roomba, DiffDrive, and CarRobot, are reported.

$$S = [-1, 1] \times [-1, 1] \tag{14}$$
$$\mathcal{S}_{goal} = [-0.1, 0.1] \times [-0.1, 0.1] \tag{15}$$
$$\mathcal{S}_{unsafe} = \partial S \cup C_1 \cup C_2 \cup C_3 \cup C_4, \tag{16}$$

with the subdomains of the unsafe region given by

$$C_1 = [-0.5, -0.3] \times [-0.5, -0.3] \tag{17}$$
$$C_2 = [-0.5, -0.3] \times [0.3, 0.5] \tag{18}$$
$$C_3 = [0.3, 0.5] \times [-0.5, -0.3] \tag{19}$$
$$C_4 = [0.3, 0.5] \times [0.3, 0.5]. \tag{20}$$

Initial conditions are sampled as:

$$x(0), y(0) \sim U[-0.9, -0.6] \cup [0.6, 0.9] \tag{21}$$
$$\theta(0) \sim U[0, 2\pi]. \tag{22}$$

We derive a harmonic CLBF using finite element methods using DOLFIN [14]. We used piecewise linear trial functions and a triangular mesh. The domain has been divided into 5,000 triangular meshes of equal area. This numerical solution is plotted in 2D view in Fig. 1a.

Although in theory, the solution should take values less than 1.0 outside the unsafe region, the distinction is highly unclear in the numerical solution. To clarify the distinction, we replace the Laplace equation with the Poisson equation $\nabla^2 V = -6$. We plotted the solution in 2D view in Fig. 1b. This solution shows a better distinction between safe and unsafe regions, but it shows more points where $V(x)$ has a local optimum. Local optima are undesirable as $\langle f(x, u), \nabla V(x) \rangle = 0$ in those points so $V(x)$ is unlikely to show a significant decrease.

System	$\nabla^2 V$	σ	μ_T	σ_T	unsafe	no reach
Roomba	0	0	97.278	65.607	0	105
Roomba	-6	0	58.224	50.388	0	459
Roomba	**0**	**0.1**	**114.706**	**81.527**	**0**	**0**
Roomba	-6	0.1	139.606	180.247	0	185
DiffDrive	0	0	301.403	216.495	0	95
DiffDrive	-6	0	127.045	115.348	0	492
DiffDrive	**0**	**0.1**	**239.759**	**220.689**	**0**	**2**
DiffDrive	-6	0.1	158.837	167.421	0	5
CarRobot	0	0	452.125	323.942	0	193
CarRobot	**-6**	**0**	**277.398**	**253.148**	**0**	**47**
CarRobot	0	0.1	21.200	69.889	0	930
CarRobot	-6	0.1	26.887	88.174	0	920

TABLE I: Results for Problem I. Each system was run with 1,000 different randomly initialized initial conditions under harmonic and superharmonic CLBFs with deterministic and stochastic controllers.

The numerical results are listed in Table I. Roomba achieves the best results under a harmonic CLBF with a stochastic control policy as all the trajectories converged to

$\mathcal{S}_{goal}$. DiffDrive achieves the best results under a harmonic CLBF with a stochastic control policy with a 99.8% safety rate and similar results with a superharmonic CLBF with a stochastic control policy with a 99.5% safety rate. CarRobot achieves the best results under a superharmonic CLBF with a deterministic policy with a 95.3% safety rate.

B. Problem II

In this problem, we explore an environment that contains a goal region near the origin and two unsafe regions in the interior of the domain. In a similar vein to problem I, the three models are tested with this reach-avoid task.

$$S = [-1, 1] \times [-1, 1] \tag{23}$$
$$\mathcal{S}_{goal} = [-0.1, 0.1] \times [-0.1, 0.1] \tag{24}$$
$$\mathcal{S}_{unsafe} = \partial S \cup C_5 \cup C_6, \tag{25}$$

with the subdomains of the unsafe region given by

$$C_5 = [-0.5, -0.3] \times [-0.5, 0.5] \tag{26}$$
$$C_6 = [0.3, 0.5] \times [-0.5, 0.5]. \tag{27}$$

Initial conditions are sampled as:

$$x(0) \sim U[-0.9, -0.6] \cup [0.6, 0.9] \tag{28}$$
$$y(0) \sim U[-0.3, 0.3] \tag{29}$$
$$\theta(0) \sim U[0, 2\pi]. \tag{30}$$

The numerical solution to the harmonic CLBF for this problem is plotted in 2D view in Fig. 1c.

