Maneuvering with safety guarantees using control barrier functions ⋆

Mathias Marley * **Roger Skjetne** * **Erlend Basso** *
Andrew R. Teel **

* *Norwegian University of Science & Technology, Trondheim, Norway*
{mathias.marley,roger.skjetne,erlend.basso}@ntnu.no
** *University of California, Santa Barbara, USA, teel@ucsb.edu*

Abstract: Control barrier functions (CBFs) ensure safety of controlled dynamical systems, by restricting the control inputs to render desired sets forward invariant. In this paper we propose a dynamic guidance scheme for autonomous vehicles, using CBFs to reactively generate an obstacle-free trajectory. By implementing the safety constraints on a kinematic guidance level, rather than on a lower-level control layer, we do not need to account for uncertainty in the ship dynamics explicitly. Moreover, for ships with well-proven and resilient control systems, this is an appropriate interface level, since it does not require modification of lower-level feedback control. The guidance scheme is applied to maneuvering of underactuated ships, using a virtual vessel with unicycle dynamics to trace out a feasible trajectory.

Keywords: Control barrier functions, maneuvering, AMVs, path following, obstacle avoidance.

1. INTRODUCTION

Control barrier functions (CBFs) (Ames et al. (2019)) ensure safety of controlled dynamical systems, by restricting the control input to admissible sets that depend on the instantaneous state of the system. In CBF design, the safety objective is solved separately from the nominal control objective, thus providing flexibility in the design process. The standard CBF formulation assumes that the system dynamics are exactly known, which precludes safety guarantees for systems with unknown dynamics (Xu et al. (2015)). A robust CBF formulation is proposed by Emam et al. (2019), where the safety-critical control input accounts for an assumed worst-case bounded disturbance. In Taylor and Ames (2020), the authors propose adaptive CBFs to ensure safety of systems with parametric uncertainty. A drawback of the proposed adaptive CBFs is that, similar to adaptive control Lyapunov functions (CLFs), each sublevel set of the CBFs, not just the subzero level set, is rendered forward invariant, imposing unnecessary restrictions on the system dynamics. Improvements to adaptive CBF designs are proposed by Maghenem and Sanfelice (2021) and Isaly et al. (2021).

CBF designs for obstacle avoidance for autonomous ships are proposed by Thyri et al. (2020) and Basso et al. (2020), with safety constraints implemented on the actuator level. In the former, the system dynamics of the ship are assumed fully known. The latter accounts for unknown ocean currents, using integral action to solve the path-following problem with zero tracking error, and the robust CBF formulation of Emam et al. (2019) for obstacle avoidance. While this guarantees safety of the ship, large transients occur following an evasive manuever, due to integral wind-up when the ship is forced to deviate from the path.

In this paper we propose implementing the safety constraints on a dynamic guidance level, reactively modifying the desired trajectory if unknown obstacles are encountered during the voyage. This is solved by having a virtual vessel tracing out the desired trajectory. The virtual vessel follows a reference path when safety allows it, and deviates from the reference path when safety demands it. Feedback from the ship to the guidance system is enabled by a *directional* gradient feedback term, along the direction of the reference path. Feasibility of the trajectory is achieved by having the virtual vessel emulate the dynamic capabilities of the ship. As a recurring example, we present a guidance design suitable for underactuated ships at transit speeds.

The remainder of this paper is organized as follows. The problem formulation is stated in Section II. The maneuvering problem is briefly reviewed in Section III. In Section IV we review and expand on the theory of CBFs. The main contribution of this paper is found in Section V, where a reactive guidance scheme is proposed. Numerical simulations are presented in Section VI. Section VII concludes the paper.

Notation: $\mathbb{R}$ is the set of real numbers and $\mathbb{R}^n$ is the n-dimensional Euclidean space. $\mathbb{R}_{\geq 0}$ and $\mathbb{R}_{>0}$ are the set of non-negative and positive numbers, respectively. For a function $f : \mathbb{R}^n \to \mathbb{R}^m$, the Jacobian matrix is denoted $\frac{\partial f}{\partial x} \in \mathbb{R}^{m \times n}$. When convenient, we use the Lie derivative notation: $L_f B(x) := \frac{\partial B}{\partial x} f(x)$, where $B : \mathbb{R}^n \to \mathbb{R}$ is a scalar function. For a set K, ∂K and $\mathrm{Int} K$ denote the boundary and interior, respectively. We use $\mathcal{K}_e$ to denote extended class-$\mathcal{K}$ functions, i.e. the family of functions $\alpha : \mathbb{R} \to \mathbb{R}$ that are strictly increasing with $\alpha(0) = 0$. Finally, $\dot{x}$ is the time derivative of x, and $|x| := \sqrt{x^\top x}$.

⋆ Research supported in part by the Research Council of Norway through the Centre of Excellence NTNU AMOS (RCN prj 223254), SFI AutoShip (RCN project 309230), and the Air Force Office of Scientific Research under grant FA9550-18-1-0246.

Preliminaries: We adopt the unit circle representation of orientation used in Marley et al. (2020). Accordingly, we define the unit circle as $\mathcal{S}^1 := \{z \in \mathbb{R}^2 : z^\top z = 1\}$, and group of planar rotations as $SO(2) := \{R \in \mathbb{R}^{2\times 2} : R^\top R = I, det(R) = 1\}$. The map $R : \mathcal{S}^1 \to SO(2)$ is defined by $R(z) := [z\ Sz]$, where

$$S := \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \in SO(2) \tag{1}$$

is the 90 degree rotation matrix. For two vectors $z_a, z_b \in \mathcal{S}^1$ corresponding to angles $a, b \in \mathbb{R}$, we have the convenient calculation rules $z_{a+b} = R(z_b)z_a = R(z_a)z_b$ and $z_{a-b} = R(z_b)^\top z_a$. Note the relations $R(z)^\top z = [1\ 0]^\top =: \varepsilon_1$ and $R(z)^\top Sz = [0\ 1]^\top =: \varepsilon_2$.

For a ship with position $p \in \mathbb{R}^2$, yaw angle $\psi \in [-\pi, \pi)$, yaw rate $r \in \mathbb{R}$, and body-fixed linear velocities $\nu \in \mathbb{R}^2$, the kinematics is commonly expressed as

$$\dot{p} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \nu, \quad \dot{\psi} = r. \tag{2}$$

An equivalent representation, with z representing the vessel orientation, is given by $\dot{p} = R(z)\nu$ and $\dot{z} = rSz$.

Central to this paper is the notion of forward invariance, defined below for the differential inclusion

$$\dot{x} \in F(x), \tag{3}$$

with state $x \in \mathbb{R}^n$ and set-valued mapping $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$.

Definition 1. The set $K \subset \mathbb{R}^n$ is (strongly) forward invariant for (3) if each solution $x : [0, T) \to \mathbb{R}^n$ of (3) with $x(0) \in K$ satisfies $x(t) \in K$ for all $t \in [0, T)$.

If K is a safe set, we refer to forward invariance of K as safety. Definition 1 does not require solutions to be complete. For instance, if K is a non-compact set, solutions may escape to infinity inside the set K.

