

# Invariant Sets for Integrators and Quadrotor Obstacle Avoidance

Ludvig Doerer,<sup>1</sup> Petter Nilsson,<sup>2</sup> Aaron D. Ames,<sup>2</sup> and Richard M. Murray<sup>3</sup>

**Abstract**—Ensuring safety through set invariance has proven a useful method in a variety of applications in robotics and control. However, finding analytical expressions for maximal invariant sets, so as to maximize the operational freedom of the system without compromising safety, is notoriously difficult for high-dimensional systems with input constraints. Here we present a generic method for characterizing invariant sets of  $n$ th-order integrator systems, based on analyzing roots of univariate polynomials. Additionally, we obtain analytical expressions for the orders  $n \leq 4$ . Using differential flatness we subsequently leverage the results for the  $n = 4$  case to the problem of obstacle avoidance for quadrotor UAVs. The resulting controller has a light computational footprint that showcases the power of finding analytical expressions for control-invariant sets.

## I. INTRODUCTION

The last few years have seen a concerted research effort in autonomous systems, with the goal of integrating these systems into everyday life. As a result, safety—guarantees that the system always satisfies certain state constraints—is of critical importance in the design of such systems. The notion of set invariance has emerged as a theoretical foundation for safety of dynamical systems. Finding a control-invariant set  $C$  so that  $C \subset E$ , where  $E$  is the set containing all safe states, means that safety can be assured by constraining the system to operate within  $C$ . In order to maximize the operational freedom of the system it is desirable that  $C$  is as large as possible, so as to not be overly conservative. So far, however, there is no cheap-to-evaluate way of finding maximal control-invariant sets, which is especially difficult for high-dimensional systems subject to input constraints.

Though the problem of finding maximal control-invariant sets for arbitrary systems is intractable, there is an extensive literature on computing numerical approximations—previous work can broadly be classified by the way sets are represented. Early work resulted in iterative algorithms for discrete-time linear systems that yield polyhedral approximations [1]; more recently “one-shot” methods have been proposed for polyhedra [2], [3], [4]. Other choices for set descriptions include semi-algebraic sets [5] via polynomial programming, the special case of ellipsoids via linear matrix inequalities [6], and level set methods that rely on numerical solutions to PDEs [7]. At the same time, invariance-driven approaches have also been considered in a variety of applications with safety-critical state constraints, including pedestrian avoidance [8],

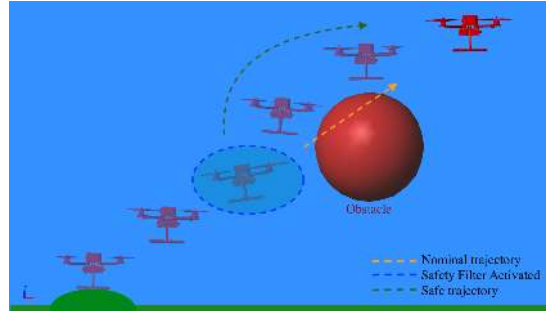


Fig. 1. An obstacle-avoiding quadrotor via invariance control.

stability of a two-wheeled inverted pendulum [9], bipedal locomotion [10], obstacle-avoiding quadrotors [11], and others [12], [13]. With the exception of [8], these earlier efforts either rely on computationally demanding methods for computing approximate invariant sets, or do not take input bounds into consideration in the construction of invariant sets. For systems where it is possible to analytically derive control-invariant sets this is desirable since it yields an exact answer at a smaller computational cost.

This work aims to analytically characterize invariant sets for a specific class of dynamical systems: the  $n$ th-order integrator  $\frac{d^n x}{dt^n} = u$  for  $n \geq 1$ . Understanding the nature of invariant sets for these systems is an interesting theoretical problem in its own right—as far as we know no explicit characterizations are known for  $n > 2$ . Understanding this problem may provide insight into phenomena that occur also in other linear or nonlinear problems.

The importance of enforcing invariance in  $n$ th-order integrators extends beyond intellectual curiosity. Firstly, many systems of interest follow or approximate integrator dynamics—the  $n = 2$  system was for example used to ensure safety of a lane keeping system in [12]. Secondly, the ideas can also be applied to high-dimensional non-linear differentially flat systems such as the standard model for multirotor UAVs [14], since integrator dynamics govern the evolution of the flat output space. In particular, in this paper we leverage invariance of the  $n = 4$  integrator to achieve obstacle avoidance for the 12D nonlinear quadrotor model, as shown in Fig. 1. Quadrotor obstacle avoidance via set invariance has previously been considered in [11] and [15], but without explicitly accounting for actuator limitations. While our flatness-based approach does not directly account for that either, it does provide a bound on the snap (fourth derivative) of the resulting motion, and minimizing snap is a common method to design smooth trackable trajectories [16].

<sup>1</sup>Royal Institute of Technology, Stockholm, Sweden. doerer@kth.se.

<sup>2</sup>Dept. of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA, 91125 USA. {pettni, ames}@caltech.edu.

<sup>3</sup>Dept. of Computing + Mathematical Sciences, California Institute of Technology, Pasadena, CA, 91125 USA. murray@cds.caltech.edu.

## A. Organization & Notation

After preliminary definitions and notation Section II contains a statement of the problem we seek to solve. In Section III a general method for characterizing control-invariant sets in the presence of individual state constraints is presented, including explicit characterizations for the  $n \leq 4$  cases. In the subsequent Section IV we provide results that enable multiple state constraints to be enforced simultaneously. The results are then applied to quadrotor obstacle avoidance in Sections V and VI. Lastly, Section VII concludes this work.

*Notation:* For a dynamical system

$$\frac{dx(t)}{dt} = f(x(t), u(t)), \quad u(t) \in \mathcal{U} \subset \mathbb{R}^m, \quad (1)$$

with  $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$  we define the *flow operator*  $\phi(t; x, u) : \mathbb{R}^n \rightarrow \mathbb{R}^n$  that for a given initial condition  $x$  and input signal  $u(t)$  represents the trajectory of the associated initial value problem. In particular,

$$\begin{aligned} \frac{d}{dt} \phi(t; x, u) &= f(\phi(t; x, u), u(t)), \\ \phi(0; x, u) &= x. \end{aligned} \quad (2)$$

We write  $[\phi(t; x, u)]_k$  to denote the  $k$ :th coordinate of the  $n$ -dimensional trajectory  $\phi(t; x, u)$ , and for an  $n$ -dimensional vector  $x$  we write  $x_{k:l}$  as an abbreviation of the coordinates  $x_k, \dots, x_l$ .

The central objects of study in this paper are control-invariant sets (sometimes also referred to as *viable sets* [17]), which are defined as follows (e.g. [18]).

**Definition 1.** A set  $C \subset \mathbb{R}^n$  is said to be *control-invariant* for a dynamical system if for any  $x \in C$  there exists an input signal  $u(t)$  such that

$$\phi(t; x, u) \in C \quad \forall t \geq 0. \quad (3)$$

For a set  $E$  of state constraints, we say that  $C$  is the *maximal control-invariant set* contained in  $E$ , if  $C$  is control-invariant and contains all other control-invariant sets that are subsets of  $E$ . This is well-defined since the union of control-invariant sets is also control-invariant.

A common way of representing control-invariant sets is as *super-level sets* of a function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$ ; for such a function we denote the corresponding set  $C^h = \{x \in \mathbb{R}^n : h(x) \geq 0\}$ .