To make the distinction between safe and unsafe regions clearer, we plotted the solution to the Poisson equation $\nabla^2 V = -6$ in 2D view in Fig. 1d. The numerical results are

System	$\nabla^2 V$	σ	μ_T	σ_T	unsafe	no reach
Roomba	0	0	85.600	54.603	12	0
Roomba	**-6**	**0**	**202.568**	**199.603**	**0**	**91**
Roomba	0	0.1	86.846	55.789	9	78
Roomba	-6	0.1	48.033	21.363	0	880
DiffDrive	**0**	**0**	**258.502**	**168.726**	**0**	**55**
DiffDrive	-6	0	96.077	28.211	0	730
DiffDrive	0	0.1	265.348	164.137	7	9
DiffDrive	-6	0.1	209.452	171.338	0	118
CarRobot	**0**	**0**	**380.014**	**230.751**	**30**	**124**
CarRobot	-6	0	415.246	257.179	0	570
CarRobot	0	0.1	264.435	205.930	45	942
CarRobot	-6	0.1	342.060	262.481	0	987

TABLE II: Results for Problem II. Each system was run with 1,000 different randomly initialized initial conditions under harmonic and superharmonic CLBFs with deterministic and stochastic controllers.

posted in Table II. Roomba achieves the best results under a superharmonic CLBF with a deterministic control policy with a 90.9% safety rate. DiffDrive achieves the best results under a harmonic CLBF with a deterministic control policy with a 94.5% safety rate. And CarRobot achieves the best results under a harmonic CLBF with a deterministic policy with a 84.6% safety rate, although it has 30 trajectories converging to $\mathcal{S}_{unsafe}$ and 124 trajectories not converging to $\mathcal{S}_{goal}$. Furthermore, this table shows that deterministic
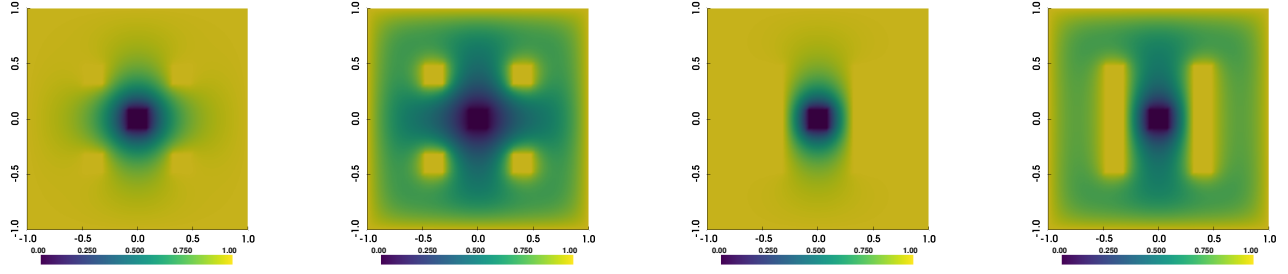
(a) Problem I with $\nabla^2 V = 0$ (b) Problem I with $\nabla^2 V = -6$ (c) Problem II with $\nabla^2 V = 0$ (d) Problem II with $\nabla^2 V = -6$

Fig. 1: Contour plots of the CLBFs for problems I and II. (a) & (b) The plots for problem I with the CLBF obtained by solving Laplace equation (left) and Poisson's equation (right) respectively; (c) & (d) The plots for problem II with the CLBF obtained by solving Laplace equation (left) and Poisson's equation (right) respectively.

control policies achieve better results than stochastic control policies and have significantly more trajectories that converge to $\mathcal{S}_{goal}$.

Tables I and II show that deterministic policies under superharmonic CLBFs always avoid $\mathcal{S}_{unsafe}$ as expected since the Poisson equation makes the distinction between safe and unsafe regions more evident. But it comes with the drawback that there is a high risk that the trajectory will converge to a local minimum and not reach $\mathcal{S}_{goal}$. Deterministic policies generally outperform stochastic policies as the noise added to control inputs comes with the risk that it drives the trajectory away from the goal or towards an unsafe region. Despite using control inputs that solve the minimization problem in Equation (5), in many episodes, the trajectory does not converge to $\mathcal{S}_{goal}$. This is likely due to numerical errors associated with estimating ∇V or with solving the system dynamics.