2. PROBLEM FORMULATION

The path-following problem for autonomous vehicles is for the vehicle to converge to, and thereafter follow, a geometric path, while satisfying a dynamic assignment along the path (typically a prescribed velocity profile).

Let $y \in \mathbb{R}^p$ be the output of the nonlinear system

$$\dot{x} = f(x, u) \quad y = h(x), \tag{4}$$

with state $x \in \mathbb{R}^n$ and input $u \in \mathbb{R}^m$. If (4) represents a vehicle, the output y will typically be the position or pose (position and orientation) of the vehicle. Given a sufficiently smooth map $s \mapsto y_r(s) \in \mathbb{R}^p$, the reference path Y_r is a one-dimensional manifold defined by

$$Y_r := y_r(\mathbb{R}) = \{y \in \mathbb{R}^p : \exists s \in \mathbb{R} \text{ s.t. } y = y_r(s)\}. \tag{5}$$

The geometric task of convergence to the path may be solved by driving the error $|y(t) - y_r(s(t))| \to 0$, with the path-variable s as a controlled state. Contrary to trajectory tracking, there are no temporal specifications in path following. This increases flexibility in the control design process (Aguiar et al. (2004)), and improves robustness (Skjetne et al. (2005)). In particular, it enables feedback from the vehicle output to the guidance system.

In path-following design, it is common to assume that the path is safe. For systems evolving in dynamic or uncertain environments, this may not be the case. If y is the output of a system that is affine in the control input, we may construct a CBF that ensures safety, forcing y to leave the set Y_r if safety demands it. However, as discussed in the introduction, this approach has two major limitations: 1) accounting for uncertainty in the system dynamics in the CBF formulation is non-trivial, 2) adaptive control schemes that rely on integrating the tracking error will result in poor performance if the system output is forced to deviate from the desired trajectory.

To mitigate these potential issues, we propose using an auxiliary point $y_d \in \mathbb{R}^p$ to reactively trace out a safe trajectory for y to follow. Contrary to y_r, y_d is not constrained to the manifold Y_r, but can move freely in the output space $\mathbb{R}^p$. Path-following for y is then achieved by controlling $|y(t) - y_d(t)| \to 0$, and simultaneously $|y_d(t) - y_r(s(t))| \to 0$. To enable feedback from the vehicle to the guidance system, we introduce a feedback term in the dynamics of y_d, along the direction of the path tangent

$$\tau(s) := \frac{\partial y_r}{\partial s}(s). \tag{6}$$

Problem statement: Given a sufficiently smooth reference path represented by the map $s \mapsto y_r(s)$, design a guidance system that reactively traces out a safe trajectory. The desired position y_d shall adhere to the following control objectives:

(1) *Nominal objective:* y_d shall converge to and follow the reference path, i.e. $\lim_{t\to\infty} |y_d(t) - y_r(s(t))| = 0$.
(2) *Safety objective:* Given some unsafe domain $K_u \subset \mathbb{R}^p$, render $K_s := \mathbb{R}^p \setminus K_u$ forward invariant for y_d.

The safety objective shall have the highest priority. Moreover, the design shall enable feedback from the vehicle to the guidance system.

2.1 Recurring example

We will use a recurring example to exemplify our design, where we design a guidance system for underactuated ships. We let $p \in \mathbb{R}^2$ represent the position of the ship, and $p_d \in \mathbb{R}^2$ represent the desired position. The reference path, represented by the map $s \mapsto p_r(s)$, is assumed sufficiently smooth and feasible for the ship to follow. The objective is to guide the ship along the path with constant reference speed $v_r \in \mathbb{R}_{>0}$, while avoiding obstacles encountered during the voyage. For ease of exposition we assume the path tangent has unit length, i.e. $|\frac{\partial p_r}{\partial s}(s)| = 1$, which implies $\tau(s) = \frac{\partial p_r}{\partial s}(s) \in \mathcal{S}^1$.

3. THE MANUEVERING PROBLEM

To motivate our reactive guidance design, in particular the directional gradient feedback term introduced in Section 5.1, we review the maneuvering problem as defined in Skjetne et al. (2004) and Skjetne (2005). The traditional maneuvering problem is a guidance scheme for path following, where the control objective is separated into a geometric task (converge to the desired path), and a dynamic task (typically following the path at a prescribed speed). Given a speed assignment $v(s)$ for $\dot{s}$, the two tasks are stated as:

- Geometric task: $\lim_{t\to\infty} |y(t) - y_r(s(t))| = 0$.

- Dynamic task: $\lim_{t\to\infty} |\dot{s}(t) - \upsilon(s(t))| = 0$.

3.1 Maneuvering control design

The geometric task is equivalent to stabilizing the set

$$\mathcal{A} := \{(s, x) \in \mathbb{R} \times \mathbb{R}^n : h(x) = y_r(s)\}. \tag{7}$$

Let $V : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ be a control Lyapunov function (CLF) relative to the set $\mathcal{A}$. Feedback from y to $y_r(s)$ is obtained by the assignment

$$\dot{s} = f_s(s, x) := \sigma(s, x)\upsilon(s) - \omega_s(s, x). \tag{8}$$

where $\omega_s : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ is a gradient feedback given by

$$\omega_s(s, x) := \mu_s \frac{\partial V}{\partial s}(s, x), \quad \mu_s \geq 0. \tag{9}$$

With sufficiently large gain μ_s this will quickly move y_r to a point that (at least locally) minimizes V. In (8), we have additionally allowed for a smooth gain $\sigma : \mathbb{R} \times \mathbb{R}^n \to [0, 1]$, that shall satisfy $(s, x) \in \mathcal{A} \implies \sigma(s, x) = 1$. A similar strategy was used in Skjetne et al. (2011), and enables us to slow down the reference point $y_r(s)$ when $(s, x) \notin \mathcal{A}$. If $\mathcal{A}$ is rendered asymptotically stable, by appropriate design of the vehicle controller, the dynamic task is solved in the limit since $(s, x) \in \mathcal{A} \implies f_s(s, x) = \upsilon(s)$.

The maneuvering guidance system, represented by the update law $\dot{s} = f_s(s, x)$ and the map $s \mapsto y_r(s)$, outputs the reference position $y_r(s)$ with dynamics

$$\dot{y}_r(s) = \tau(s)(\sigma(s, x)\upsilon(s) - \omega_s(s, x)), \quad y_r(s_0) \in Y_r. \tag{10}$$

Moreover, $\tau(s)\upsilon(s)$ often serves as reference signal for $\dot{y}$.

