

Decomposition of Reachable Sets and Tubes for a Class of Nonlinear Systems

Mo Chen, Sylvia L. Herbert, Mahesh S. Vashishtha, Somil Bansal, Claire J. Tomlin

Abstract—Reachability analysis provides formal guarantees for performance and safety properties of nonlinear control systems. Here, one aims to compute the backward reachable set (BRS) or tube (BRT) – the set of states from which the system can be driven into a target set at a particular time or within a time interval, respectively. The computational complexity of current approaches scales poorly, making application to high-dimensional systems intractable. We propose a technique that decomposes the dynamics of a general class of nonlinear systems into subsystems which may be coupled through common states, controls, and disturbances. Despite this coupling, BRSs and BRTs can be computed efficiently using our technique without incurring additional approximation errors and without the need for linearizing dynamics or approximating sets as polytopes. Computations of BRSs and BRTs now become orders of magnitude faster, and for the first time BRSs and BRTs for many high-dimensional nonlinear control systems can be computed using the Hamilton-Jacobi (HJ) formulation. In situations involving bounded adversarial disturbances, our proposed method can obtain slightly conservative results. We demonstrate our theory by numerically computing BRSs and BRTs for several systems, including the 6D Acrobatic Quadrotor using the HJ formulation.

I. INTRODUCTION

Many important real-world systems are described by complex nonlinear models whose behavior can be non-intuitive and difficult to predict. These systems include power [2], [3], biological [4], [5], and robotic systems such as autonomous cars and unmanned aerial vehicles [6], [7]. With the recent advancements of sophisticated system modeling in these areas, the dimensionality of system models has grown significantly. Many of these systems are also safety-critical, making their verification extremely important. As a result, computationally tractable tools for the analysis of these nonlinear, high-dimensional, and safety-critical systems are urgently needed.

Verification of systems is challenging for many reasons. First, all possible system behaviors must be accounted for. This makes most simulation-based approaches insufficient, and thus formal verification methods are needed. Second, many practical systems are affected by disturbances in the environment. In addition, the systems evolve in continuous time with complex, nonlinear dynamics. Lastly, perhaps the most difficult challenge of all is that these systems often have high-dimensional state spaces.

Reachability analysis is an important formal verification method for guaranteeing performance and safety properties of systems. However, current methods do not simultaneously

address all of the above challenges. For example, [8], [9] excel in determining whether system trajectories from a small set of initial conditions could potentially enter a set of unsafe states, but do not provide the backward reachable set (BRS) or tube (BRT) – the set of all initial states from which entering some target set is inevitable. Due to the challenges of computing BRSs and BRTs, the state-of-the-art methods need to make trade offs on different axes of considerations such as computational scalability, generality of system dynamics, existence of control and/or disturbance variables, and flexibility in representation of sets.

For example, the methods presented in [10]–[13] have had success in analyzing relatively high-dimensional affine systems using sets of pre-specified shapes, such as polytopes or hyperplanes. Other potentially less scalable methods are able to handle systems with the more complex dynamics [10], [14]–[17]. Computational scalability varies among these different methods, with the most scalable methods requiring that the system dynamics do not involve control and disturbance variables. The work in [18] accounts for both control and disturbances, but is only applicable to linear systems. Methods that can account for general nonlinear systems such as [19] also sometimes represent sets using simple shapes such as polytopes, potentially sacrificing representation fidelity in favor of the other aspects mentioned earlier. Hamilton-Jacobi (HJ) formulations [20]–[23] excel in handling general nonlinear dynamics, control and disturbance variables, and flexible set representations via a grid-based approach; however, these methods are the least computationally scalable. Still other methods make a variety of other assumptions to make desirable trade offs [24]–[26]. In addition, under some special scenarios, it may be possible to obtain small computational benefits while minimizing trade offs in other axes of consideration by exploiting system structure [27]–[32].

In this paper, we present a system decomposition method for computing BRSs and BRTs of a class of nonlinear systems. Our method drastically reduces dimensionality without making any other trade offs. Our method first computes BRSs for lower-dimensional subsystems, and then reconstructs the full-dimensional BRS without incurring additional approximation errors other than those arising from the lower-dimensional computations. Crucially, the subsystems can be coupled through common states, controls, and disturbances. The treatment of this coupling distinguishes our method from others which consider completely decoupled subsystems, potentially obtained through transformations [33], [34]. Since BRTs are also of great interest in many situations, we prove conditions under which BRTs can also be decomposed.

The theory we present in this paper is compatible with any methods that compute BRSs and BRTs, such as [10], [14], [15], [27], [30] and others mentioned earlier. In addition, when

This work has been supported in part by NSF under CPS:ActionWebs (CNS-931843), by ONR under the HUNT (N0014-08-0696) and SMARTS (N00014-09-1-1051) MURIs and by grant N00014-12-1-0609, by AFOSR under the CHASE MURI (FA9550-10-1-0567). The research of M. Chen has received funding from the “NSERC PGS-D” Program.

All authors are with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley. {mochen72, sylvia.herbert, msv, somil, tomlin}@berkeley.edu

Theorem 2 of this paper is partially taken from our conference paper [1].

Digital Object Identifier 10.1109/TAC.2018.2797194

different decomposition methods are combined together, even more dimensionality reduction can be achieved. This paper will be presented as follows:

- In Sections II and III we introduce the basic concept of reachability, and all the definitions needed for our proposed decomposition technique.
- In Sections IV and V we present our theoretical results related to decomposing BRSs for systems involving a control variable, but *not* involving a disturbance variable.
- In Section VI we show how BRTs can be decomposed.
- In Section VII we demonstrate our decomposition method on high-dimensional systems.
- In Section VIII we discuss how the presence of disturbances affects the above theoretical results.
- We will also present numerical results obtained through the Hamilton-Jacobi (HJ) reachability formulation in [21] throughout the paper to validate our theory.

II. BACKGROUND

There are various formulations for computing the BRS and BRT when the system dimensionality is low. In this section, we give the basic mathematical problem setup to provide a foundation on which we build the new proposed theory.