C. 2D Quadrotor

In this problem, we control a quadrotor to land at a safe ground-level region while avoiding two obstacles. The dynamics of the quadrotor are in Appendix V-D. We are running the same problem as in [6] and the horizontally flipped problem in [5]. The safe and unsafe subsets are given as the following.

$$S = [-1, 2] \times [0, 2] \tag{31}$$
$$\mathcal{S}_{goal} = [-1, 0.5] \times [0.25, 0.75] \tag{32}$$
$$\mathcal{S}_{unsafe} = \partial S \cup C_7 \cup C_8 \cup C_9, \tag{33}$$

with the subdomains of the unsafe region given by

$$C_7 = [-1, 0] \times [1.25, 2] \tag{34}$$
$$C_8 = [-1, 2] \times [0, 0.25] \tag{35}$$
$$C_9 = [0.5, 1] \times [0, 1]. \tag{36}$$

Initial conditions are sampled as:

$$x(0) \sim U[1.4, 1.6] \tag{37}$$
$$z(0) \sim U[0.3, 1.5] \tag{38}$$
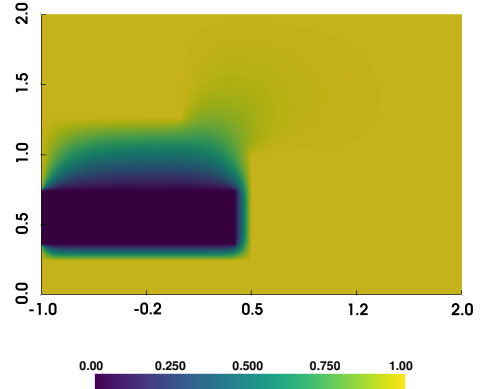$$x'(0), z'(0), \theta(0), \theta'(0) \sim U[-0.1, 0.1]. \tag{39}$$

Fig. 2: Countour plot of the harmonic CLBF for the 2D Quadrotor environment

The numerical solution to the Harmonic CLBF for this problem is plotted in Fig. 2.

In this case, we use a neural network deterministic controller $\pi_\kappa(\mathbf{x})$ parameterized by κ, where $\mathbf{x} = (x, \dot{x}, z, \dot{z}, \theta, \dot{\theta})$, and we train the controller using model-predictive control (MPC). After numerically solving the ode $\dot{\mathbf{x}} = f(\mathbf{x}, \pi_\kappa(\mathbf{x}))$ for a horizon of 128 time steps with time step size $\Delta t = 0.01$ using RK4, we compute the loss as the cosine similarity of $f(\mathbf{x}_t, \pi_\kappa(\mathbf{x}_t))$ and $\nabla V(\mathbf{x}_t)$ for all time steps. The time step size is 0.01. We repeat this for 100,000 different initial conditions.

μ_T	σ_T	unsafe	no reach	safety rate
70.350	16.114	43	96	86.1%

TABLE III: Results for 2D Quadrotor

After training, we test the controller on the environment for 1,000 different initial conditions. The numerical results are posted in Table III. The safety rate of the controller is 86.1%, which is higher than the safety rates for rCLBF-QP (83%) and Robust-MPC (53%) reported in [5]. To compare with [6], we have computed the trajectories of the quadrotor for the initial conditions $\mathbf{x}_1 = [1.5, 0, 0.35, 0.0.0]$ and $\mathbf{x}_2 = [1.5, 0, 1.2, 0.0.0]$. In Fig. 3, we overlay the trajectories on

a superharmonic CLBF plot with $\nabla^2 V = -6$ that we computed to make the distinction between safe and unsafe regions clearer. The trajectories in the plot show that the quadrotor can effectively avoid unsafe regions and enter the goal regions.
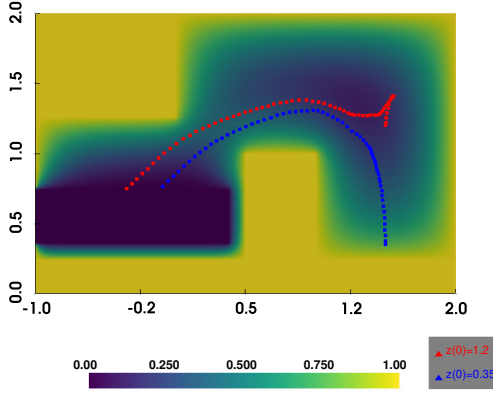

Fig. 3: Plot of trajectories on the 2D Quadrotor environment with $z(0) = 0.35$ (blue) and $z(0) = 1.2$ (red).