3.2 Recurring example

For a ship with $\nu_1 \gg |\nu_2|$, i.e., surge speed much greater than sway speed, the kinematics are approximated by $\dot{p} \approx \nu_1 z$. Accordingly, the unicycle model

$$\dot{p}_d = v_d z_d, \quad v_d \in \mathbb{R}, \quad z_d \in \mathcal{S}^1, \tag{11}$$

is a reasonable representation of an underactuated ship in transit. The system (11) will serve as basis for the dynamics of our virtual vessel. We solve the maneuvering problem for (11) relative to the path $p_r(s)$, with orientation z_d and speed v_d as control inputs. To this end, define the error variable

$$e := R(\tau(s))^\top (p_d - p_r(s)). \tag{12}$$

We refer to $e_1 = \varepsilon_1^\top e = (p_d - p_r(s))^\top \tau(s)$ as the along-track error, and $e_2 = \varepsilon_2^\top e = (p_d - p_r(s))^\top S\tau(s)$ as the cross-track error. To drive $e_2 \to 0$, we orient z_d towards some point in front of $p_r(s)$, along the direction of the path tangent $\tau(s)$. This is achieved by

$$z_d = \kappa_z(s, p_d) := R(\tau(s)) z_\Delta(e), \tag{13}$$

where

$$z_\Delta(e) := \frac{1}{\sqrt{\Delta^2 + e_2^2}} \begin{bmatrix} \Delta \\ -e_2 \end{bmatrix} \tag{14}$$

is the Line-Of-Sight (LOS) algorithm (Fossen (2011)) represented on vector form, with lookahead-distance $\Delta > 0$.

Since $|\tau(s)| = 1$, $\forall s \in \mathbb{R}$, the dynamic task becomes $\dot{s} \to v_r$. At this point we directly assign $v_d = v_r$, and let $\dot{s}$ adhere to the along-track speed of p_d. This is obtained by

$$f_s(s, p_d) := \varepsilon_1^\top z_\Delta(e) v_r + \mu_s e_1, \tag{15}$$

where $\mu_s \in \mathbb{R}_{>0}$ must be positive to drive the along-track error to zero. Note that $\varepsilon_1^\top z_\Delta(e)$ is the along-track speed of p_d. Stability is shown by the Lyapunov function

$$V(s, p_d) := \frac{(p_d - p_r(s))^\top (p_d - p_r(s))}{2} = \frac{e^\top e}{2}. \tag{16}$$

Differentiating V in (16) along the solutions of

$$\dot{p}_d = v_r \kappa_z(s, p_d), \quad \dot{s} = f_s(s, p_d), \tag{17}$$

results in

$$\dot{V} \frac{\partial V}{\partial p_d} \dot{p}_d + \frac{\partial V}{\partial s} \dot{s} = -\frac{v_r e_2^2}{\sqrt{\Delta^2 + e_2^2}} - \mu_s e_1^2. \tag{18}$$

This shows uniform global asymptotic stability (UGAS) of the set $\{p_d, s : |p_d - p_r(s)| = 0\}$ (Skjetne et al. (2011)). Moreover, $\lim_{t\to\infty} \dot{s}(t) = v_r$, since $|p_d - p_r(s)| = 0 \implies f_s(s, p_d) = v_r$.

4. CONTROLLED FORWARD INVARIANCE

To satisfy the safety objective, we will use control barrier functions. CBFs, first introduced in Wieland and Allgöwer (2007), merge the ideas of CLFs (Artstein (1983)) and barrier certificates (Prajna et al. (2007)). While CLFs are used to asymptotically stabilize some desired set, CBFs are used to render safe sets controlled forward invariant, independent of the underlying control objective.

4.1 Control barrier functions

We consider affine control systems on the form

$$\dot{x} = f(x) + g(x)u, \tag{19}$$

with state $x \in \mathbb{R}^n$ and input $u \in U \subset \mathbb{R}^m$, where U is a convex set. The mappings $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to R^{n\times m}$ are assumed continuous. The following definition of CBFs is modified from (Ames et al., 2017, Definition 5):

Definition 2. Let $B : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function that defines the set

$$K := \{x \in \mathbb{R}^n : B(x) \leq 0\}. \tag{20}$$

B is a CBF for (19) if there exists $\alpha \in \mathcal{K}_e$ and a set X with $K \subset \mathrm{Int}X$, such that, $\forall x \in X$,

$$\inf_{u\in U} \left[L_f B(x) + L_g B(x) u\right] \leq -\alpha(B(x)). \tag{21}$$

Remark 3. Note that we, contrary to much existing literature, have defined CBFs to be *negative* on Int$K$, to highlight the strong connection with CLFs. ■

The following theorem states safety of (19) with inputs constrained to the admissible input set

$$U_B(x) := \{u \in U : L_f B(x) + L_g B(x) u \leq -\alpha(B(x))\}. \tag{22}$$

Theorem 4. If B is a CBF on $\mathbb{R}^n$ defining K, then K is forward invariant for the system

$$\dot{x} \in F_B(x) := \{f(x) + g(x)u : u \in U_B(x)\}. \tag{23}$$

Proof. $K \subset \mathrm{Int}X$ implies that X contains an open neighborhood of K. $\forall x \in X \setminus K$ and $\forall u \in U_B(x)$ we have

$$\dot{B} = L_f B(x) + L_g B(x) u \leq -\alpha(B(x)) < 0. \tag{24}$$

The proof follows from (Maghenem and Sanfelice, 2021, Theorem 1). □

In Theorem 4 we have made use of the fact that $u \in U_B$ implies that B is decreasing on a neighborhood *outside* the safe set. This allows us to omit the commonly assumed regularity properties in existing CBF literature: Lipschitz continuity of $f(x) + g(x)u$ and $\frac{\partial B}{\partial x} \neq 0, \forall x \in \partial K$. Clearly, Theorem 4 does not require $\alpha \in \mathcal{K}_e$: it suffices that $\alpha(\phi) \geq 0, \forall \phi \geq 0$. A salient feature of CBF design is the use of extended class-$\mathcal{K}$ functions in the construction of U_B. This ensures a smooth transition from the practically unconstrained dynamics on the interior of K, to the constrained dynamics when approaching the boundary of K. In addition, K is rendered locally attractive, which adds robustness.

Remark 5. In view of Theorem 4, we observe that B is not required to be continuously differentiable, or even defined, outside the set $X \supset K$. ■

4.2 Higher order control barrier functions

Higher order CBFs (HOCBFs) bear resemblance to backstepping of CLFs, and allow us to construct CBFs for systems where $L_g B(x) = 0, \forall x \in \mathbb{R}^n$. HOCBFs is a generalization of the exponential CBFs proposed in Nguyen et al. (2016). The following definition is modified from Xiao and Belta (2019):

Definition 6. For the system (19), let $B_1 : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function that defines the set

$$K_1 := \{x \in \mathbb{R}^n : B_1(x) \leq 0\}. \tag{25}$$

B_1 is a HOCBF candidate of order q if

$$L_g L_f^{q-i} B_1(x) = 0, \quad \forall x \in \mathbb{R}^n, \ \forall i \geq 2, \tag{26}$$

$$L_g L_f^{q-1} B_1(x) \neq 0, \quad \text{for some } x \in \mathbb{R}^n. \tag{27}$$

Given a HOCBF candidate, the set of admissible control inputs may be obtained by iteratively constructing new CBFs until the control input appears. Let B_i, for $i \in \{2, .., q\}$, be defined by

$$B_i(x) := L_f B_{i-1}(x) + \alpha_{i-1}(B_{i-1}(x)), \tag{28}$$

where $\alpha_{i-1} \in \mathcal{K}_e$ are sufficiently differentiable functions. From (26)-(27) it follows that $L_g B_q(x) = L_g L_f^{q-1} B_1(x) \neq 0$ for some $x \in \mathbb{R}^n$.