A. System Dynamics

Let $z \in \mathbb{R}^n$ be the system state, which evolves according to the ordinary differential equation (ODE)

$$\frac{dz(s)}{ds} = \dot{z}(s) = f(z(s), u(s)), s \in [t, 0], u(s) \in \mathcal{U} \quad (1)$$

In general, the theory we present is applicable when some states are periodic dimensions (such as angles), but for simplicity we will consider \mathbb{R}^n . The control is denoted by $u(s)$, with the control function $u(\cdot) \in \mathcal{U}$ being drawn from the set of measurable functions.

The control signal $u \in \mathcal{U} \subset \mathbb{R}^{n_u}$ is compact and $t < 0$. The system dynamics, or flow field, $f : \mathbb{R}^n \times \mathcal{U} \rightarrow \mathbb{R}^n$ is assumed to be uniformly continuous, bounded, and Lipschitz continuous in¹ z for fixed u . Therefore, given $u(\cdot) \in \mathcal{U}$, there exists a unique trajectory solving (1) [35]. We will denote solutions, or trajectories of (1) starting from state z at time t under control $u(\cdot)$ as $\zeta(s; z, t, u(\cdot)) : [t, 0] \rightarrow \mathbb{R}^n$. ζ satisfies (1) with an initial condition almost everywhere:

$$\begin{aligned} \frac{d}{ds} \zeta(s; z, t, u(\cdot)) &= f(\zeta(s; z, t, u(\cdot)), u(s)) \\ \zeta(t; z, t, u(\cdot)) &= z \end{aligned} \quad (2)$$

Since the dynamics (1) is time-invariant, the time variables in trajectories can also be shifted by some constant² τ :

$$\zeta(s; z, t, u(\cdot)) = \zeta(s + \tau; z, t + \tau, u(\cdot)), \forall z \in \mathbb{R}^n \quad (3)$$

B. Backward Reachable Sets and Tubes

We consider two different definitions of the BRS and two different definitions of the BRT.

Intuitively, a BRS represents the set of states $z \in \mathbb{R}^n$ from which the system can be driven into some set $\mathcal{T} \subseteq \mathbb{R}^n$ at the end of a time horizon of duration $|t|$. We call \mathcal{T} the “target set”.

¹For the remainder of the paper, we will omit the notation “(s)” from variables such as z and u when referring to function values.

²In this case, it is implicit that the control function $u(\cdot)$ is also time-shifted by the same amount τ .

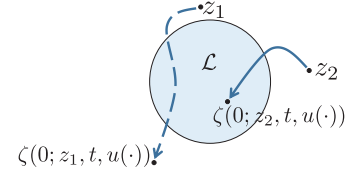


Fig. 1: The difference between a BRS and a BRT. The dashed trajectory starts at z_1 and passes through \mathcal{T} during the period $[t, 0]$, but exits \mathcal{T} by the end of the time period. Therefore the z_1 is in the BRT, but not in the BRS. The solid trajectory starting from z_2 is in \mathcal{T} at the end of the time period. Therefore, z_2 is in both the BRS and the BRT.

First we define the “Maximal BRS”; in this case the system seeks to enter \mathcal{T} using some control function. We can think of \mathcal{T} as a set of goal states. The Maximal BRS represents the set of states from which the system is guaranteed to reach \mathcal{T} . The second definition is for the “Minimal BRS”; in this case the BRS is the set of states that will lead to \mathcal{T} for all possible controls. Here we often consider \mathcal{T} to be an unsafe set such as an obstacle. The Minimal BRS represents the set of states that leads to violation of safety requirements. Formally, the two definitions of BRSs are below³:

Definition 1: Maximal BRS.

$$\mathcal{R}(t) = \{z : \exists u(\cdot) \in \mathcal{U}, \zeta(0; z, t, u(\cdot)) \in \mathcal{T}\}$$

Definition 2: Minimal BRS.

$$\mathcal{A}(t) = \{z : \forall u(\cdot) \in \mathcal{U}, \zeta(0; z, t, u(\cdot)) \in \mathcal{T}\}$$

While BRSs indicate whether a system can be driven into \mathcal{T} at the end of a time horizon, BRTs indicate whether a system can be driven into \mathcal{T} at *some time* during the time horizon of duration $|t|$. Figure 1 demonstrates the difference. BRTs are very important notions especially in safety-critical applications, in which we are interested in determining the “Minimal BRT”: the set of states that could lead to danger at some time within a specified time horizon. Formally, the two definitions of BRTs are as follows:

Definition 3: Maximal BRT.

$$\bar{\mathcal{R}}(t) = \{z : \exists u(\cdot) \in \mathcal{U}, \exists s \in [t, 0], \zeta(s; z, t, u(\cdot)) \in \mathcal{T}\}$$

Definition 4: Minimal BRT.

$$\bar{\mathcal{A}}(t) = \{z : \forall u(\cdot) \in \mathcal{U}, \exists s \in [t, 0], \zeta(s; z, t, u(\cdot)) \in \mathcal{T}\}$$

The terms “maximal” and “minimal” refer to the role of the optimal control [36]. In the maximal (or minimal) case, the control causes the BRS or BRT to contain as many (or few) states as possible – to have maximal (or minimal) size.

While BRSs and BRTs indicate sets of states of interest, from a practical implementation perspective controller synthesis based on the reachable sets is extremely important. Much of the prior work on reachable set computations also include controller synthesis, which is usually done by casting the reachability problem as an optimal control or optimization

³Sometimes in the literature, the argument of \mathcal{R} , \mathcal{A} , $\bar{\mathcal{R}}$, or $\bar{\mathcal{A}}$ is some non-negative number $\tau = -t$; however, for simplicity we will use the non-positive number t to refer to the time horizon of the BRS and BRT.

problem, often with a functional representation of BRSs and BRTs. The controller is given as decision variables in the optimization [16], [20], [21], [23], [37], [38]. We will not delve into the details of controller synthesis, since the theory we present in this paper is agnostic to these details.

III. PROBLEM FORMULATION

In this paper, we seek to obtain the BRSs and BRTs in Definitions 1 to 4 via computations in lower-dimensional subspaces under the assumption that the system (1) can be decomposed into self-contained subsystems (SCSs) (5). Such a decomposition is common, since many systems involve components that are loosely coupled. In particular, the evolution of position variables in vehicle dynamics is often weakly coupled though other variables such as heading.