## II. PROBLEM FORMULATION

As discussed above, the problem of finding maximal control-invariant sets for a general dynamical system (1) is intractable. To the best of our knowledge, expressions are unknown even for the simple third- and fourth-order integrator systems. In the following, we therefore restrict attention to a particular class of systems: the single input  $n$ th-order integrator system on the form

$$\begin{cases} \dot{x}_1 = u \\ \dot{x}_2 = x_1 \\ \vdots \\ \dot{x}_n = x_{n-1}, \end{cases} \quad (4)$$

that is subject to state and input constraints

$$E_{1:n} = \{x_{1:n} \in \mathbb{R}^n : \underline{x}_i \leq x_i \leq \bar{x}_i, i = 1, \dots, n\}, \quad (5a)$$

$$\mathcal{U} = \{u \in \mathbb{R} : \underline{u} \leq u \leq \bar{u}\}, \quad (5b)$$

where we assume that  $\underline{u} < 0 < \bar{u}$  to exclude situations where the system is uncontrollable.

**Problem 1.** Given the  $n$ th-order integrator system (4) together with rectangular state constraints  $E_{1:n}$  as in eqn. (5a) and input constraints  $\mathcal{U}$  as in eqn. (5b), characterize the maximal control-invariant set contained in  $E_{1:n}$ .

As we show in the following, we can analytically characterize the maximal control-invariant set in some, but not all, instances of this problem. To this end we will use the famous Nagumo's theorem [19] that states that for a set  $C^h$  defined by a differentiable function  $h$  it is necessary and sufficient that the input  $u$  satisfies the sub-tangentiality condition

$$h(x(t)) = 0 \implies \mathcal{L}h(x(t), u(t)) \geq 0 \quad (6)$$

in order for  $C^h$  to be rendered invariant. Here  $\mathcal{L}h(x, u)$  denotes the Lie derivative of  $h$  at  $x$  with respect to the dynamics resulting from the input  $u$ , i.e.

$$\mathcal{L}h(x, u) = \lim_{\epsilon \rightarrow 0} \frac{h(\phi(\epsilon; x, u)) - h(x)}{\epsilon}. \quad (7)$$

## III. INTEGRATOR SYSTEMS WITH A SINGLE BOUND

An  $n$ -dimensional instance of Problem 1 contains lower-dimensional instances as sub-problems. For example, setting  $\underline{x}_n = -\infty$  and  $\bar{x}_n = \infty$  reduces the  $n$ -dimensional instance to an  $n-1$ -dimensional instance. In order to exploit this symmetry we divide Problem 1 into  $2n$  problems—one for each state constraint. The next subsection presents a polynomial-based method for finding invariant sets for these subproblems. We then show in Section IV that the sub-solutions can be assembled into a solution of Problem 1. Since the treatment with upper and lower bounds are analogous we consider without loss of generality a lower bound.

**Problem 2.** Given an  $n$ -dimensional system (4) subject to input constraints (5b), find a function  $g_n$  that characterizes the maximal control invariant set  $\underline{C}_n$  contained inside the constraint set  $\underline{E}_n = \{x \in \mathbb{R}^n : \underline{x}_n \leq x_n\}$ .

The integrator system (4) is monotone everywhere with respect to the positive orthant [20], which means that if  $u_1(t) \leq u_2(t)$  for all  $t$ , then  $\phi(t; x_0, u_1) \preceq \phi(t; x_0, u_2)$ , where  $\preceq$  stands for component-wise inequality. It therefore follows that a given point  $x$  belongs to  $\underline{C}_n$  if and only if the maximal input  $u(t) = \bar{u}$  satisfies the lower bound for all times  $t \geq 0$ . Therefore, we can write a corresponding level set function as

$$g_n(x; \bar{u}, \underline{x}_n) = \min_{t \geq 0} [\phi(t; x, \bar{u})]_n - \underline{x}_n, \quad (8)$$

from which the following trivially follows:

**Proposition 1.** The set  $\underline{C}_n = \{x \in \mathbb{R}^n : g_n(x; \bar{u}, \underline{x}_n) \geq 0\}$  solves Problem 2.

### A. Characterization via Polynomial Roots

We can further expand the implicit characterization in (8) since the flow operator for the integrator system corresponding to a constant input is a polynomial in  $t$ . More precisely, the  $n$ :th coordinate of the flow operator  $[\phi(t; x, u)]_n$  for a constant control signal  $u(t) \equiv u$  can be written as the degree  $n$  polynomial

$$[\phi(t; x, u)]_n = \sum_{i=1}^n \frac{x_i t^{n-i}}{(n-i)!} + \frac{ut^n}{n!}, \quad (9)$$

and it follows that the condition  $\underline{g}_n \geq 0$  is equivalent to a global lower bound on the corresponding degree  $n$  polynomial.

As is well-known, for a sufficiently long  $t$  the local extremal values of (9) are attained at times  $t^* \in \{t_1, \dots, t_{n-1}\}$  (where we assume an ordering  $\text{Re}(t_i) \leq \text{Re}(t_{i+1})$ ) such that

$$\frac{d}{dt} [\phi(t^*; x, u)]_n = [\phi(t^*; x, u)]_{n-1} = 0. \quad (10)$$

Since  $\bar{u} > 0$  it follows that  $[\phi(t^*; x, \bar{u})]_n \rightarrow \infty$  as  $t \rightarrow \infty$ , and therefore the global minimum on the interval  $t \geq 0$  is attained either at  $t = 0$  or at a local minimum, so it suffices to check the value of the polynomial at these points in order to find the global minimum. The following algorithm formalizes this idea into a way of evaluating (8).

---

**Algorithm 1:** Evaluate  $\underline{g}_n(x; \bar{u}, \underline{x}_n)$

---

```

1  $\underline{g}_n \leftarrow x_n - \underline{x}_n$ 
2 for  $i = 1, \dots, n-1$  do
3    $t_i \leftarrow i$ :th root of  $[\phi(t; x, \bar{u})]_{n-1} = 0$ 
4   if  $t_i$  real,  $t_i \geq 0$ , and  $[\phi(t_i; x, \bar{u})]_n - \underline{x}_n \leq \underline{g}_n$  then
5      $\underline{g}_n \leftarrow [\phi(t_i; x, \bar{u})]_n - \underline{x}_n$ 
6 return  $\underline{g}_n$ 

```

---

If there is no real  $t > 0$  such that  $[\phi(t; x, \bar{u})]_{n-1} = 0$  then the minimum is simply given at  $t = 0$  as  $[\phi(0; x, \bar{u})]_n = x_n$  and the  $\underline{g}_n$  functions become

$$\underline{g}_n(x; \underline{x}_n, \bar{u}) = x_n - \underline{x}_n. \quad (11)$$

Note that this always happens for the case of  $n = 1$ .

For a problem in dimension  $n$  the corresponding polynomial root equation is of degree  $n-1$ , which implies that Algorithm 1 can be efficiently implemented for  $n \leq 5$  using the known closed-form algebraic expressions for roots of linear, quadratic, cubic, and quartic polynomials, respectively.

Problem 2 was stated for a lower state bound and an upper input bound. By symmetry, the same technique can be used to evaluate the function

$$\bar{g}_n(x; \underline{u}, \bar{x}_n) = \bar{x}_n - \max_{t \geq 0} [\phi(t; x, \underline{u})]_n \quad (12)$$

that represents the maximal control-invariant set for an upper state bound  $x_n \leq \bar{x}_n$  and a lower input bound  $\underline{u} \leq u$ . The same roots  $\{t_1, \dots, t_{n-1}\}$  are candidates for yielding the global maximum of  $\phi(t; x, \underline{u})$ .

We next state a technical lemma which is useful to determine which inputs  $u$  that are necessary to remain within the control-invariant set.