Definition 7. Let B_1 be a HOCBF candidate of order q that defines the set K_1. Let $B_i, i \in \{2, .., q\}$ be defined as in (28), and define the safe set

$$K := \bigcap_{i=1}^{q} K_i, \quad K_i := \{x \in \mathbb{R}^n : B_i(x) \leq 0\}. \tag{29}$$

B_1 is a HOCBF of order q if there exists $\alpha_q \in \mathcal{K}_e$ and a set X with $K \subset \text{Int}X$, such that, $\forall x \in X$,

$$\inf_{u \in U} \left[L_f B_q(x) + L_g B_q(x) u \right] \leq -\alpha_q(B(x)). \tag{30}$$

Note that we do not require $X \supset K_q$. The inequality (30) needs to hold only on a neighborhood of $K \subset K_q$. While we do not require $L_g B_q(x) \neq 0, \forall x \in \mathbb{R}^n$, sufficient control authority is required for there to exist a function α_q that satisfies (30). If the control authority vanishes at critical points, safety may be achieved using synergistic CBFs (Marley et al. (2021)). Similar to before, let B_q and α_q define the admissible input set

$$U_{B_q}(x) := \{u \in U : L_f B_q(x) + L_g B_q(x) u \leq -\alpha_q(B_q(x))\}. \tag{31}$$

The theorem below states safety, with respect to K_1, for solutions starting in $K \subset K_1$.

Theorem 8. If B_1 is a HOCBF on $\mathbb{R}^n$ defining K_1 then $K \subset K_1$ is forward invariant for the system

$$\dot{x} \in F_{B_q}(x) := \{f(x) + g(x)u : u \in U_{B_q}(x)\}. \tag{32}$$

Proof. Suppose, for the moment, that $X \supset K_q$. Then, by Theorem 4, K_q is forward invariant. $\forall x \in K_q \setminus K_{q-1}$,

$$\dot{B}_{q-1} = L_f B_{q-1}(x) \leq -\alpha_{q-1}(B_{q-1}(x)) < 0, \tag{33}$$

which shows forward invariance of $K_q \cap K_{q-1}$. By recursively applying similar arguments, K is forward invariant. Since solutions starting in K cannot leave K, it suffices that $X \supset K$. □

Remark 9. Theorem 8 is similar to (Xiao and Belta, 2019, Theorem 5), but omits the requirement of Lipschitz continuity of $f(x) + g(x)u$. A consequence of omitting the Lipschitz requirement is that X must contain a neighborhood of K. See (Maghenem and Sanfelice (2021)) for an in-depth theoretical presentation of barrier functions. ■

4.3 Safety-critical controller

Given a nominal control law $\kappa : \mathbb{R}^n \to U$, and a CBF defining $U_B : \mathbb{R}^n \rightrightarrows U$, an optimal safety-critical controller, in the least-square sense, is obtained by

$$\kappa_B(x) := \arg \min_{u \in U_B(x)} (u - \kappa(x))^\top P (u - \kappa(x)), \tag{34}$$

with positive definite cost matrix $P \in \mathbb{R}^{m \times m}$. If $U = \mathbb{R}^m$ and P is diagonal, the closed-form solution

$$\begin{aligned} \kappa_B(x) &= \begin{cases} \kappa(x), & \kappa(x) \in U_B(x) \\ \kappa(x) - \dfrac{ab^\top}{bb^\top} & \kappa(x) \notin U_B(x) \end{cases} \\ a &:= L_f B(x) + L_g B(x) \kappa(x) + \alpha(B(x)), \\ b &:= L_g B(x) P^{-0.5}, \end{aligned} \tag{35}$$

follows from the KKT-conditions (see e.g. Nocedal and Wright (2006)), where $P^{-0.5}$ is the inverse of the principal square root of P. Xu et al. (2015) showed that κ_B is locally Lipschitz, provided that f, g and κ are locally Lipschitz.

4.4 Recurring example

Selecting linear acceleration and heading rate as control inputs transforms (11) into an affine system,

$$\dot{p}_d = v_d z_d, \quad \dot{v}_d = u_{d_1}, \quad \dot{z}_d = u_{d_2} S z_d, \tag{36}$$

with input $u_d = (u_{d_1}, u_{d_2}) \in \mathbb{R}^2$. The function

$$B_1(p_d) := r_o - |p_e|, \quad p_e := p_d - p_o, \tag{37}$$

defines a safe set $K_s = \{p_d \in \mathbb{R}^2 : |p_e| \geq r_o\}$, relative to a circular obstacle domain centered at $p_o \in \mathbb{R}^2$ with radius r_o. While the arguably simpler quadratic function $B_1(p_d) = r_o^2 - p_e^\top p_e$ defines the same safe set, the formulation in (37) is preferred since the magnitude of the gradient of B_1 on a neighborhood of ∂K_s becomes independent of obstacle radius r_o.

The time derivative of B_1 is given by

$$\dot{B}_1 = \frac{\partial B_1}{\partial p_d} \dot{p}_d = -\frac{p_e^\top}{|p_e|} v_d z_d. \tag{38}$$

From this we define

$$B_2(p_d, v_d, z_d) := -\frac{p_e^\top}{|p_e|} v_d z_d + \alpha_1(B_1(p_d)). \tag{39}$$

Selecting $\alpha_1(\phi) = \phi/t_1$, with time constant $t_1 \in \mathbb{R}_{>0}$, yields $B_2 = \dot{B}_1 + B_1/t_1$. This is equivalent to the commonly used CBF formulation for obstacle avoidance (see e.g. Borrmann et al. (2015)). To better shape the gradient of B_2 with respect to p_d, we select the saturating function

$$\alpha_1(\phi) := v_r \arctan \frac{\phi}{\delta}, \tag{40}$$

where $\delta \in \mathbb{R}_{>0}$ regulates the slope. The multiplying term v_r is included to ensure that $v_d = v_r$ implies $B_2 < 0$ for sufficiently large $|p_e|$, and any orientation z_d. The key idea behind α_1 in (40) is that the gradient,

$$\frac{\partial \alpha_1}{\partial \phi}(\phi) = v_r \frac{\delta}{\delta^2 + \phi^2} \tag{41}$$

attains its maximum at $\phi = 0$, and decays to zero as $\phi \to \pm\infty$. We select a time constant $t_2 \in \mathbb{R}_{>0}$ to define

$$U_{B_2}(p_d, v_d, z_d) := \left\{ u_d \in \mathbb{R}^2 : \dot{B}_2 \leq \frac{-B_2}{t_2} \right\}, \tag{42}$$

where

$$\dot{B}_2 = -\frac{v_d^2}{|p_e|} \left(\frac{p_e^\top}{|p_e|} S^\top z_d \right)^2 - \frac{v_r \delta}{\delta^2 + B_1(p_d)^2} \frac{p_e^\top}{|p_e|} v_d z_d - \frac{p_e^\top}{|p_e|} \begin{bmatrix} z_d & v_d S z_d \end{bmatrix} u_d. \tag{43}$$

$U_{B_2} : \mathbb{R}^2 \times \mathbb{R} \times \mathcal{S}^1 \rightrightarrows \mathbb{R}^2$ is well-defined on a neighborhood of $K_s \times \mathbb{R} \times \mathcal{S}^1$. Then any $u_d \in U_{B_2}$ ensures that p_d does not enter the obstacle domain, provided that obstacles are detected sufficiently early. The slope parameter δ and time constant t_2 are selected depending on the dynamic capabilities of the ship to be guided, ensuring feasibility of the desired trajectory during evasive maneuvers.