V. CONCLUSIONS

In this paper, we introduced harmonic CLBFs that exploit the maximum principle that harmonic functions satisfy to encode the properties of CLBFs. This paper is the first to unify harmonic functions with CLBFs for an application to control theory. We select control inputs that maximize the inner product of the system dynamics with the steepest descent direction of the harmonic CLBF. This has been applied to reach-avoid problems and demonstrated a low risk of entering unsafe regions while converging to the goal region.

APPENDIX I: DYNAMICS OF EACH SYSTEM

A. Roomba

The dynamics of the Roomba are as follows:

$$\dot{x} = v\cos\theta, \quad \dot{y} = v\sin\theta, \quad \dot{\theta} = \omega,$$

where $v \in [-1, 1]$ and $\omega \in [-1, 1]$ are the control inputs.

The minimization problem to Equation (5) is:

$$\inf_{v\in[-1,1],\omega\in[-1,1]} (v\cos\theta, v\sin\theta)(V_x(x, y), V_y(x, y))^T.$$

The infimum is reached when v maximizes the negative of the magnitude of the inner product, and w is set to the direction that maximizes the magnitude of the inner product, which yields

$$v = -\text{sign}(V_x(x, y)\cos\theta + V_y(x, y)\sin\theta),$$
$$w = \text{sign}(-V_x(x, y)\cos\theta + V_y(x, y)\sin\theta).$$

These control input pairs are used in computing the results in Tables I and II.

B. DiffDrive

The dynamics of the diff-drive robot are as follows:

$$\dot{x} = (u_L + u_R)\frac{r}{2}\cos\theta, \dot{y} = (u_L + u_R)\frac{r}{2}\sin\theta,$$
$$\dot{\theta} = (u_R - u_L)\frac{r}{2d},$$

where $u_L \in [-1, 1]$ and $u_R \in [-1, 1]$ are the control inputs subject to the constraint $|u_L| + |u_R| \leq 1$. $r = 0.1$ and $d = 0.1$.

The minimization problem to Equation (5) is:

$$\inf_{|u_L|+|u_R|\leq 1} ((u_L + u_R)\frac{r}{2}\cos\theta, (u_L + u_R)\frac{r}{2}\sin\theta)\nabla V(x, y).$$

The infimum is reached when $u_L + u_R$ maximizes the negative of the magnitude of the inner product, and $u_R - u_L$ is set to the direction that maximizes the magnitude of the inner product, which yields

$$u_R = [\text{sign}(-V_x(x, y)\cos\theta + V_y(x, y)\sin\theta)$$
$$- \text{sign}(V_x(x, y)\cos\theta + V_y(x, y)\sin\theta)]/2,$$
$$u_L = [-\text{sign}(-V_x(x, y)\cos\theta + V_y(x, y)\sin\theta)$$
$$- \text{sign}(V_x(x, y)\cos\theta + V_y(x, y)\sin\theta)]/2.$$

C. CarRobot

The dynamics of the car-like robot are as follows:

$$\dot{x} = v\cos\theta, \quad \dot{y} = v\sin\theta, \quad \dot{\theta} = v\tan(\psi)/l, \quad \dot{\psi} = w,$$

where $v \in [-1, 1]$ and $w \in [-1, 1]$ are the control inputs subject to the constraint $|v| \leq |w|$. $l = 0.1$. $\psi(0)$ is initialized to 0. We set $l = 0.1$.

The minimization problem to Equation (5) is:

$$\inf_{|v|\leq|w|\leq 1} (v\cos\theta, v\sin\theta)(V_x(x, y), V_y(x, y))^T.$$

The infimum is reached when v maximizes the negative of the magnitude of the inner product, and θ is set to the direction that maximizes the magnitude of the inner product.

$$v = -\text{sign}(V_x(x, y)\cos\theta + V_y(x, y)\sin\theta),$$
$$w = v\text{sign}(-V_x(x, y)\cos\theta + V_y(x, y)\sin\theta) - \text{sign}(\psi).$$

D. 2D Quadrotor

We use the implementation of the 2D quadrotor environment by [21]. The dynamics of the quadrotor are as follows:

$$\ddot{x} = \sin\theta(T_1 + T_2)/m$$
$$\ddot{z} = \cos\theta(T_1 + T_2)/m - g$$
$$\ddot{\theta} = (T_2 - T_1)d/I_{yy},$$

where $T_1, T_2 \in [g/2 - 4, g/2 + 4]$ are the control inputs. $m = 0.033$, $I_{yy} = 1.436e - 05$, $g = 9.81$ and $d = 0.00397/\sqrt{2}$ are constants.