**Lemma 1.** For  $n = 1$  we have  $\mathcal{L}g_1(x, u) = u$  and  $\mathcal{L}\bar{g}_1(x, u) = -u$ . For  $n \geq 2$  when the minimum (maximum) of  $\underline{g}_n$  ( $\bar{g}_n$ ) is attained at time zero,  $\mathcal{L}\underline{g}_n(x, u) = x_{n-1}$ . In the other case, when the minimum (maximum) is attained at  $t^* \in \mathbb{R}_{>0}$ ,

$$\mathcal{L}\underline{g}_n(x, \bar{u}) = 0, \quad \mathcal{L}\bar{g}_n(x, \underline{u}) = 0, \quad (13)$$

and for  $u < \bar{u}$  we have:

$$\mathcal{L}\underline{g}_n(x, u) < \mathcal{L}\underline{g}_n(x, \bar{u}), \quad \mathcal{L}\bar{g}_n(x, u) > \mathcal{L}\bar{g}_n(x, \bar{u}). \quad (14)$$

*Proof.* For  $n = 1$ , and similarly for  $n \geq 2$  if the minimum of  $\phi(t; x, \bar{u})$  is obtained at  $t = 0$ , we have  $\min [\phi(t; x, \bar{u})]_n = x_n$ , with the Lie derivatives  $\mathcal{L}g_1(x, u) = u$  and  $\mathcal{L}\underline{g}_n(x, u) = x_{n-1}$  since  $\dot{x}_1 = u$  and  $\dot{x}_n = x_{n-1}$  respectively. An analogous argument can be made for  $\bar{g}_n$ . Next, for  $n \geq 2$  with  $t^* \in \mathbb{R}_{>0}$  giving min (max), we see from (8) and (10) that the derivative along  $\phi$  is zero when using the constant control signal  $u = \bar{u}$  ( $u = \underline{u}$ ) in the integration (9). Finally, (14) is a direct consequence of monotonicity.  $\square$

Below we use Lemma 1 to show what input constraints that are implied when different  $\underline{g}_n$  ( $\bar{g}_n$ ) are equal to zero.

**Lemma 2.** For  $\underline{g}_1 = 0$  the input  $u \geq 0$  is needed to maintain invariance (resp.  $u \leq 0$  for  $\bar{g}_1 = 0$ ). If  $\underline{g}_n = 0$  ( $\bar{g}_n = 0$ ) with no  $t^* \in \mathbb{R}_{>0}$  there are no constraints on  $u$ ; if  $\exists t^* \in \mathbb{R}_{>0}$  then  $u = \bar{u}$  (resp.  $\underline{u}$ ) is required to maintain invariance.

*Proof.* From eqn. (6) and Lemma 1 we see that  $\mathcal{L}g_1(x, u) \geq 0$  is achieved by  $u \geq 0$ . For  $\underline{g}_n = 0$  with no  $t^* \in \mathbb{R}_{>0}$  we have  $x_{n-1} \geq 0$  and there are, thus, no constraints on  $u$  as  $\mathcal{L}u \underline{g}_n = x_{n-1} \geq 0$ . If  $\exists t^* \in \mathbb{R}_{>0}$  in the case  $n \geq 2$  we note that (13) only allows  $u = \bar{u}$  for  $\underline{g}_n$  as (14) implies that all other  $u < \bar{u}$  give  $\mathcal{L}\underline{g}_n(x, u) < \mathcal{L}\underline{g}_n(x, \bar{u}) = 0$ . The symmetric argument applies for  $\bar{g}_n$ .  $\square$

### B. Explicit expressions for $n \leq 4$

It is of course sufficient to consider all candidate roots in Algorithm 1 to find the global minimum of  $\underline{g}_n$  or  $\bar{g}_n$ , but it is in fact possible to do better. As we show next, certain roots can be excluded from the algorithm which further improves efficiency. In particular, we obtain closed-form expressions for the  $n \leq 3$  cases, and for the  $n = 4$  case it is sufficient to consider just two of the three roots.

a)  $n = 1$ : Here, the level set functions corresponding to lower and upper bounds are simply given by

$$\begin{aligned} \underline{g}_1(x; \bar{u}, \underline{x}_1) &= x_1 - \underline{x}_1 \\ \bar{g}_1(x; \underline{u}, \bar{x}_1) &= \bar{x}_1 - x_1, \end{aligned} \quad (15)$$

since  $\min_{t \geq 0} [\phi(t; x, \bar{u})]_1 = \min_{t=0} [\phi(t; x, \bar{u})]_1 = x_1$ . This is a result of using maximum inputs  $u = \bar{u}$  and  $u = \underline{u}$  when constructing the functions, meaning that the value of  $x_1$  can only increase and decrease, respectively.

b)  $n = 2$ : For the second-order system, solving (10) results in

$$\min_{t \geq 0} [\phi(t; x, \bar{u})]_2 = x_2 - \frac{x_1^2}{2\bar{u}} \quad (16)$$

and the two  $g_2$  functions are given by insertion of (16) into (8) and (12), yielding

$$g_2(x_{1:2}; \bar{u}, \underline{x}_2) = \begin{cases} x_2 - \frac{x_1^2}{2\bar{u}} - \underline{x}_2 & \text{if } x_1 \leq 0, \\ x_2 - \underline{x}_2 & \text{o/w,} \end{cases} \quad (17)$$

and

$$\bar{g}_2(x_{1:2}; \underline{u}, \bar{x}_2) = \begin{cases} \bar{x}_2 - x_2 + \frac{x_1^2}{2\underline{u}} & \text{if } x_1 \geq 0, \\ \bar{x}_2 - x_2 & \text{o/w,} \end{cases} \quad (18)$$

where the conditions on  $x_1$  guarantee that  $t \in \mathbb{R}_{>0}$ .

c)  $n = 3$ : Although there are two possible roots in the  $n = 3$  case, only one of them needs to be evaluated.

**Proposition 2.** For  $n = 3$ , the extremal values in (8) and (12) are attained at the root  $t_2$ .

*Proof.* As  $g_3$  is constructed using  $u = \bar{u} > 0$  the highest order term in the  $n = 3$  polynomial (9) has a positive coefficient. The same holds for its derivative, which thus has the property

$$\lim_{t \rightarrow \pm\infty} [\phi(t; x, \bar{u})]_2 = \infty, \quad (19)$$

with either zero, one or two real roots  $t_i \in \mathbb{R}_{t>0}$ . Zero real roots imply that none of  $t_1, t_2$  are relevant and one root implies that  $t_1 = t_2$  is a double root. In the case of two real roots, as  $[\phi(t^*; x, u)]_2$  is a convex second order polynomial, the region  $t \in [t_1, t_2]$  implies that  $[\phi(t; x, u)]_2 \leq 0$  and that  $[\phi(t; x, u)]_3$  decreases from its local maximum  $t_1$  until its local minimum  $t_2$  is reached. In conclusion,  $t_1$  is never needed. By symmetry an analogous argument holds for the upper bound case.  $\square$