5. REACTIVE GUIDANCE CONTROL DESIGN

We now deduce a safe maneuvering guidance design for the general system before returning to the ship case. Let y_d be the output of a second-order affine control system with desired dynamics:

$$\dot{x}_d = f_d(x_d) + g_d(x_d) u_d, \quad y_d = h_d(x_d), \tag{44}$$

with state $x_d \in \mathbb{R}^{n_d}$, input $u_d \in \mathbb{R}^{m_d}$ and output map

$$h_d(x_d) := \begin{bmatrix} I_{p\times p} & 0_{p\times(n_d-p)} \end{bmatrix} x_d, \tag{45}$$

such that y_d is the first p elements of x_d, i.e. $y_d = h_d(x_d) = [x_{d_1}, ..., x_{d_p}]^\top$. This enables the feedback term from vehicle to the guidance system, introduced in Section 5.1.

Let $\kappa_d : \mathbb{R} \times \mathbb{R}^{n_d} \to \mathbb{R}^{m_d}$ be a nominal controller that solves the maneuvering problem for the closed-loop system

$$\dot{x}_d = f_d(x_d) + g_d(x_d)\kappa_d(s, x_d), \quad \dot{s} = f_s(s, x_d). \tag{46}$$

Assumption 10. The set

$$\mathcal{A}_r := \{s, x_d \in \mathbb{R} \times \mathbb{R}^{n_d} : h_d(x_d) = y_r(s)\} \tag{47}$$

is UGAS for the system (46).

5.1 Directional gradient feedback

We augment the system (44) with a feedback term that enables us to move the desired point y_d along a trajectory parallel to $y_r(s)$, to a position that minimizes the distance $|y_d - y|$. (Or, more precise, to a point that minimizes a CLF that is positive definite with respect to $|y_d - y|$). This is achieved by

$$\dot{x}_d = f_d(x_d) + g_d(x_d)u_d + \gamma(s)w, \ \gamma(s) := \begin{bmatrix} \tau(s) \\ 0_{n_d-p} \end{bmatrix}, \tag{48}$$

where $w \in \mathbb{R}$ is an additional feedback speed in the direction of the path tangent τ. If Assumption 10 holds, the geometric task $|y_d(t) - y_r(s(t))| \to 0$ is satisfied for the system

$$\dot{x}_d = f_d(x_d) + g_d(x_d)\kappa_d(s, x_d) + \gamma(s)w, \tag{49}$$
$$\dot{s} = f_s(s, x_d) + w, \tag{50}$$

where w has been added to the update law for s. Path-following for y is then solved by driving $|y(t) - y_d(t)| \to 0$.

Let $V_d : \mathbb{R}^{n_d} \times \mathbb{R}^n \to \mathbb{R}$ be a CLF relative to the set

$$\mathcal{A}_d := \{x_d, x \in \mathbb{R}^{n_d} \times \mathbb{R}^n : h_d(x_d) = h(x)\}. \tag{51}$$

A *directional* gradient feedback from y to y_d is obtained by the assignment $w = -\omega_w(s, x_d, x)$, where

$$\omega_w(s, x_d, x) := \mu_w \frac{\partial V_d}{\partial x_d}(x_d, x)\gamma(s), \tag{52}$$

with $\mu_w \in \mathbb{R}_{\geq 0}$. Recall that we have selected (44) to be a second-order system, which implies $L_{g_d} h_d(x_d) u_d = 0$. With the direct assignment $w = -\omega_w(s, x_d, x)$ the dynamics of y_d becomes

$$\dot{y}_d = L_{f_d} h_d(x_d) - \tau(s)\omega_w(s, x_d, x). \tag{53}$$

We recognize this structure from the dynamics of y_r given in (10), with the key difference that y_d is not constrained to the path Y_r. However, to accomodate safety constraints using CBFs, we require that $\dot{w}$ exists and is available. Accordingly, we propose a filtered directional gradient feedback law,

$$\dot{w} = -\lambda_w(w - \omega_w(s, x_d, x)), \tag{54}$$

with $\lambda_w \in \mathbb{R}_{>0}$. To show stability we adopt the strategy from (Skjetne et al., 2004, Theorem 3.2). The time derivative of the augmented Lyapunov function

$$V_{d,2}(x_d, x, w) := V_d(x_d, x) + \frac{1}{2\lambda_w \mu_w} w^2, \tag{55}$$

is given by

$$\dot{V}_{d,2} = \frac{\partial V_{d,2}}{\partial x_d}\dot{x}_d + \frac{\partial V_{d,2}}{\partial x}\dot{x} + \frac{\partial V_{d,2}}{\partial w}\dot{w}. \tag{56}$$

Restricting our attention to the feedback term $\gamma(s)w$ in the dynamics of x_d, and inserting the update law in (54) for $\dot{w}$, we obtain

$$\begin{aligned} &\frac{\partial V_{d,2}}{\partial x_d}\gamma(s)w + \frac{\partial V_{d,2}}{\partial w}(-\lambda_w(w - \omega_w(s, x_d, x)) \\ =&w\left(\frac{\partial V_{d,2}}{\partial x_d}\gamma(s) - \frac{\lambda_w}{\lambda_w \mu_w}\left(w + \mu_w \frac{\partial V_{d,2}}{\partial x_d}\gamma(s)\right)\right) \\ =&-\frac{1}{\mu_w} w^2 \leq 0. \end{aligned} \tag{57}$$

5.2 Introducing safety guarantees

Let $\xi := (s, w, x_d) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^{n_d} =: \Xi$ collect the states that constitute the guidance system. Defining

$$\bar{f}(\xi) := \begin{bmatrix} f_s(s,x_d)+w \\ -\lambda_w w \\ f_d(x_d)+\gamma(s)w \end{bmatrix},\ \bar{g}(\xi) := \begin{bmatrix} 0_{1\times m_d} \\ 0_{1\times m_d} \\ g_d(x_d) \end{bmatrix} \tag{58}$$

$$\bar{\gamma}(\xi,x) := \begin{bmatrix} 0 \\ \lambda_w \omega_w(s,x_d,x) \\ 0_{n_d} \end{bmatrix} \tag{59}$$

we obtain an affine control system

$$\dot{\xi} = \bar{f}(\xi) + \bar{g}(\xi)u_d + \bar{\gamma}(\xi,x), \tag{60}$$

with control input u_d and disturbance x.