A. Definitions

1) **Subsystem Dynamics:** Let the state $z \in \mathbb{R}^n$ be partitioned as $z = (z_1, z_2, z_c)$, with $z_1 \in \mathbb{R}^{n_1}$, $z_2 \in \mathbb{R}^{n_2}$, $z_c \in \mathbb{R}^{n_c}$, $n_1, n_2 > 0$, $n_c \geq 0$, $n_1 + n_2 + n_c = n$. Note that n_c could be zero. We call z_1, z_2, z_c “state partitions” of the system. Intuitively, z_1 and z_2 , are states belonging to subsystems 1 and 2, respectively, and z_c states belong to both subsystems.

Under the above notation, the system dynamics (1) become

$$\begin{aligned}\dot{z}_1 &= f_1(z_1, z_2, z_c, u) \\ \dot{z}_2 &= f_2(z_1, z_2, z_c, u) \\ \dot{z}_c &= f_c(z_1, z_2, z_c, u)\end{aligned}\quad (4)$$

In general, depending on how the dynamics f depend on u , some state partitions may be independent of the control.

We group these states into subsystems by defining the SCS states $x_1 = (z_1, z_c) \in \mathbb{R}^{n_1+n_c}$ and $x_2 = (z_2, z_c) \in \mathbb{R}^{n_2+n_c}$, where x_1 and x_2 in general share the “common” states in z_c . Note that our theory is applicable to any finite number of subsystems defined in the analogous way, with $x_i = (z_i, z_c)$; however, without loss of generality (WLOG), we assume that there are just two subsystems.

Definition 5: Self-contained subsystem. Consider the following special case of (4):

$$\begin{aligned}\dot{z}_1 &= f_1(z_1, z_c, u) \\ \dot{z}_2 &= f_2(z_2, z_c, u) \\ \dot{z}_c &= f_c(z_c, u)\end{aligned}\quad (5)$$

We call each of the subsystems with states defined as $x_i = (z_i, z_c)$ a “self-contained subsystem” (SCS), or just “subsystem” for short. Intuitively (5) means that the evolution of each subsystem depends only on the subsystem states: \dot{x}_i depends only on $x_i = (z_i, z_c)$. Explicitly, the dynamics of the two subsystems are as follows:

$$\begin{aligned}\dot{z}_1 &= f_1(z_1, z_c, u) & \dot{z}_2 &= f_2(z_2, z_c, u) \\ \dot{z}_c &= f_c(z_c, u) & \dot{z}_c &= f_c(z_c, u)\end{aligned}\quad \begin{array}{l} \text{(Subsystem 1)} \\ \text{(Subsystem 2)} \end{array}$$

Note that the two subsystems are coupled through the common state partition z_c and control u . When the subsystems are coupled through u , we say that the subsystems have “shared control”.

An example of a system that can be decomposed into SCSs is the Dubins Car with constant speed v :

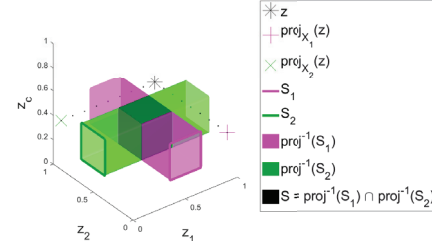


Fig. 2: This figure shows the back-projection of sets in the z_1 - z_c plane S_1 and the z_2 - z_c plane (S_2) to the 3D space to form the intersection shown as the black cube (S). The figure also shows projection of a point z onto the lower-dimensional subspaces in the z_1 - z_c and z_2 - z_c planes.

$$\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v \cos \theta \\ v \sin \theta \\ \omega \end{bmatrix}, \quad \omega \in \mathcal{U} \quad (6)$$

with state $z = (p_x, p_y, \theta)$ representing the x position, y position, and heading, and control $u = \omega$ representing the turn rate. The state partitions are simply the system states: $z_1 = p_x, z_2 = p_y, z_c = \theta$. The subsystem states x_i and the subsystem controls w_i are

$$\begin{aligned}x_1 &= \begin{bmatrix} z_1 \\ z_c \end{bmatrix} = \begin{bmatrix} p_x \\ \theta \end{bmatrix} = \begin{bmatrix} v \cos \theta \\ \omega \end{bmatrix} \\ x_2 &= \begin{bmatrix} z_2 \\ z_c \end{bmatrix} = \begin{bmatrix} p_y \\ \theta \end{bmatrix} = \begin{bmatrix} v \sin \theta \\ \omega \end{bmatrix} \\ u &= \omega\end{aligned}\quad (7)$$

where the overlapping state is θ , and the subsystem controls and their shared component is the control u itself. The control partitions u_1, u_2 do not exist, since the state partitions z_1, z_2 do not depend on the control. For more examples of systems decomposed into SCSs, see (59), (60) and other numerical examples in this paper.

Although there may be common or overlapping states in x_1 and x_2 , the evolution of each subsystem does not depend on the other explicitly. In fact, if we for example entirely ignore the subsystem x_2 , the evolution of the subsystem x_1 is well-defined and can be considered a full system on its own; hence, each subsystem is self-contained.