The closed-form expression for  $n = 3$  obtained by only considering  $t_2$  is

$$\min_{t \geq 0} [\phi(t; x, \bar{u})]_3 = x_3 + \frac{x_1^3 - (x_1^2 - 2\bar{u}x_2)^{3/2}}{3\bar{u}^2} - \frac{x_1x_2}{\bar{u}} \quad (20)$$

which after insertion in (8) and (12) results in

$$g_3 = \begin{cases} x_3 + \frac{x_1^3 - (x_1^2 - 2\bar{u}x_2)^{3/2}}{3\bar{u}^2} - \frac{x_1x_2}{\bar{u}} - \underline{x}_3 & x \in \underline{X} \\ x_3 - \underline{x}_3 & \text{o/w,} \end{cases} \quad (21)$$

and

$$\bar{g}_3 = \begin{cases} \bar{x}_3 - x_3 - \frac{x_1^3 - (x_1^2 - 2\underline{u}x_2)^{3/2}}{3\underline{u}^2} + \frac{x_1x_2}{\underline{u}} & x \in \bar{X} \\ \bar{x}_3 - x_3 & \text{o/w,} \end{cases} \quad (22)$$

where  $g_3 = g_3(x_{1:3}; \bar{u}, \underline{x}_3)$  and  $\bar{g}_3 = \bar{g}_3(x_{1:3}; \underline{u}, \bar{x}_3)$ . Also,  $x \in \underline{X}$  and  $x \in \bar{X}$  guarantee that  $t_2 \in \mathbb{R}_{>0}$ . The sets are constructed as  $\underline{X} = \mathcal{X}(\bar{u})$  and  $\bar{X} = \mathcal{X}(\underline{u})$  where  $\mathcal{X}(u) = \{x : x_2u \leq 0 \text{ if } x_1 \geq 0 \wedge x_1^2 \geq 2x_2u \text{ if } x_1 \leq 0\}$ .

d)  $n = 4$ : In this case there are three possible roots, but one can be excluded.

**Proposition 3.** For  $n = 4$ , the extremal values in (8) and (12) are attained at one of the roots  $t_1$  and  $t_3$ .

*Proof.* The corresponding property to (19) for the derivative in the  $n = 4$  case is

$$\lim_{t \rightarrow \pm\infty} [\phi(t; x, \bar{u})]_3 = \pm\infty. \quad (23)$$

The number of roots of  $[\phi(t; x, \bar{u})]_3$  in  $\mathbb{R}_{>0}$  is one, two, or three. Let us assume that  $t_2 \in \mathbb{R}_{>0}$ , as it could not give a minimum otherwise. As imaginary solutions to  $[\phi(t; x, \bar{u})]_3 = 0$  only exist in conjugate pairs we deduce that  $t_1, t_3 \in \mathbb{R}$  as  $t_2$  can not be the sole real root. If  $t_2$  is the *only* root, then  $t_1 = t_2 = t_3$  and  $t_2$  is redundant. Otherwise  $t_1 < t_2$  and the region  $t \in (t_1, t_2)$  is such that  $[\phi(t; x, \bar{u})]_3 \geq 0$  due to (23). Thus,  $[\phi(t; x, \bar{u})]_4$  can only obtain a local maximum at  $t_2$ . Conversely, for  $\bar{g}_4$ ,  $t_2$  can only be a local minimum. We conclude that  $t_2$  can be neglected in Algorithm 1.  $\square$

Even though the two candidates for  $g_4$  are analytical expressions they exceed the length appropriate for display.

#### IV. INTEGRATOR SYSTEMS WITH MULTIPLE BOUNDS

We now return to Problem 1 with both upper and lower constraints on all coordinates, i.e. the constraint is the intersection of the constraints in  $2n$  instances of Problem 2. A naïve approach would be to take the intersection of the corresponding control-invariant sets, i.e.,

$$h(x) = \min \left\{ \begin{array}{l} g_1(x_1), g_2(x_{1:2}), \dots, g_n(x_{1:n}), \\ \bar{g}_1(x_1), \bar{g}_2(x_{1:2}), \dots, \bar{g}_n(x_{1:n}) \end{array} \right\}, \quad (24)$$

and expect the resulting set to be invariant with respect to all constraints. However, the intersection of control-invariant sets is not in general control-invariant. Even if a point  $x$  lies in the intersection  $C_1 \cap C_2$  of two control-invariant sets  $C_1$  and  $C_2$ , there is not necessarily a single input that enforces invariance of both sets simultaneously. When this occurs we say that the two invariant sets are in *conflict*. Indeed, (24) fails to yield a set that is control-invariant for certain problem instances where  $n \geq 3$ . The method does actually work in the  $n \leq 2$  cases though, as illustrated next.

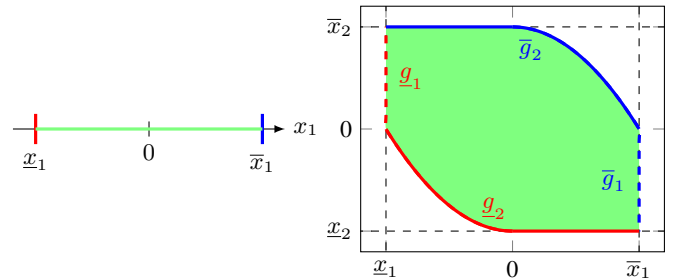


Fig. 2. Combined control-invariant set (24) for  $n = 1$  (left) and  $n = 2$  (right).

**Example 1.** In the  $n \leq 2$  cases the sets represented by (24) take the forms shown in Fig. 2. For  $n = 1$  the zero-level sets of individual  $g$  consist of the disjoint sets  $\{\underline{x}_1\}$  and  $\{\bar{x}_1\}$ , which excludes the possibility of a conflict. The same is true in the 2D case: the only potential points of conflict are  $(\bar{x}_1, \underline{x}_2)$  and  $(\underline{x}_1, \bar{x}_2)$ , but at both of these points the input  $u = 0$  satisfies both constraints by Lemma 2, which implies that the set is invariant.

We now note that different  $\underline{g}_n$  functions can not cause conflicts with each other.

**Proposition 4.** Let  $E_{1:n} = \{x : x_i \geq \underline{x}_i, i = 1, \dots, n\}$ . The maximal control-invariant set contained in  $E_{1:n}$  is given by  $C^{\underline{h}_n} = \{x : \underline{h}_n(x) \geq 0\}$  for

$$\underline{h}_n(x) = \min \left\{ \underline{g}_1(x_1), \underline{g}_2(x_{1:2}), \dots, \underline{g}_n(x_{1:n}) \right\}. \quad (25)$$

*Proof.* Functions  $\underline{g}_n$  correspond to lower bounds on the input signal  $u$  according to Lemma 2, so multiple such bounds can always be satisfied by selecting a  $u$  that satisfies the tightest bound. E.g. in the case of  $\underline{h}_n(x) = 0$ , choose  $\bar{u}$ .  $\square$

However, a conflict may arise between a lower bound corresponding to a function  $\underline{g}_n$  and an upper bound corresponding to a  $\bar{g}_m$ . We consider two cases of such conflicts: one where  $\underline{g}_n$  conflicts with  $\bar{g}_m$  for  $m < n$ , and the case when  $\underline{g}_n$  conflicts with  $\bar{g}_n$ . Below we discuss the  $n = 3$  case and present two ways in which bounds may be mutually conflicting. Having understood the nature of these conflicts we present a way to avoid the first type of conflict, and a condition that guarantees that the second conflict does not arise.

#### A. Conflict Between $\underline{g}_3$ and $\bar{g}_1$

Consider naively attempting to enforce invariance via the composition (24); the system may then be in a situation where it needs to utilize  $u = \bar{u}$  in order to avoid  $\underline{x}_3$ , but while doing so  $x_1$  grows so large that the bound  $x_1 \leq \bar{x}_1$  is in danger of being violated. At this point  $\underline{g}_3$  requires an input no smaller than  $\bar{u}$ , whereas  $\bar{g}_1$  requires an input no larger than 0, which is an obvious conflict.