REFERENCES

[1] Dario Amodei Alex Ray, Joshua Achiam. Benchmarking safe exploration in deep reinforcement learning. *OpenAI*, 2019.

[2] Sheldon Axler, Paul Bourdon, and Ramey Wade. *Harmonic function theory*, volume 137. Springer Science & Business Media, 2013.

[3] Jason Choi, Fernando Castaneda, Claire J Tomlin, and Koushil Sreenath. Reinforcement learning for safety-critical control under model uncertainty, using control lyapunov functions and control barrier functions. *arXiv preprint arXiv:2004.07584*, 2020.

[4] Christopher I Connolly, J Brian Burns, and Rich Weiss. Path planning using laplace's equation. In *Proceedings., IEEE International Conference on Robotics and Automation*, pages 2102–2106. IEEE, 1990.

[5] Charles Dawson, Zengyi Qin, Sicun Gao, and Chuchu Fan. Safe nonlinear control using robust neural lyapunov-barrier functions. In *5th Annual Conference on Robot Learning*, 2021.

[6] Desong Du, Shaohang Han, Naiming Qi, Haitham Bou Ammar, Jun Wang, and Wei Pan. Reinforcement learning for safe robot control using control Lyapunov barrier functions, 2023.

[7] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points — online stochastic gradient for tensor decomposition, 2015.

[8] David Gilbarg and Neil S. Trudinger. *Elliptic Partial Differential Equations of Second Order, Second Edition*. Springer-Verlag, 1983.

[9] Thomas Gurriet, Andrew Singletary, Jacob Reher, Laurent Ciarletta, Eric Feron, and Aaron Ames. Towards a framework for realizable safety critical control through active set invariance. In *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, pages 98–106. IEEE, 2018.

[10] Nicolas Heess, Dhruva Tb, Srinivasan Sriram, Jay Lemmon, Josh Merel, Greg Wayne, Yuval Tassa, Tom Erez, Ziyu Wang, SM Eslami, et al. Emergence of locomotion behaviours in rich environments. *arXiv preprint arXiv:1707.02286*, 2017.

[11] Ching-An Cheng Hoai-An Nguyen. Provable reset-free reinforcement learning by no-regret reduction. *International Conference on Machine Learning*, 2023.

[12] Jin-Oh Kim and Pradeep Khosla. Real-time obstacle avoidance using harmonic potential functions. *IEEE Transactions on Robotics and Automation*, June 1992.

[13] Jason D. Lee, Max Simchowitz, Michael I. Jordan, and Benjamin Recht. Gradient descent converges to minimizers, 2016.

[14] A. Logg and G. N. Wells. Dolfin: Automated finite element computing. *ACM Transactions on Mathematical Software 37*, 2010.

[15] Kevin M. Lynch and Frank C. Park. *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, 2017.

[16] Yiming Meng, Yinan Li, Maxwell Fitzsimmons, and Jun Liu. Smooth converse lyapunov-barrier theorems for asymptotic stability with safety constraints and reach-avoid-stay specifications. *Automatica*, 144:110478, 2022.

[17] Murray H. Protter and Hans F. Weinberger. *Maximum Principles in Differential Equations*. Springer, 1984.

[18] S. Kawamura S. Akishita and K. Hayashi. Laplace potential for moving obstacle avoidance and approach of a mobile robot. In *I990 Japan-USA Symposium on Flexible Automation, A Pacific Rim Conference*, 1990.

[19] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[20] A Yanushauskas. The zeros of the gradient and the hessian of an harmonic function. *Siberian Mathematical Journal*, 10(3):497–501, 1969.

[21] Zhaocong Yuan, Adam W. Hall, Siqi Zhou, Lukas Brunke, Melissa Greeff, Jacopo Panerati, and Angela P. Schoellig. Safe-control-gym: A unified benchmark suite for safe learning-based control and reinforcement learning in robotics. *IEEE Robotics and Automation Letters*, 7(4):11142–11149, 2022.