2) **Projection Operators:** For the projection operators, it will be helpful to refer to Fig. 2. Define the projection of a state $z = (z_1, z_2, z_c)$ onto a subsystem state space $\mathbb{R}^{n_i+n_c}$ as

$$\text{proj}_i(z) = x_i = (z_i, z_c) \quad (8)$$

This projects a point in the full dimensional state space onto a point in the subsystem state space. Also define the back-projection operator to be

$$\text{proj}^{-1}(x_i) = \{z \in \mathcal{Z} : (z_i, z_c) = x_i\} \quad (9)$$

This back-projection lifts a point from the subsystem state space to a set in the full dimensional state space. We will also need the ability to apply the back-projection operator on subsystems set to full dimensional sets. In this case, we overload the back-projection operator:

$$\text{proj}^{-1}(S_i) = \{z \in \mathcal{Z} : \exists x_i \in S_i, (z_i, z_c) = x_i\} \quad (10)$$

3) **Subsystem Trajectories:** Since each subsystem in (5) is self-contained, we can denote the subsystem trajectories $\xi_i(s; x_i, t, u(\cdot))$. When needed, we will write the subsystem trajectories more explicitly in terms of the state partitions as $\xi_i(s; z_i, z_c, t, u(\cdot))$. The subsystem trajectories satisfy the subsystem dynamics and initial condition:

$$\begin{aligned} \frac{d}{ds} \xi_i(s; x_i, t, u(\cdot)) &= g_i(\xi_i(s; x_i, t, u(s))) \\ \xi_i(t; x_i, t, u(\cdot)) &= x_i \end{aligned} \quad (11)$$

where $g_i(x_i, u) = (f_i(z_i, z_c, u), f_c(z_c, u))$, and the full system trajectory and subsystem trajectories are simply related to each other via the projection operator:

$$\text{proj}_i(\zeta(s; z, t, u(\cdot))) = \xi_i(s; x_i, t, u(\cdot)) \quad (12)$$

where $x_i = \text{proj}_i(z)$.

B. Goals of This Paper

We assume that the full system target set \mathcal{T} can be written in terms of the subsystem target sets $\mathcal{T}_1 \subseteq \mathcal{X}_1, \mathcal{T}_2 \subseteq \mathcal{X}_2$ in one of the following ways:

$$\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2) \quad (13)$$

where the full target set is the intersection of the back-projections of subsystem target sets, or

$$\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2) \quad (14)$$

where the full target set is the union of the back-projections of subsystem target sets. Fig. 2 helps provide intuition for these concepts: applying (13) to \mathcal{S}_1 and \mathcal{S}_2 results in the black cube. Applying (14) would result in the cross-shaped set encompassing both $\text{proj}^{-1}(\mathcal{S}_1)$ and $\text{proj}^{-1}(\mathcal{S}_2)$.

In practice, this is not a strong assumption, since \mathcal{T}_1 and \mathcal{T}_2 share the common variables z_c . Relatively complex shapes, for example those in Fig. 4 and 6, can be represented by an intersection or union of back-projections of lower-dimensional sets that share common variables. In addition, such an assumption is reasonable since the full system target set should at least be representable in some way in the lower-dimensional spaces.

Next, we define the subsystem BRSs $\mathcal{R}_i, \mathcal{A}_i$ the same way as in Definitions 1 and 2, but with the subsystems in (5) and subsystem target sets $\mathcal{T}_i, i = 1, 2$, respectively:

$$\begin{aligned} \mathcal{R}_i(t) &= \{x_i : \exists u(\cdot), \xi_i(0; x_i, t, u(\cdot)) \in \mathcal{T}_i\} \\ \mathcal{A}_i(t) &= \{x_i : \forall u(\cdot), \xi_i(0; x_i, t, u(\cdot)) \in \mathcal{T}_i\} \end{aligned} \quad (15)$$

Subsystem BRTs are defined analogously:

$$\begin{aligned} \bar{\mathcal{R}}_i(t) &= \{x_i : \exists u(\cdot), \exists s \in [t, 0], \xi_i(s; x_i, t, u(\cdot)) \in \mathcal{T}_i\} \\ \bar{\mathcal{A}}_i(t) &= \{x_i : \forall u(\cdot), \exists s \in [t, 0], \xi_i(s; x_i, t, u(\cdot)) \in \mathcal{T}_i\} \end{aligned} \quad (16)$$

Given a system in the form of (5) with target set that can be represented by (13) or (14), our goals are as follows.

- **Decomposition of BRSs.** First, we would like to compute full-dimensional BRSs by performing computations in lower-dimensional subspaces. Specifically, we would like to first compute the subsystem BRSs $\mathcal{R}_i(t)$ or $\mathcal{A}_i(t)$, and then reconstruct the full system BRS $\mathcal{R}(t)$ or $\mathcal{A}(t)$. This process greatly reduces computation burden by decomposing the full system into two lower-dimensional subsystems. Formally,

we would like to investigate the situations in which the following four cases is true:

$$\begin{aligned} (13) &\Rightarrow \mathcal{R}(t) = \text{proj}^{-1}(\mathcal{R}_1(t)) \cap \text{proj}^{-1}(\mathcal{R}_2(t)) \\ (13) &\Rightarrow \mathcal{A}(t) = \text{proj}^{-1}(\mathcal{A}_1(t)) \cap \text{proj}^{-1}(\mathcal{A}_2(t)) \\ (14) &\Rightarrow \mathcal{R}(t) = \text{proj}^{-1}(\mathcal{R}_1(t)) \cup \text{proj}^{-1}(\mathcal{R}_2(t)) \\ (14) &\Rightarrow \mathcal{A}(t) = \text{proj}^{-1}(\mathcal{A}_1(t)) \cup \text{proj}^{-1}(\mathcal{A}_2(t)) \end{aligned} \quad (17)$$

Results related to BRSs are outlined for SCSs in Theorems 1 and 2. In the case that the subsystem controls do not share any components, Propositions 1 and 2 state stronger results.

- **Decomposition of BRTs.** BRTs are useful since they provide guarantees over a time horizon as opposed to at a particular time. However, often BRTs cannot be decomposed the same way as BRSs. Therefore, our second goal is to propose how BRTs can be decomposed. These results are stated in Propositions 3 and 4, and Theorem 3.
- **Treatment of disturbances.** Finally, we investigate how the above theoretical results change in the presence of disturbances. In Section VIII, we will show that slightly conservative BRSs and BRTs can still be obtained using our decomposition technique.

Tables I, II, and III summarize our theoretical results and where details of each result can be found.

IV. SELF-CONTAINED SUBSYSTEMS

Suppose the full system (1) can be decomposed into SCSs in (5), then the full-dimensional BRS can be reconstructed without incurring additional approximation errors from lower-dimensional BRSs in situations stated in Theorem 1 and 2.

Remark 1: If \mathcal{T} represents states the system aims to reach, then $\mathcal{R}(t)$ represents the set of states from which \mathcal{T} can be reached. If the system goal states are the union of subsystem goal states, then it suffices for any subsystem to reach its subsystem goal states, regardless of any coupling that exists between the subsystems. Theorem 1 states this intuitive result.