In order to avoid this situation lookahead information needs to be added to account for the fact that it might not be possible to use  $u(t) \equiv \bar{u}$  as long as there exists a  $t^* \in \mathbb{R}_{>0}$  for  $\underline{g}_3$ . However, the system is still monotone under the constraint  $x_1 \leq \bar{x}_1$ , so we can consider an alternative extremal trajectory. In this case the trajectory that maximizes  $x_3$  while respecting  $x_1 \leq \bar{x}_1$  is one that uses  $u = \bar{u}$  until  $x_1 = \bar{x}_1$ , and then switches to  $u = 0$ . We can account for this additional constraint by introducing a modified  $g$  function as follows:

$$\underline{g}_3^b(x_{1:3}; \bar{u}, \bar{x}_1, \underline{x}_3) = \min_{t \in [0, \frac{\bar{x}_1 - x_1}{\bar{u}}]} \underline{g}_2([\phi(t; x_{1:3}, \bar{u})]_{2:3}; \bar{x}_1, \underline{x}_3). \quad (26)$$

That is, via the flow  $\phi$  we pull back  $\underline{g}_2$  corresponding to the two-dimensional system

$$\begin{aligned} \dot{x}_2 &= \bar{x}_1, \\ \dot{x}_3 &= x_2, \end{aligned} \quad (27)$$

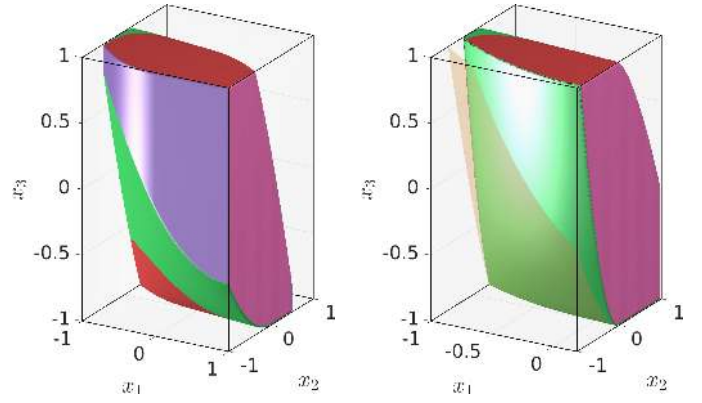


Fig. 3. Invariant set for a triple integrator system ( $n = 3$ ), where the surfaces in the zero-level set of (28) corresponding to  $\bar{g}_1$  and  $\underline{g}_1$  are pink,  $\bar{g}_2$  and  $\underline{g}_2$  are purple,  $\bar{g}_3$  and  $\underline{g}_3$  are red, and  $\bar{g}_3^b$  and  $\underline{g}_3^b$  are green. In the left figure all upper bounds are set to 1, and the lower ones to -1. In this case the surface corresponding to  $\underline{g}_3^b$  lies very close to the one corresponding to  $\underline{g}_3$ . In the right figure the upper bound on  $x_1$  is changed to 0.2, which makes the effect of adding  $\underline{g}_3^b$  more pronounced ( $\underline{g}_3$  is shown translucent).

that governs the dynamics on the surface  $x_1 = \bar{x}_1$ . We remark that as  $\bar{x}_1 \rightarrow \infty$ ,  $\underline{g}_3^b \rightarrow \underline{g}_3$ , i.e. we retrieve the solution to the problem without an upper bound on  $\bar{x}_1$ . Furthermore, it is sufficient to evaluate the minimum at the endpoint  $\frac{\bar{x}_1 - x_1}{\bar{u}}$  if jointly considering  $\underline{g}_3$ , since a minimum of  $\underline{g}_3^b$  that is attained somewhere on the interval  $[0, \frac{\bar{x}_1 - x_1}{\bar{u}}]$  already corresponds to a minimum of  $\underline{g}_3$ .

#### B. Conflict Between $\underline{g}_3$ and $\bar{g}_3$

The other type of conflict is more difficult to resolve since it has a circular dependency that prevents recursive feasibility from being established. Instead, we present a result that guarantees that this type of conflict does not occur in the first place.

**Theorem 1.** Consider the function

$$h(x) = \min \left\{ \underline{g}_1(x_1), \bar{g}_1(x_1), \underline{g}_2(x_{1:2}), \bar{g}_2(x_{1:2}), \underline{g}_3(x_{1:3}), \bar{g}_3(x_{1:3}), \underline{g}_3^b(x_{1:3}), \bar{g}_3^b(x_{1:3}) \right\}. \quad (28)$$

If there exist points  $z_1, z_2, z_3, z_4$  such that  $h(\underline{x}_1, z_1, \underline{x}_3) \geq 0$ ,  $h(\underline{x}_1, z_2, \bar{x}_3) \geq 0$ ,  $h(\bar{x}_1, z_3, \underline{x}_3) \geq 0$ , and  $h(\bar{x}_1, z_4, \bar{x}_3) \geq 0$ , then  $C^h = \{x : h(x) \geq 0\}$  is the maximal control-invariant set contained in  $E_{1:3}$ .

Equivalently, the condition says that the projection of the set  $\{x : h(x) \geq 0\}$  onto the first and last coordinate should equal the projection of the constraint set  $E_{1:3}$ . This guarantees that surfaces corresponding to input  $\bar{u}$  do not touch surfaces corresponding to input  $\underline{u}$ . Fig. 3 illustrates the  $n = 3$  maximal invariant set, and the influence of the different components of (28). As can be seen in the figures, the set touches the bounding box at the four edges that lie parallel to the  $x_2$  axis, which implies that the condition in Theorem 1 is fulfilled.

*Proof.* As discussed above, conflicts between  $\bar{g}_3$  and  $\underline{g}_1$ , and between  $\underline{g}_3$  and  $\bar{g}_1$ , are eliminated by the functions  $\bar{g}_3^b$  and

$\underline{g}_3^b$ . There can not be conflicts between  $\bar{g}_3$  and  $\underline{g}_2$ , since by necessity  $x_2 > 0 > \underline{x}_2$  whenever  $\bar{g}_3 = 0$  and  $\exists t^* \in \mathbb{R}_{>0}$ , and vice versa for  $\underline{g}_3$  and  $\bar{g}_2$ . What remains is to show that there are no conflicts possible between  $\bar{g}_3$  and  $\underline{g}_3$ , i.e. that there are no points  $x$  inside  $C^h$  s.t.  $\bar{g}_3(x) = \underline{g}_3(x) = 0$ . This follows from the condition above since  $C^h$  is a convex set (it is an intersection of maximal control-invariant sets inside a convex region  $E_{1:3}$ , which are all convex). Therefore the rectangle with corners  $(\underline{x}_1, z_1, \underline{x}_3)$ ,  $(\underline{x}_1, z_2, \bar{x}_3)$ ,  $(\bar{x}_1, z_3, \underline{x}_3)$ , and  $(\bar{x}_1, z_4, \bar{x}_3)$  splits  $C^h$  into two parts, and the sets  $\{x : \bar{g}_3 = 0 \vee \underline{g}_3 = 0\}$  and  $\{x : \underline{g}_3^b = 0 \vee \underline{g}_3 = 0\}$  are in different parts since those functions are monotone in  $x_2$ . Thus no conflicts are possible which implies that the set is control-invariant.  $\square$

For cases  $n \geq 4$  the same methodology of “pulling back” lower-order constraint surfaces works in principle, but the extremal trajectories are more complicated than in the  $n = 3$  case, and it may not be enough to evaluate a finite number of points to find the minimum. For example, avoiding the lower bound  $\underline{x}_4$  while respecting upper bounds  $\bar{x}_1$  and  $\bar{x}_2$  requires accounting for several cases depending on whether  $\bar{x}_1$  and/or  $\bar{x}_2$  saturates.