Given a safe operating region in the output space, $K_s \subset \mathbb{R}^p$, let $\bar{B}_1 : \Xi \to \mathbb{R}$ satisfy

$$\begin{aligned} \bar{B}_1(\xi) &\le 0, \quad \forall \bar{h}(\xi) \in K_s, \\ \bar{B}_1(\xi) &> 0, \quad \forall \bar{h}(\xi) \in \mathbb{R}^p \setminus K_s, \end{aligned} \tag{61}$$

where $\bar{h} : \Xi \to \mathbb{R}^p$ is the output map satisfying $y_d = \bar{h}(\xi)$. Then $\bar{B}_1$ defines a safe set $\bar{K}_1 = \mathbb{R}\times\mathbb{R}\times K_s \times \mathbb{R}^{n_d-p} \subset \Xi$. By construction, $L_{\bar{g}}\bar{B}_2(\xi) = 0$ and $\frac{\partial \bar{B}_1}{\partial \xi}(\xi)\bar{\gamma}(\xi,x) = 0$, which implies $\dot{\bar{B}}_1 = L_{\bar{f}}\bar{B}_1$. This allows us to construct a CBF

$$\bar{B}_2(\xi) := L_{\bar{f}}\bar{B}_1(\xi) + \bar{\alpha}_1(\bar{B}_1(\xi)), \tag{62}$$

with $\bar{\alpha}_1 \in \mathcal{K}_e$. In the next step we must account for the perturbation term $\gamma(s)w$. Differentiating $\bar{B}_2$ along the solutions of (60), we obtain the admissible input set

$$\begin{aligned} \bar{U}_{B_2}(\xi,x) := \{u_d \in \mathbb{R}^{m_d} : L_{\bar{f}}\bar{B}_2(\xi) + L_{\bar{g}}\bar{B}_2(\xi)u_d \\ \frac{\partial \bar{B}_2}{\partial \xi}(\xi)\bar{\gamma}(\xi,x) \le -\bar{\alpha}_2(\bar{B}_2(\xi))\}, \end{aligned} \tag{63}$$

for some suitable choice of $\bar{\alpha}_2 \in \mathcal{K}_e$. Applying the safety-critical controller

$$\begin{aligned} &\bar{\kappa}_{B_2}(\xi,x) := \\ &\arg\min_{u_d \in \bar{U}_{B_2}(\xi,x)} (u_d - \kappa_d(s,x_d))^\top P (u_d - \kappa_d(s,x_d)), \end{aligned} \tag{64}$$

to the open-loop system (60), we arrive at the reactive guidance system

$$\dot{\xi} = \bar{f}(\xi) + \bar{g}(\xi)\bar{\kappa}_{B_2}(\xi,x) + \bar{\gamma}(\xi,x). \tag{65}$$

The system (65) outputs a safe trajectory $y_d(t) = \bar{h}(\xi(t)) \in K_s$, provided that solutions start in $\bar{K} := \bar{K}_1 \times \bar{K}_2$, where $\bar{K}_2$ is the safe set defined by $\bar{B}_2$. Moreover, x_d serves as a continuous reference signal for the desired dynamics of the vehicle.

$\bar{\kappa}_{B_2} : \Xi \times \mathbb{R}^{n_d} \to \mathbb{R}^{m_d}$ is explicitly dependent on the vehicle states x through the update law for w. Equally important, the instantaneous value of w shrinks or enlarges the projection of $\bar{K}_2$ onto $\mathbb{R}^{n_d}$. As a result, during evasive manuevers, the feedforward states x_d implicitly depend on the vehicle state x. This fact influences the tuning of the gains μ_w and λ_w. In Skjetne et al. (2004) it is shown that a separation of time scales is obtained by selecting sufficiently large gradient feedback gains, such that the feedback dynamics of the guidance system are significantly faster than the dynamics of the vehicle. For the reactive guidance design proposed herein, best performance is obtained by selecting comparatively lower gradient feedback gains.

Remark 11. The virtual control input $u_d = \kappa_{B_2}$ may also be used in the lower-level vehicle controller, to obtain a (possibly discontinuous) reference signal for $\ddot{y}$. Reference signals for higher-order derivatives are also possible, by increasing the relative degree of virtual vehicle represented by the system (44). Safety-constraints are then implemented by using a HOCBF of required order. In this case, the order of the dynamics of s and w must also be increased, to ensure that the resulting admissible input set is well-defined. ■

5.3 Recurring example

Feedback from p to p_d is achieved by the system

$$\begin{aligned} \dot{s} &= f_s(s,p_d) + w, \\ \dot{w} &= -\lambda_w(w + \mu_w(p-p_d)^\top \tau(s)), \\ \dot{p}_d &= v_d z_d + w\tau(s),\ \dot{v}_d = u_{d_1},\ \dot{z}_d = u_{d_2} S z_d, \end{aligned} \tag{66}$$

with $f_s(s,p_d)$ defined in (15). The update law for w was designed using the CLF $V = (p-p_d)^\top(p-p_d)$.

A CBF for the system (66) is constructed by differentiating B_1 defined in (37), along the solutions of (66), resulting in

$$\bar{B}_2(\xi) := -\frac{p_e^\top}{|p_e|}(v_d z_d + w\tau(s)) + \alpha_1(B_1(p_d)), \tag{67}$$

where $\xi := (s, w, p_d, v_d, z_d)$ collects the states that constitute the guidance system. The set $\bar{U}_{B_2}$ is obtained similar to (42), with $\dot{B}_2$ replaced with

$$\begin{aligned} \dot{\bar{B}}_2 := &-\frac{(v_d+w)^2}{|p_e|}\left(\frac{p_e^\top}{|p_e|}S^\top \frac{v_d z_d + w\tau(s)}{|v_d z_d + w\tau(s)|}\right)^2 \\ &- \frac{v_r \delta}{\delta^2 + B_1(p_d)^2}\frac{p_e^\top}{|p_e|}(v_d z_d + \tau(s)w) \\ &- \frac{p_e^\top}{|p_e|}\left(\tau(s)\dot{w} + \frac{\partial \tau(s)}{\partial s}\dot{s}w\right) - \frac{p_e^\top}{|p_e|}\begin{bmatrix} z_d & v_d S z_d \end{bmatrix} u. \end{aligned} \tag{68}$$

Before continuing, we design a nominal controller $\kappa_d := (\kappa_{d_1}, \kappa_{d_2})$ for (u_{d_1}, u_{d_2}) that asymptotically stabilizes $|p_d - p_r(s)| = 0$. To drive $v_d \to v_r$ we simply select

$$u_{d_1} = \kappa_{d_1}(v_d, v_r) := -k_1(v_d - v_r), \quad k_1 \in \mathbb{R}_{>0}. \tag{69}$$