Theorem 1: Suppose that the full system in (1) can be decomposed into the form of (5), then

$$\begin{aligned} \mathcal{T} &= \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \mathcal{R}(t) &= \text{proj}^{-1}(\mathcal{R}_1(t)) \cup \text{proj}^{-1}(\mathcal{R}_2(t)) \end{aligned} \quad (18)$$

Remark 2: If \mathcal{T} represents the set of unsafe states, then $\mathcal{A}(t)$ is the set of states from which the system will be driven into danger. Thus outside of $\mathcal{A}(t)$, there exists a control for the system to avoid the unsafe states. For the system to avoid \mathcal{T} , it suffices to avoid the unsafe states in either subsystem, regardless of any coupling that exists between the subsystems. Theorem 2 formally states this intuitive result.

Theorem 2: Suppose that the full system in (1) can be decomposed into the form of (5), then

$$\begin{aligned} \mathcal{T} &= \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \mathcal{A}(t) &= \text{proj}^{-1}(\mathcal{A}_1(t)) \cap \text{proj}^{-1}(\mathcal{A}_2(t)) \end{aligned} \quad (19)$$

These two theorems imply that the optimal closed-loop state feedback controller is decomposable: at any given (full) state, the optimal control depends only on either of the subsystem states, but not the full state. Therefore, BRSs for a system using a pre-specified closed-loop controller can be decomposed if the control depends only on subsystem states at any given

TABLE I: Backward Reachable Set Decomposition

Section	IV		V		VIII-A		VIII-B	
Shared Controls	Yes		No		Yes		No	
Shared Disturbance	No		No		Yes		Yes	
Target	Intersection	Union	Intersection	Union	Intersection	Union	Intersection	Union
Recover Max. BRS?	No	Yes, exact	Yes, exact	Yes, exact	No	Yes, consrv	Yes, consrv	Yes, consrv
Recover Min. BRS?	Yes, exact	No	Yes, exact	Yes, exact	Yes, consrv	No	Yes, consrv	Yes, consrv
Locations & Equation(s)	Thm 2, (19)	Thm 1, (18)	Prop 1, (34) Thm 2, (19)	Thm 1, (18) Prop 2, (35)	Cor 4, (66)	Cor 3, (64)	Cor 5, (67)	Cor 6, (69)

Summary of possible decompositions of the BRS, whether they are possible, and if so whether they are exact or conservative. Exact means that no additional approximation errors are introduced. Note that in the cases marked “no” for shared control (or shared disturbance), the results hold for both decoupled control (or disturbance) and for no control (or disturbance). All cases shown are for scenarios with shared states, with the shared states being z_c in (5); in the case that there are no shared states this becomes a straightforward decoupled system.

TABLE II: BRT Results for Reconstruction from Tubes

Section	VI-A				VIII-C			
Shared Controls	Yes		No		Yes		No	
Shared Disturbance	No		No		Yes		Yes	
Target	Intersection	Union	Intersection	Union	Intersection	Union	Intersection	Union
Recover Max. BRT?	No	Yes, exact	No	Yes, exact	No	Yes, conserv	No	Yes, conserv
Recover Min. BRT?	No	No	No	Yes, exact	No	No	No	Yes, conserv
Equation(s)	N/A	Prop 3, (40)	N/A	Prop 3, (40) Prop 3, (41)	N/A	Cor 7, (70)	N/A	Cor 7, (70) Cor 8, (71)

Summary of possible decompositions of the BRT, whether they are possible, and if so whether they are exact or conservative. Exact means that no additional approximation errors are introduced. Note that in the cases marked “no” for shared control (or shared disturbance), the results hold for both decoupled control (or disturbance) and for no control (or disturbance). All cases shown are for scenarios with shared states, with the shared states being z_c in (5); in the case that there are no shared states this becomes a straightforward decoupled system.

TABLE III: BRT Results for Reconstruction from Sets

Section	VI-B	VIII-C
Disturbance	No	Yes
Recover Max. BRT?	Yes, exact	Yes, conserv
Recover Min. BRT?	Yes, exact*	Yes, exact*
Equation(s)	Prop 4, (42) Thm 3, (44)	Cor 9, (72) Thm 3, (44)

Summary of computation of the BRT via union of BRSSs, whether they are exact or conservative. Exact means that no additional approximation errors are introduced. These results can be combined with those of Table I to, for the negative cases in Table II, efficiently compute BRTs by first computing BRSSs via decomposition and then taking the union of BRSSs to obtain the BRT. “*” indicates that the solution here can be found only if the minimum BRSSs are non-empty for the entire time period.

(full) state. It is also interesting to note that BRSSs for a system using a pre-specified open-loop controller can be decomposed, since augmenting the state space with $\dot{t} = 1$ would maintain the structure of self-contained subsystems. The additional computational burden from the state augmentation can be eliminated through a time-varying formulation of reachability, for example, in reference [28].

To prove the theorems, we need some intermediate results.

Lemma 1: Let $\bar{z} \in \mathcal{Z}$, $\bar{x}_i = \text{proj}_i(\bar{z})$, $\mathcal{S}_i \subseteq \mathcal{X}_i$. Then,

$$\bar{x}_i \in \mathcal{S}_i \Leftrightarrow \bar{z} \in \text{proj}^{-1}(\mathcal{S}_i) \quad (20)$$

Proof of Lemma 1: Forward direction: Suppose $\bar{x}_i \in \mathcal{S}_i$, then trivially $\exists x_i \in \mathcal{S}_i$, $\text{proj}_i(\bar{z}) = x_i$. By the definition of back-projection in (10), we have $\bar{z} \in \text{proj}^{-1}(\mathcal{S}_i)$.