## V. INVARIANT SET FOR QUADROTOR DYNAMICS USING DIFFERENTIAL FLATNESS

Finding analytical expressions for maximal control-invariant sets not only has a value in itself; in fact, the sets can also be used in various applications. In this article, we utilize the results for  $n = 4$  to construct an obstacle-avoidance controller for UAVs. The most important state constraint in obstacle avoidance is a lower bound on the relative distance to another object, and potentially the derivatives of this quantity. Therefore only the lower bounds  $\underline{g}_1, \underline{g}_2, \underline{g}_3$  and  $\underline{g}_4$  are used, which, by Proposition 4 represent the maximum control-invariant set for these constraints.

The dynamics of a quadrotor can be described by the model

$$\begin{aligned} m\ddot{r} &= -mge_3 + f_z Qe_3, \\ \dot{Q} &= Q\hat{\omega}, \\ J\dot{\omega} &= -\hat{\omega}J\omega + \tau, \\ \ddot{f}_z &= F_z, \end{aligned} \quad (29)$$

evolving on  $SE(3) = \mathbb{R}^3 \otimes SO(3)$ , where  $r \in \mathbb{R}^3$  is the position of the quadrotor in Cartesian coordinates,  $f_z, \dot{f}_z, \ddot{f}_z \in \mathbb{R}$  is the vertical thrust from its four propellers and its derivatives,  $Q \in SO(3)$  is a rotation matrix representing attitude,  $\omega \in \mathbb{R}^3$  is the angular velocity in the body frame and  $J \in \mathbb{R}^{3 \times 3}$  is the inertia matrix. Moreover, the hat operator  $\wedge$  maps  $\omega$  to the corresponding skew-symmetric matrix. The angular torque  $\tau$  and the second time derivative of the thrust  $F_z$  are the inputs. Note that the last equation has been added as a dynamical model extension compared to the standard model in e.g. [21] so as to include  $f_z$  and  $\dot{f}_z$  as states, which is required to make the mapping from the state space to the flat space invertible.

The key idea behind our approach is to do obstacle avoidance in flat space, which requires mapping the state of the system to flat space, potentially modifying the behavior using a *safety filter*, and then mapping the resulting modified state back to the original state space. To achieve this we introduce the procedure in Fig. 4 and separately discuss each step.

### A. $SE(3)$ to Flat Space

In general, the differential flatness property of a dynamical system (1) implies that the state vector  $x$  and input vector  $u$  can be rewritten compactly in a fewer number of so-called flat outputs  $\sigma$  and a finite number  $q$  of time derivatives of those outputs,

$$\begin{cases} u = \beta(\sigma, \dot{\sigma}, \dots, \sigma^{(q)}), \\ x = \mu(\sigma, \dot{\sigma}, \dots, \sigma^{(q)}), \end{cases} \quad (30)$$

as discussed in [14]. These functions  $(\beta, \mu)$ , known as the endogenous transformation, will be used in step V-E. As the transformation is invertible one can compute the flat outputs as a function of the state and input vectors as  $\sigma = \eta(x, u, \dot{u}, \dots, u^{(p)})$  for some finite number of time derivatives  $p \in \mathbb{R}$  and the invertible function  $\eta$ , which we use in this step.

For this quadrotor model  $u(t) \equiv (F_z(t), \tau(t))$  and  $x(t) \equiv (r(t), \dot{r}(t), Q(t), \omega(t), f_z(t), \dot{f}_z(t))$ . One usually chooses the flat outputs as the position and yaw angle [14], that is  $\sigma = \eta(x) = [x, y, z, \psi]^T$ , with the last  $x$  being the coordinate. However, without loss of generality the yaw angle  $\psi$  can be considered fixed as it only rotates the quadrotor around its own axis. The flat output simply becomes the position

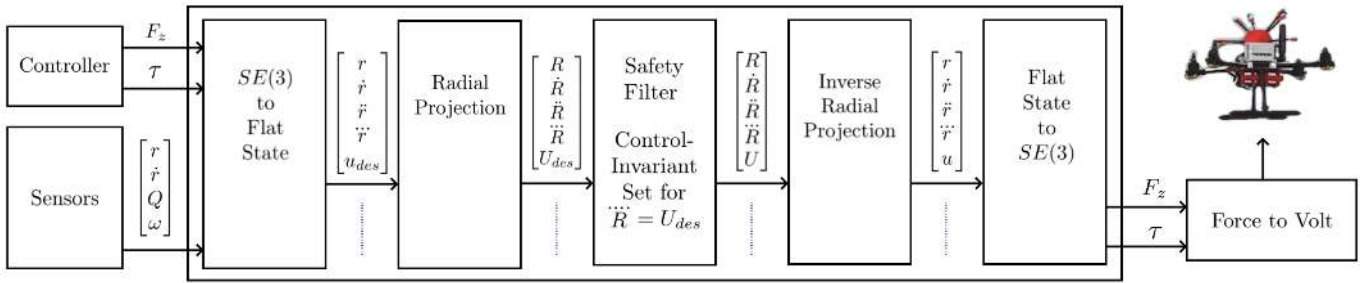


Fig. 4. High-level schematic of the blocks making obstacle avoidance on a quadrotor possible. The differential flatness property of the quadrotor allows a transformation from its 12-dimensional state to its relative distance  $R \in \mathbb{R}$  to another object and its derivatives. The invariant set for an integrator system of order  $n = 4$  can be used to guarantee that the quadrotor satisfies the lower constraint on  $R$ , i.e., never comes too close to another object. The inverse transformations enable the inputs  $F_z$ —the second derivative of the vertical thrust—and  $\tau$ —the angular torque—to be updated so as to keep the quadrotor safe.

vector  $\sigma = [x, y, z]^T = r$ . The lowest  $q$  in the endogenous transformation (30) taking the complete quadrotor model into consideration is  $q = 4$ , so we want to derive the flat dynamics  $(\sigma, \dot{\sigma}, \ddot{\sigma}, \overset{\cdot\cdot\cdot}{\sigma}, \overset{\cdot\cdot\cdot\cdot}{\sigma}) = (r, \dot{r}, \ddot{r}, \overset{\cdot\cdot\cdot}{r}, \overset{\cdot\cdot\cdot\cdot}{r})$  from  $(r, \dot{r}, Q, \omega, f_z, \dot{f}_z)$  and inputs  $(F_z, \tau)$ . Differentiation of equation one in (29) gives

$$\begin{aligned}\ddot{r} &= \frac{1}{m} \left( \dot{f}_z Q e_3 + f_z Q \dot{\omega} e_3 \right), \\ \overset{\cdot\cdot\cdot}{r} &= \frac{1}{m} \left( F_z Q e_3 + 2 \dot{f}_z Q \dot{\omega} e_3 + f_z \overset{\cdot\cdot\cdot}{Q} \dot{\omega} e_3 + \right. \\ &\quad \left. + f_z Q [J^{-1}(\tau - \dot{\omega} J \omega)]^\wedge e_3 \right).\end{aligned}\quad (31)$$