To steer the vehicle towards the path, we use u_{d_2} to drive z_d to $\kappa_z(s,p_d)$ in (13). This is equivalent to driving $\tilde{z} := R(\kappa_z(s,p_d))^\top z_d \to \varepsilon_1$. We select the control law

$$u_{d_2} = \kappa_{d_2} := -k_2 \frac{\tilde{z}_2}{\sqrt{1-\lambda^2\tilde{z}_1^2}} + \omega_\Delta + \omega_r, \tag{70}$$

with gain $k_2 \in \mathbb{R}_{>0}$ and regularization parameter $\lambda \in (0,1)$. The first term is the non-hybrid feedback controller proposed in Marley et al. (2020), while

$$\omega_\Delta := -\frac{v_d e_2 \Delta}{(\Delta^2 + e_2^2)^{3/2}} \tag{71}$$

is the rotational velocity of z_Δ, and

$$\omega_r := -\tau(s)^\top S^\top \frac{\partial \tau(s)}{\partial s} v_r \tag{72}$$

is the feedforward rotational velocity of the path.

Applying the safety-critical controller

$$\kappa_B(\xi,p) := \arg\min_{u_d \in \bar{U}_{B_2}(\xi,p)} (u_d - \kappa_d)^\top P (u_d - \kappa_d) \tag{73}$$

to the system (66) finalizes the design. Selecting P as $P := \mathrm{diag}(P_1, P_2)$, with $P_1 \gg P_2 > 0$, will result in turning as the preferred evasive maneuver.

Remark 12. When approaching an obstacle centered on the path, p_d may slow down significantly, or even get stuck in an equilibrium position in front of the obstacle. While increasing P_1 mitigates this issue, too large P_1 will result in chattering, especially for noisy measurements of obstacle position. A preferred solution is to use the hybrid CBF formulation presented in (Marley et al. (2021)), where only heading rate is used as the evasive control input. ■

6. NUMERICAL SIMULATIONS

A simulation study is performed with the dynamics of p emulating an underactuated ship of length $L = 100$m, moving in an unknown current with speed 1 m/s towards North.

6.1 Controller for the numerical ship model

The numerical model of the ship has independent actuation in surge and yaw. The speed controller uses proportional feedback, combined with reference feedforward, to drive the surge speed to v_d. The heading controller uses the integral Line-Of-Sight (iLOS) algorithm of Børhaug et al. (2008) to account for the unknown sideslip. Proportional-derivative feedback, combined with reference feedforward, is used to drive the yaw angle towards the desired heading given by the iLOS algorithm. The heading rate of the virtual vessel, obtained from the safety-critical controller (73), is included in the feedforward heading rate.

The iLOS algorithm achieves zero cross-track error under the assumption of constant sideslip. Due to the combination of current and non-constant path curvature, this assumption does not hold for the simulations presented herein. The sideslip further implies that the total ship speed $|\nu|$ differs from v_d, resulting in a non-zero along-track error of p relative to p_d. Moreover, the feedforward terms use speed-over-ground ν as input, as opposed to velocity relative to current, resulting in yet another source of tracking error.

6.2 Parameters for guidance system

The lookahead distance of the ship is $2L$, while the lookahead distance for the virtual vessel is set to $\Delta = 10L$. This achieves reasonably fast convergence of p to p_d, and moderate convergence of p_d to $p_r(s)$. The remaining parameters for the guidance system are: gradient gains $\mu_s = 5\text{s}^{-1}$, $\mu_w = 0.05\text{s}^{-1}$ and $\lambda_w = 1\text{s}^{-1}$; nominal controller gains $k_1 = 0.1\text{s}^{-1}$, $k_2 = \pi/180$rad/s and $\lambda = 0.95$; CBF parameters $\delta = 100$s, $t_2 = 50$s, and cost matrix $P = \text{diag}(10, 1)$. The reference path is a circular path $p_r(s) = 10000\left[(\sin s - \sin \pi/3)\ (-\cos s + \cos \pi/3)\right]^\top$ m, with speed assignment $v_r = 5$m/s. A static obstacle with radius of $r_o := 200$m is centered slightly North of the path at $p_o := (1150, 2800)$m.

For comparison purposes, simulations with the commonly used CBF formulation obtained by replacing α_1 in (40) with $\alpha_1(\phi) := \phi/t_1$ are also presented. A time constant of $t_1 = 20s$ was used, to obtain a comparable evasive maneuver initiated at approximately the same distance from the obstacle.

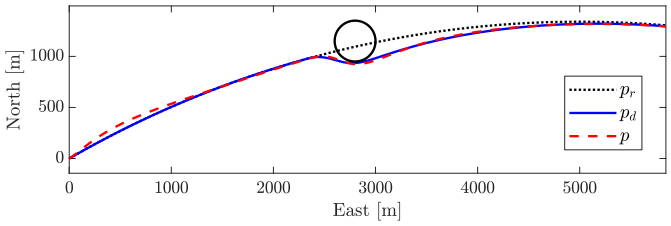

Fig. 1. Trajectory of p_r, p_d and p. All trajectories start at the origin. Obstacle domain shown in solid black.

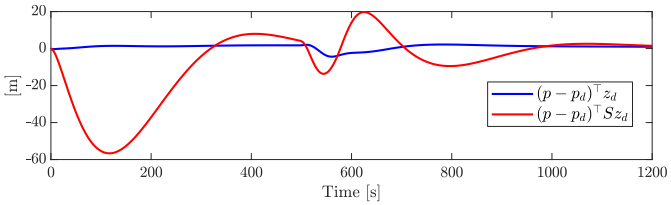

Fig. 2. Along-track error (blue) and cross-track error (red) of p relative to p_d, along the direction z_d.

6.3 Simulation results

Simulations are initialized with p and p_d on the path, moving at the prescribed velocity, but with the integral state of the ship iLOS algorithm set to zero. The initial states of the guidance system are $s_0 = \pi/3$, $p_0 = p_{d,0} = p_r(s_0)$, $z_0 = z_{d,0} = \tau(s_0)$, $v_{d,0} = v_r$ and $w_0 = 0$. The resulting trajectories are shown in Fig. 1. As can be observed, p_d traces out a safe trajectory, deviating from the path when approaching the obstacle, and converging back towards the path when safety allows it.

The tracking error of the ship is shown in Fig. 1. Initially, the ship deviates from the desired trajectory due to the unknown current, before the integral action steers the ship back towards p_d. The tracking error increases during the evasive maneuver, due to sideslip induced by turning. However, no integral wind-up is observed, illustrating the advantage of implementing safety constraints on the guidance level. Increased feasibility of the trajectory may be achieved by augmenting the virtual vessel with sway dynamics, to emulate the sideslip experienced by ships during turning.

A closer view of the desired trajectory during the evasive manuever is shown in Fig. 3, with corresponding heading rate of the virtual vessel presented in Fig. 4. The trajectory and heading rate resulting from the alternative CBF formulation $B_2 = \dot{B}_1 + B_1/t_1$ is included for comparison. Selecting α_1 as the saturating function in (40) results in favorable behavior: the evasive maneuver is less aggressive, with reduced deviation of the desired trajectory from the reference path.