Backward direction: Suppose $\bar{z} \in \text{proj}^{-1}(\mathcal{S}_i)$, then by (10) we have $\exists x_i \in \mathcal{S}_i$, $\text{proj}_i(\bar{z}) = x_i$. Denote such an x_i to be \hat{x}_i , and suppose $\bar{x}_i \notin \mathcal{S}_i$. Then, we have $\hat{x}_i \neq \bar{x}_i$, a contradiction, since $\bar{x}_i = \text{proj}_i(\bar{z}) = \hat{x}_i$. ■

Corollary 1: If $\mathcal{S} = \text{proj}^{-1}(\mathcal{S}_1) \cup \text{proj}^{-1}(\mathcal{S}_2)$, then

$$\bar{z} \in \mathcal{S} \Leftrightarrow \bar{x}_1 \in \mathcal{S}_1 \vee \bar{x}_2 \in \mathcal{S}_2, \text{ where } \bar{x}_i = \text{proj}_i(\bar{z})$$

Corollary 2: If $\mathcal{S} = \text{proj}^{-1}(\mathcal{S}_1) \cap \text{proj}^{-1}(\mathcal{S}_2)$, then

$$\bar{z} \in \mathcal{S} \Leftrightarrow \bar{x}_1 \in \mathcal{S}_1 \wedge \bar{x}_2 \in \mathcal{S}_2, \text{ where } \bar{x}_i = \text{proj}_i(\bar{z})$$

A. Proof of Theorem 1

We will prove the following equivalent statement:

$$\bar{z} \in \mathcal{R}(t) \Leftrightarrow \bar{z} \in \text{proj}^{-1}(\mathcal{R}_1(t)) \cup \text{proj}^{-1}(\mathcal{R}_2(t)) \quad (21)$$

Consider the relationship between the full system trajectory and subsystem trajectory in (12). Define $\bar{x}_i = \text{proj}_i(\bar{z})$ and $\xi_i(0; \bar{x}_i, t, u(\cdot)) = \text{proj}_i(\zeta(0; \bar{z}, t, u(\cdot)))$.

Backward direction: By Corollary 1, (21) is equivalent to

$$\bar{x}_1 \in \mathcal{R}_1(t) \vee \bar{x}_2 \in \mathcal{R}_2(t) \quad (22)$$

WLOG, assume $\bar{x}_1 \in \mathcal{R}_1(t)$. By the subsystem BRS definition in (15), this is equivalent to

$$\exists u(\cdot), \xi_1(0; \bar{x}_1, t, u(\cdot)) \in \mathcal{T}_1 \quad (23)$$

By Lemma 1, we equivalently have $\bar{z} \in \text{proj}^{-1}(\mathcal{R}_1(t))$. This proves the backward direction.

Forward direction: We begin with $\bar{z} \in \mathcal{R}(t)$, which by Definition 1 is equivalent to $\exists u(\cdot), \zeta(0; \bar{z}, t, u(\cdot)) \in \mathcal{T}$. By Corollary 1, we then have

$$\exists u(\cdot), \xi_1(0; \bar{x}_1, t, u(\cdot)) \in \mathcal{T}_1 \vee \xi_2(0; \bar{x}_2, t, u(\cdot)) \in \mathcal{T}_2 \quad (24)$$

Finally, distributing “ $\exists u(\cdot)$ ” gives (22). ■

B. Proof of Theorem 2

We will prove the following equivalent statement:

$$\bar{z} \notin \mathcal{A}(t) \Leftrightarrow \bar{z} \notin \text{proj}^{-1}(\mathcal{A}_1(t)) \cap \text{proj}^{-1}(\mathcal{A}_2(t)) \quad (25)$$

The above statement is equivalent to

$$\bar{z} \in \mathcal{A}^c(t) \Leftrightarrow \bar{z} \in [\text{proj}^{-1}(\mathcal{A}_1(t))]^c \cup [\text{proj}^{-1}(\mathcal{A}_2(t))]^c \quad (26)$$

By the Definition 2 (minimal BRS), we have that $\bar{z} \in \mathcal{A}^c(t)$ is equivalent to $\exists u(\cdot) \in \mathbb{U}$, $\zeta(0; \bar{z}, t, u(\cdot)) \in \mathcal{T}^c$. Also,

$$\bar{z} \in [\text{proj}^{-1}(\mathcal{A}_1(t))]^c \cup [\text{proj}^{-1}(\mathcal{A}_2(t))]^c$$

is equivalent to $\bar{x}_1 \in \mathcal{A}_1^c(t) \vee \bar{x}_2 \in \mathcal{A}_2^c(t)$.

From here, we can proceed in the same fashion as the proof of Theorem 1, with “ $\mathcal{R}(t)$ ” replaced with “ $\mathcal{A}^c(t)$ ”,

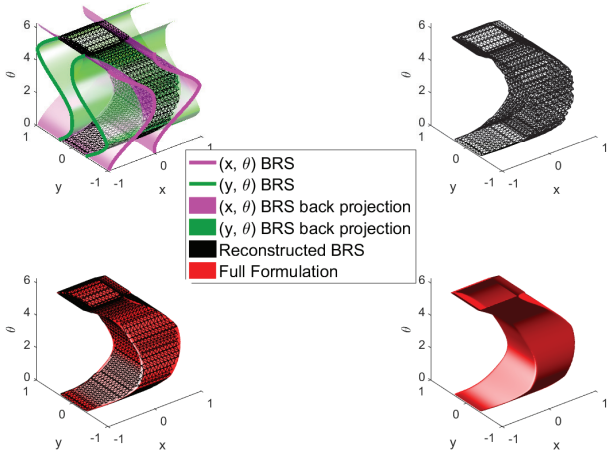


Fig. 3: Comparison of the Dubins Car BRS $\mathcal{A}(t = -0.5)$ computed using the full formulation and via decomposition. Left top: BRSs in the lower-dimensional subspaces and how they are combined to form the full-dimensional BRS. Top right: BRS computed via decomposition. Bottom left: BRSs computed using both methods, superimposed, showing that they are indistinguishable. Bottom right: BRS computed using the full formulation.

“ $\text{proj}^{-1}(\mathcal{R}_i(t))$ ” replaced with “ $[\text{proj}^{-1}(\mathcal{A}_i(t))]^c$ ”, and “ \mathcal{T} ”, “ \mathcal{T}_i ” replaced with “ \mathcal{T}^c ”, “ \mathcal{T}_i^c ”, respectively. ■

The conditions for reconstruction the maximal BRS for an intersection of targets, as well as the minimal BRS for a union of targets, are more complicated and beyond the scope of this paper. A summary of the results from this section can be seen in Table I under Section IV.