### B. Radial Projection

We next go from three-dimensional space to the one-dimensional space describing the relative motion with respect to an obstacle. Using spherical transformation of the position  $r$  and its derivatives we obtain the scalars  $(R, \dot{R}, \ddot{R}, \overset{\cdot\cdot\cdot}{R}, \overset{\cdot\cdot\cdot\cdot}{R})$ . Redefining the position as  $r = r - r_{obstacle}$  we achieve the notion of a relative distance to another object. We find the radial projection by  $R = (r^T r)^{1/2}$  and the higher-order derivatives are given by

$$\begin{aligned}\dot{R} &= \frac{r^T \dot{r}}{R}, \quad \ddot{R} = \frac{\dot{r}^T \dot{r} + r^T \ddot{r}}{R} - \frac{r^T \dot{r}}{R^2} \dot{R}, \\ \overset{\cdot\cdot\cdot}{R} &= \frac{3 \dot{r}^T \ddot{r} + r^T \overset{\cdot\cdot\cdot}{r}}{R} - 2 \frac{\dot{r}^T \dot{r} + r^T \ddot{r}}{R^2} \dot{R} + 2 \frac{r^T \dot{r}}{R^3} \dot{R}^2 - \frac{r^T \dot{r}}{R^2} \ddot{R}, \\ \overset{\cdot\cdot\cdot\cdot}{R} &= \frac{4 \dot{r}^T \overset{\cdot\cdot\cdot}{r} + 3 \ddot{r}^T \ddot{r} + r^T \overset{\cdot\cdot\cdot\cdot}{r}}{R} - 3 \frac{\dot{r}^T \ddot{r} + r^T \overset{\cdot\cdot\cdot}{r}}{R^2} \dot{R} \\ &\quad - 3 \frac{\dot{r}^T \dot{r} + r^T \ddot{r}}{R^2} \ddot{R} - \frac{r^T \dot{r}}{R^2} \overset{\cdot\cdot\cdot}{R} + 6 \frac{\dot{r}^T \dot{r} + r^T \ddot{r}}{R^3} \dot{R}^2 \\ &\quad + 6 \frac{r^T \dot{r}}{R^3} \ddot{R} \dot{R} - 6 \frac{r^T \dot{r}}{R^4} \dot{R}^3.\end{aligned}\quad (32)$$

Interpreting  $\overset{\cdot\cdot\cdot}{r}$  as  $u_{des} \in \mathbb{R}^3$  and  $\overset{\cdot\cdot\cdot\cdot}{R}$  as  $U_{des} \in \mathbb{R}$  the transformed radial system now constitutes an integrator system of order  $n = 4$ . This means that we can control a fourth-order integrator system and then use the flatness equations (30) to calculate the control actions and trajectories for the real system, the quadrotor.

### C. Safety Filter

The safety filter is implemented to guarantee safety of the quadrotor and uses the methodology of enforcing invariance via control barrier functions (CBFs). This approach of using CBFs was introduced a couple of years ago [22] and extends the concept of Nagumo's tangentiality condition (6) to include an extra term  $\alpha h$  with  $\alpha$  being a class  $K$  function; we here choose a scalar  $\alpha \in \mathbb{R}_{>0}$ . The effect of the extra term is smoother behavior as the system approaches the boundary of the control-invariant set. The filter works by taking a nominal input  $U_{des}$  and then finding the closest safe input  $U$ , which is done by solving

$$\min_U \|U - U_{des}\| \quad \text{s.t.} \quad \mathcal{L}g_i(R_{0:i}, U) + \alpha g_i(R_{0:i}) \geq 0 \quad \forall i, \quad (33)$$

where  $R_{0:i} = \{R, \frac{dR}{dt}, \dots, \frac{d^i R}{dt^i}\}$  and  $i = 1, 2, 3, 4$ . This optimization problem is explained in more detail in [23] and yields acceptable inputs  $U$  guaranteeing safety.

### D. Inverse Radial Projection

Calculating the new  $\overset{\cdot\cdot\cdot}{r} = u$  from the update in  $\overset{\cdot\cdot\cdot\cdot}{R} = U_{des} \mapsto U$  can be done using the equation for  $\overset{\cdot\cdot\cdot}{R}$  in (32). As only one term consists of  $\overset{\cdot\cdot\cdot}{r} = u_{des}$  that we want to map to  $u$  we denote the other terms by  $T$ . Furthermore, as

$$\overset{\cdot\cdot\cdot}{R} = U = \frac{r^T u}{R} + T, \quad (34)$$

we see that only changes to  $u$  in the direction of  $r$  has an effect on  $U$ . In order to find the  $u$  closest to  $u_{des}$  that yields a desired  $U$  we insert  $u = u_{des} + r\lambda$  with  $\lambda \in \mathbb{R}$  into (34) and solve for  $\lambda$ , which yields a modified safe input  $u \in \mathbb{R}^3$ .

### E. Flat State to SE(3)

Lastly, one needs to perform the endogenous transformation in equation (30). The inputs needed are  $F_z$  and  $\tau$ , which can be calculated using the quadrotor model (29). Denoting  $t = \ddot{r} + g e_3$ , we have

$$\begin{aligned}\ddot{f}_z &= m \left( \frac{\overset{\cdot\cdot\cdot}{r}^T \ddot{r} + t^T u}{\|t\|} - \frac{(t^T \overset{\cdot\cdot\cdot}{r})^2}{\|t\|^3} \right), \\ \tau &= J \dot{\omega} + \hat{\omega} J \omega = J(Q^T \overset{\cdot\cdot\cdot}{Q})^\vee + J(\overset{\cdot\cdot\cdot}{Q}^T \overset{\cdot\cdot\cdot}{Q})^\vee + \hat{\omega} J \omega,\end{aligned}\quad (35)$$

where  $\vee$  denotes the inverse of the hat map. In turn,  $\overset{\cdot\cdot\cdot}{Q}$  and  $\overset{\cdot\cdot\cdot\cdot}{Q}$  can be computed using  $Q = [x_B \ y_B \ z_B]$  where  $z_B = \frac{t}{\|t\|}$ ,  $x_C = [\cos \psi \ \sin \psi \ 0]^T$ ,  $y_B = \frac{z_B \times x_C}{\|z_B \times x_C\|}$  and  $x_B = y_B \times z_B$ . The derivatives  $\overset{\cdot\cdot\cdot}{Q} = [\dot{x}_B \ \dot{y}_B \ \dot{z}_B]$  and  $\overset{\cdot\cdot\cdot\cdot}{Q} = [\ddot{x}_B \ \ddot{y}_B \ \ddot{z}_B]$  are obtained from  $t, \ddot{r}, u, \psi, \dot{\psi}$  and  $\ddot{\psi}$ .

## VI. APPLICATION IN QUADROTOR OBSTACLE AVOIDANCE

Bringing quadrotors into the real-world will result in deployment in cluttered environments, which implies the need to guarantee safety. Although obstacle avoidance is well studied in the quadrotor domain, existing approaches either work at the trajectory planning level which is more computationally intensive (e.g. [24]), or do not take input bounds into account [15]. The analytical approach in this work resulted in easy-to-evaluate control-invariant sets, and although the flatness mapping only gives an implicit bound on the input, the method does bound the snap (4th derivative) of the motion, which is a common proxy for input bounds in UAV control [16]. We have implemented the safety filter in a MATLAB/Simulink SimMechanics environment to validate the method.