7. CONCLUSION

This paper proposed a reactive guidance design for path following, that facilitates safety of autonomous vehicles, using control barrier functions to enforce safety of the desired trajectory. Feedback from vehicle to the guidance system was maintained by replacing the previously used gradient feedback term with a *directional* gradient feedback term, in the direction of the reference path. The

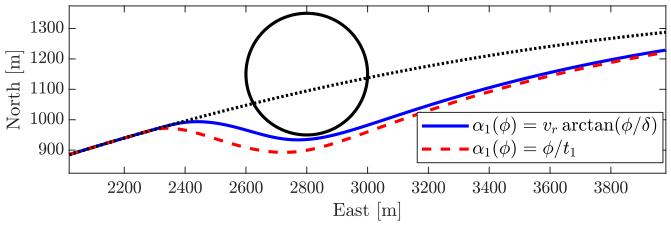

Fig. 3. Trajectory of the virtual vessel for the two choices of α_1 in the definition of the CBF $\bar{B}_2$. Trajectories start in the bottom left corner. Dashed black line shows reference path. Obstacle domain shown in solid black.

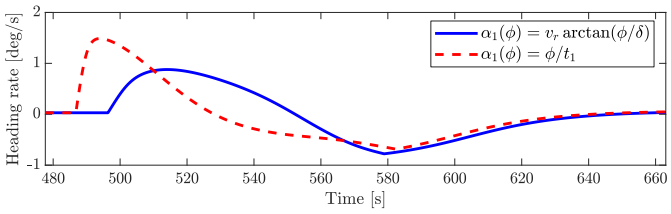

Fig. 4. Heading rate of the virtual vessel for the two choices of α_1 in the definition of the CBF $\bar{B}_2$.

design process was illustrated using a unicycle model as a virtual vessel to guide an underactuated ship.

An alternative CBF formulation for planar vehicles was proposed, using a saturating function to shape the gradient of the CBF with respect to vehicle position. This result was enabled by the theory of higher-order CBFs (HOCBFs), which yields a constructive way to design CBFs for control systems of relative degree 2 or higher.

REFERENCES

Aguiar, A.P., Dačić, D.B., Hespanha, J.P., and Kokotović, P. (2004). Path-following or reference tracking? *IFAC Proceedings Volumes*, 37(8), 167–172.

Ames, A.D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., and Tabuada, P. (2019). Control barrier functions: Theory and applications. In *Proc. European Control Conference*, 3420–3431. Napoli, Italy.

Ames, A.D., Xu, X., Grizzle, J.W., and Tabuada, P. (2017). Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8), 3861–3876.

Artstein, Z. (1983). Stabilization with relaxed controls. *Nonlinear Analysis*, 7(11), 1163–1173.

Basso, E.A., Thyri, E.H., Pettersen, K.Y., Breivik, M., and Skjetne, R. (2020). Safety-critical control of autonomous surface vehicles in the presence of ocean currents. In *Proc. IEEE Conf. Control Technology & Applications*. Montreal, Canada.

Børhaug, E., Pavlov, A., and Pettersen, K.Y. (2008). Integral LOS control for path following of underactuated marine surface vessels in the presence of constant ocean currents. In *Proc. IEEE Conf. Decision & Control*, 4984–4991.

Borrmann, U., Wang, L., Ames, A.D., and Egerstedt, M. (2015). Control barrier certificates for safe swarm behavior. *IFAC-PapersOnLine*, 48(27), 68–73.

Emam, Y., Glotfelter, P., and Egerstedt, M. (2019). Robust barrier functions for a fully autonomous, remotely accessible swarm-robotics testbed. In *Proc. IEEE Conf. Decision & Control*, 3984–3990. Nice, France.

Fossen, T.I. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons.

Isaly, A., Patil, O.S., Sanfelice, R.G., and Dixon, W.E. (2021). Adaptive safety with multiple barrier functions using integral concurrent learning. In *Proc. American Control Conf.*, 3710–3715. New Orleans, LA, USA.

Maghenem, M. and Sanfelice, R.G. (2021). Sufficient conditions for forward invariance and contractivity in hybrid inclusions using barrier functions. *Automatica*, 124, 109328.

Marley, M., Skjetne, R., and Teel, A.R. (2020). A kinematic hybrid feedback controller on the unit circle suitable for orientation control of ships. In *Proc. IEEE Conf. Decision & Control*, 1523–1529. Jeju Island, Republic of Korea.

Marley, M., Skjetne, R., and Teel, A.R. (2021). Synergistic control barrier functions with application to obstacle avoidance for nonholonomic vehicles. In *Proc. American Control Conf.*, 242–248. New Orleans, LA, USA.

Nguyen, Q., Hereid, A., Grizzle, J.W., Ames, A.D., and Sreenath, K. (2016). 3D dynamic walking on stepping stones with control barrier functions. In *Proc. IEEE Conf. Decision & Control*, 827–834. IEEE, Las Vegas, NV, USA.

Nocedal, J. and Wright, S.J. (2006). *Numerical Optimization*. Springer.

Prajna, S., Jadbabaie, A., and Pappas, G.J. (2007). A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8), 1415–1428.

Skjetne, R. (2005). *The Maneuvering Problem*. Phd thesis, Norwegian University of Science and Technology.

Skjetne, R., Fossen, T.I., and Kokotović, P.V. (2004). Robust output maneuvering for a class of nonlinear systems. *Automatica*, 40(3), 373–383.

Skjetne, R., Fossen, T.I., and Kokotović, P.V. (2005). Adaptive maneuvering, with experiments, for a model ship in a marine control laboratory. *Automatica*, 41(2), 289–298.

Skjetne, R., Jørgensen, U., and Teel, A.R. (2011). Line-of-sight path-following along regularly parametrized curves solved as a generic maneuvering problem. In *Proc. IEEE Conf. Decision & Control*, 2467–2474. Orlando, FL, USA.

Taylor, A.J. and Ames, A.D. (2020). Adaptive safety with control barrier functions. In *Proc. American Control Conf.*, 1399–1405. Denver, CO, USA.

Thyri, E.H., Basso, E.A., Breivik, M., Pettersen, K.Y., Skjetne, R., and Lekkas, A.M. (2020). Reactive collision avoidance for ASVs based on control barrier functions. In *Proc. IEEE Conf. Control Technology & Applications*. Montreal, Canada.

Wieland, P. and Allgöwer, F. (2007). Constructive safety using control barrier functions. *IFAC Proceedings Volumes*, 40(12), 462–467.

Xiao, W. and Belta, C. (2019). Control barrier functions for systems with high relative degree. In *Proc. IEEE Conf. Decision & Control*, 474–479. IEEE, Nice, France.

Xu, X., Tabuada, P., Grizzle, J.W., and Ames, A.D. (2015). Robustness of control barrier functions for safety critical control. *IFAC-PapersOnLine*, 48(27), 54–61.