C. Numerical Example: The Dubins Car

The Dubins Car is a well-known system whose dynamics are given by (6). This system is only 3D, and its BRS can be tractably computed in the full-dimensional space, so we use it to compare the full formulation with our decomposition method. The Dubins Car dynamics can be decomposed according to (7). For this example, we computed the BRS from the target set representing positions near the origin in both the p_x and p_y dimensions:

$$\mathcal{T} = \{(p_x, p_y, \theta) : |p_x|, |p_y| \leq 0.5\} \quad (27)$$

Such a target set \mathcal{T} can be used to model an obstacle that the vehicle must avoid. Given \mathcal{T} , the interpretation of the BRS $\mathcal{A}(t)$ is the set of states from which a collision with the obstacle may occur after a duration of $|t|$. From \mathcal{T} , we computed the BRS $\mathcal{A}(t)$ at $t = -0.5$. The resulting full formulation BRS is shown in Fig. 3 as the red surface which appears in the bottom subplots. To compute the BRS using our decomposition method, we write the unsafe set \mathcal{T} as

$$\begin{aligned} \mathcal{T}_1 &= \{(p_x, \theta) : |p_x| \leq 0.5\}, \mathcal{T}_2 = \{(p_y, \theta) : |p_y| \leq 0.5\} \\ \mathcal{T} &= \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2) \end{aligned} \quad (28)$$

From \mathcal{T}_1 and \mathcal{T}_2 , we computed the lower-dimensional BRSs $\mathcal{A}_1(t)$ and $\mathcal{A}_2(t)$, and then reconstructed the full-dimensional BRS $\mathcal{A}(t)$ using Theorem 2: $\mathcal{A}(t) = \text{proj}^{-1}(\mathcal{A}_1(t)) \cap \text{proj}^{-1}(\mathcal{A}_2(t))$. The subsystem BRSs and their back-projections are shown in magenta and green in the top left

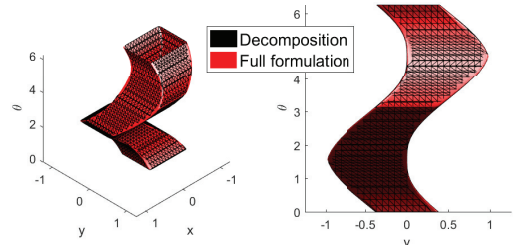


Fig. 4: The Dubins Car BRS $\mathcal{A}(t = -0.5)$ computed using the full formulation and via decomposition, other view angles.

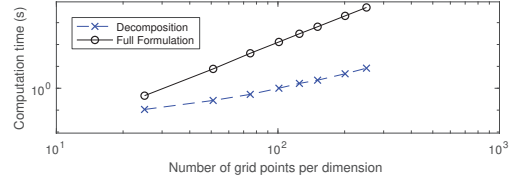


Fig. 5: Computation times of the two methods in log scale for the Dubins Car. The time of the direct computation in 3D increases rapidly with the number of grid points per dimension. In contrast, computation times in 2D with decomposition are negligible in comparison.

subplot of Fig. 3. The reconstructed BRS is shown in the top left, top right, and bottom left subplots of Fig. 3 (black mesh).

In the bottom left subplot of Fig. 3, we superimpose the full-dimensional BRS computed using the two methods. We show the comparison of the computation results viewed from two different angles in Fig. 4. The results are indistinguishable.

Theorem 2 allows the computation to be performed in lower-dimensional subspaces, which is significantly faster. Another benefit of the decomposition method is that in the numerical methods for solving the HJ PDE, the amount of numerical dissipation increases with the number of state dimensions. Thus, computations in lower-dimensional subspaces lead to a slightly more accurate numerical solution.

The computation benefits of using our decomposition method can be seen from Fig. 5. The plot shows, in log-log scale, the computation time in seconds versus the number of grid points per dimension in the numerical computation. One can see that the direct computation of the BRS in 3D becomes very time-consuming as the number of grid points per dimension is increased, while the computation via decomposition hardly takes any time in comparison. Directly computing the BRS with 251 grid points per dimension in 3D took approximately 80 minutes, while computing the BRS via decomposition in 2D only took approximately 30 seconds! The computations were timed on a computer with an Intel Core i7-2600K processor and 16GB of random-access memory.

Fig. 6 illustrates Theorem 1. We chose the target set to be $\mathcal{T} = \{(p_x, p_y, \theta) : p_x \leq 0.5 \vee p_y \leq 0.5\}$, and computed the BRS $\mathcal{R}(t), t = -0.5$ via decomposition. No additional approximation error is incurred in the reconstruction process. The target set can be written as $\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_1)$ where $\mathcal{T}_1 = \{(p_x, \theta) : p_x \leq 0.5\}, \mathcal{T}_2 = \{(p_y, \theta) : p_y \leq 0.5\}$.

V. SCSSS WITH DECOUPLED CONTROL

In this section, we consider a special case of (5) in which the subsystem controls do not have any shared components.

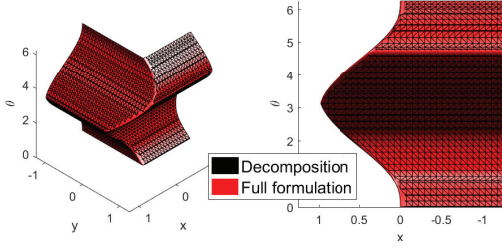


Fig. 6: Comparison of the $\mathcal{R}(t)$ computed using our decomposition method and the full formulation. The computation results are indistinguishable. Note that the surface shows the boundary of the set; the set itself is on the “near” side of the left subplot, and the left side of the right subplot.