In particular, a test flight was carried out using a nominal trajectory-tracking controller that takes the quadrotor from  $r_{start} = [0, -1, 0]$  to  $r_{end} = [4, 1, 4]$  with an obstacle in  $r_{obstacle} = [2, 0, 1]$ . In Fig. 1 one can see that the effect of the filter is to rotate the drone away from the sphere, and then pass above it. As shown in Fig. 5 the applied angular torques  $\tau$  (solid) differ from the angular torques generated by the nominal controller  $\tau_{des}$  (dashed) as the obstacle is avoided. The second component  $\tau_{des}^2$  grows large, which is a consequence of the nominal controller attempting to return the quadrotor to the original trajectory. The alteration due to the filter is also evident in the subplot displaying  $f_z$ ; the initial dip at  $t = 4$  represents the brake and rotation away from

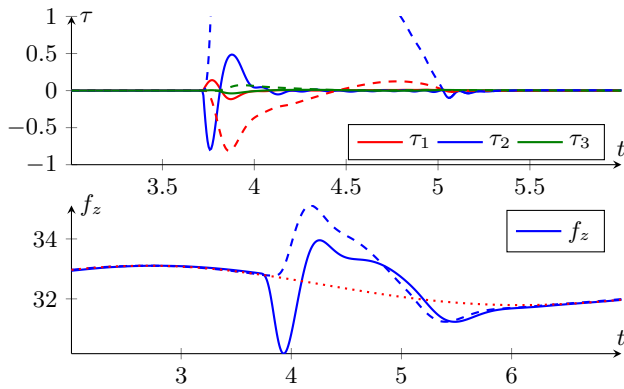


Fig. 5. Dashed lines represent desired inputs from the controller, solid lines are the filtered safe inputs that are applied to the plant. The dotted red line in the lower plot shows the vertical thrust  $f_z$  when the filter is disabled.

the obstacle and the subsequent peak represents the quadrotor flying over the obstacle. Furthermore, in Fig. 6 the desired trajectory is compared to the realized safe trajectory.

## VII. CONCLUSIONS

In this work, a generic analytical method for constructing control-invariant sets for the integrator system of order  $n$  has been introduced. Also, for the orders  $n \leq 3$  we present closed-form expressions for the case with both upper and lower bounds. Even though the method is not particularly scalable for higher-order integrator systems due to the complexity in finding roots of higher-order polynomials, it is worth noting that few systems have relative degree greater than 4.

While finding invariant sets is of independent theoretical interest, the main practical benefit is that knowledge of the invariant set informs how input signals that enforce invariance can be constructed. In addition, characterizing them via analytical expressions substantially decreases the required computational power, which is particularly beneficial for small drones. We here showed how one-sided invariant sets for the integrator system for  $n = 4$  could be utilized to formulate an obstacle avoidance controller for quadrotor via differential flatness. The same technique would also work for other systems that are differentially flat, with the caveat that in some flat systems the flat outputs do not correspond to physically meaningful quantities like for the quadrotor.

As future work one could extend the control-invariant sets to include robustness to noise, which would be useful in real-world implementations. The quadrotor application could moreover naturally be extended to include avoidance of multiple obstacles.

## REFERENCES

- [1] D. P. Bertsekas. Infinite-Time Reachability of State-Space Regions by Using Feedback Control. *IEEE Transactions on Automatic Control*, 17(5):604–613, 1972.
- [2] P. Trodden. A One-Step Approach to Computing a Polytopic Robust Positively Invariant Set. *IEEE Transactions on Automatic Control*, 61(12):4100–4105, 2016.
- [3] M. Fiacchini and M. Almir. Computing control invariant sets in high dimension is easy, 2018. <http://arxiv.org/abs/1810.10372>.

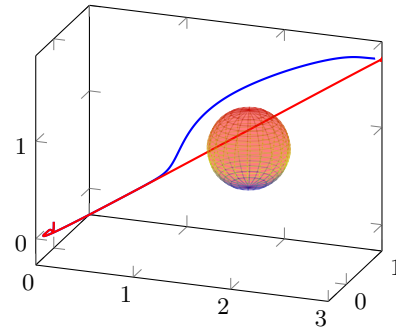


Fig. 6. The nominal trajectory (red) is shown to pass through the obstacle, while the safe trajectory (blue) stays away from it.

- [4] F. Tahir and I. M. Jaimoukha. Low-Complexity Polytopic Invariant Sets for Linear Systems Subject to Norm-Bounded Uncertainty. *IEEE Transactions on Automatic Control*, 60(5):1416–1421, 2015.
- [5] M. Korda, D. Henrion, and C. N. Jones. Convex Computation of the Maximum Controlled Invariant Set For Polynomial Control Systems. *SIAM Journal on Control and Optimization*, 52(5):2944–2969, 2014.
- [6] T. Alamo, A. Cepeda, and D. Limon. Improved computation of ellipsoidal invariant sets for saturated control systems. *Proc. IEEE CDC-ECC*, pages 6216–6221, 2005.
- [7] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control*, 50(7):947–957, 2005.
- [8] Y. Shoukry, P. Tabuada, S. Tsuei, M. B. Milam, J. W. Grizzle, and A. D. Ames. Closed-form controlled invariant sets for pedestrian avoidance. In *Proc. ACC*, pages 1622–1628, 2017.
- [9] T. Gurriet, A. Singletary, J. Reher, L. Ciarletta, E. Feron, and A. D. Ames. Towards a Framework for Realizable Safety Critical Control through Active Set Invariance. In *ACM/IEEE International Conference on Cyber-Physical Systems*, pages 98–106, 2018.
- [10] Q. Nguyen, A. Hereid, J. W. Grizzle, A. D. Ames, and K. Sreenath. 3D dynamic walking on stepping stones with control barrier functions. In *Proc. IEEE CDC*, pages 827–834, 2016.
- [11] L. Wang, A. D. Ames, and M. Egerstedt. Safe certificate-based maneuvers for teams of quadrotors using differential flatness. In *Proc. IEEE ICRA*, 2017.
- [12] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada. Control Barrier Function Based Quadratic Programs for Safety Critical Systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2017.
- [13] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames. Robustness of Control Barrier Functions for Safety Critical Control. In *Proc. IFAC ADHS*, 2016.
- [14] D. Zhou and M. Schwager. Vector field following for quadrotors using differential flatness. *Proc. IEEE ICRA*, pages 6567–6572, 2014.
- [15] G. Wu and K. Sreenath. Safety-Critical Control of a 3D Quadrotor With Range-Limited Sensing. In *Proc. ASME CDSC*, 2016.
- [16] D. Mellinger and V. Kumar. Minimum snap trajectory generation and control for quadrotors. In *Proc. IEEE ICRA*, pages 2520–2525, 2011.
- [17] J.-P. Aubin. *Viability Theory*. Birkhäuser Boston, 2009.
- [18] F. Blanchini. Set invariance in control. *Automatica*, 35(11):1747–1767, 1999.
- [19] M. Nagumo. Über die Lage der Intergralkurven gewöhnlicher Differentialgleichungen. *Proc. Phys. Math. Soc. Jpn.*, 24:551–559, 1942.
- [20] D. Angeli and E.D. Sontag. Monotone control systems. *IEEE Transactions on Automatic Control*, 48(10):1684–1698, 2003.
- [21] T. Lee. Global Exponential Attitude Tracking Controls on SO(3). *IEEE Transactions on Automatic Control*, 60(10):2837–2842, 2015.
- [22] A. D. Ames, J. W. Grizzle, and P. Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *Proc. IEEE CDC*, pages 6271–6278, 2014.
- [23] T. Gurriet, M. Mote, A. D. Ames, and E. Feron. An Online Approach to Active Set Invariance. In *Proc. IEEE CDC*, pages 3592–3599, 2018.
- [24] B. Landry, R. Deits, P. R. Florence, and R.I. Tedrake. Aggressive quadrotor flight through cluttered environments using mixed integer programming. In *Proc. IEEE ICRA*, pages 1469–1475, 2016.