The results from IV still hold, and *in addition* we can state the results in Propositions 1 and 2. The special case of (5) is as follows:

$$\begin{aligned}\dot{z}_1 &= f_1(z_1, z_c, u_1) \\ \dot{z}_2 &= f_2(z_2, z_c, u_2) \\ \dot{z}_c &= f_c(z_c)\end{aligned}\quad (29)$$

where the subsystem controls do not have any shared components, so that we have $u = (u_1, u_2)$. Furthermore, we can define the trajectory of z_c as $\eta(s; z_c, t)$, which satisfies

$$\begin{aligned}\frac{d}{ds}\eta_c(s; z_c, t) &= f_c(\eta_c(s; z_c, t)) \\ \eta_c(t; z_c, t) &= z_c\end{aligned}\quad (30)$$

Note that since the trajectory $\eta_c(s; z_c, t)$ does not depend on the control, we can treat $\eta_c(s; z_c, t)$ as a constant when given z_c and s . Therefore, given z_c , the other *state partitions* also become self-contained, with dynamics

$$\dot{z}_i = f_i(z_i, z_c, u_i) = f_i(z_i, u_i; \eta_c(s; z_c, t)) \quad (31)$$

and with trajectories $\eta_i(s; z_i, z_c, t, u_i(\cdot))$ satisfying

$$\begin{aligned}\frac{d}{ds}\eta_i(s; z_i, z_c, t, u_i(\cdot)) &= f_i(\eta_i(s; z_i, z_c, t, u_i(\cdot)), u_i(s); \eta_c(s; z_c, t)) \\ \eta_i(t; z_i, z_c, t, u_i(\cdot)) &= z_i\end{aligned}\quad (32)$$

Therefore, the *subsystem trajectories* can be written as

$$\xi_i(s; x_i, t, u_i(\cdot)) = (\eta_i(s; z_i, z_c, t, u_i(\cdot)), \eta_c(s; z_c, t)) \quad (33)$$

Proposition 1: Suppose that the full system in (1) can be decomposed into the form of (29). Then,

$$\begin{aligned}\mathcal{T} &= \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \mathcal{R}(t) &= \text{proj}^{-1}(\mathcal{R}_1(t)) \cap \text{proj}^{-1}(\mathcal{R}_2(t))\end{aligned}\quad (34)$$

Proposition 2: Suppose that the full system in (1) can be decomposed into the form of (29). Then,

$$\begin{aligned}\mathcal{T} &= \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \mathcal{A}(t) &= \text{proj}^{-1}(\mathcal{A}_1(t)) \cup \text{proj}^{-1}(\mathcal{A}_2(t))\end{aligned}\quad (35)$$

Remark 3: Systems with fully decoupled subsystems in the form of $x_1 = z_1, x_2 = z_2$ are a special case of (29).

A. Proof of Proposition 1

We will prove the following equivalent statement:

$$\bar{z} \in \mathcal{R}(t) \Leftrightarrow \bar{z} \in \text{proj}^{-1}(\mathcal{R}_1(t)) \cap \text{proj}^{-1}(\mathcal{R}_2(t)) \quad (36)$$

By Definition 1 (maximal BRS), we have

$$\bar{z} \in \mathcal{R}(t) \Leftrightarrow \exists u(\cdot), \zeta(0; \bar{z}, t, u(\cdot)) \in \mathcal{T} \quad (37)$$

Consider the relationship between the full system trajectory and subsystem trajectory in (12). Define

$$\begin{aligned}\bar{x}_i &= (\bar{z}_i, \bar{z}_c) = \text{proj}_i(\bar{z}), \text{ and} \\ \xi_i(s; \bar{x}_i, t, u_i(\cdot)) &= \text{proj}_i(\zeta(0; \bar{z}, t, u(\cdot)))\end{aligned}$$

By (33) we can write

$$(\eta_i(s; \bar{z}_i, \bar{z}_c, t, u_i(\cdot)), \eta_c(s; \bar{z}_c, t)) = \text{proj}_i(\zeta(0; \bar{z}, t, u(\cdot)))$$

Since $\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2)$, (37) is equivalent to

$$\begin{aligned}\exists (u_1(\cdot), u_2(\cdot)), \\ (\eta_1(s; \bar{z}_1, \bar{z}_c, t, u_1(\cdot)), \eta_c(s; \bar{z}_c, t)) &\in \mathcal{T}_1 \wedge \\ (\eta_2(s; \bar{z}_2, \bar{z}_c, t, u_2(\cdot)), \eta_c(s; \bar{z}_c, t)) &\in \mathcal{T}_2 \\ &\text{(by Corollary 2)}\end{aligned}\quad (38)$$

$$\begin{aligned}\Leftrightarrow \exists u_1(\cdot), (\eta_1(s; \bar{z}_1, \bar{z}_c, t, u_1(\cdot)), \eta_c(s; \bar{z}_c, t)) &\in \mathcal{T}_1 \wedge \\ \exists u_2(\cdot), (\eta_2(s; \bar{z}_2, \bar{z}_c, t, u_2(\cdot)), \eta_c(s; \bar{z}_c, t)) &\in \mathcal{T}_2 \\ &\text{(since subsystem controls do not share components)}\end{aligned}\quad (39)$$

$$\begin{aligned}\Leftrightarrow x_1 \in \mathcal{R}_1(t) \wedge x_2 \in \mathcal{R}_2(t) \\ &\text{(by definition of subsystem BRS in (15))}\end{aligned}$$

$$\begin{aligned}\Leftrightarrow \bar{z} \in \text{proj}^{-1}(\mathcal{R}_1(t)) \cap \text{proj}^{-1}(\mathcal{R}_2(t)) \\ &\text{(by Corollary 2)}\end{aligned}$$

B. Proof of Proposition 2

This proof follows the same arguments as 1, but with “ \mathcal{R} ”, “ \cap ”, “ \exists ” replaced with “ \mathcal{A} ”, “ \cup ”, “ \forall ”, respectively. ■

Remark 4: When the subsystem controls have shared components, the control chosen by each subsystem may not agree with the other. This is the intuition behind why the results of Propositions 1 and 2 only hold true when there are no shared components in the subsystem controls. Note that the theorems hold despite the state coupling between the subsystems.

A summary of the results from this section can be seen in Table I under Section V.

VI. DECOMPOSITION OF REACHABLE TUBES

Sometimes, BRTs are desired; for example, in safety analysis, the computation of the BRT $\bar{\mathcal{A}}(t)$ in Definition 2 is quite important, since if the target set \mathcal{T} represents an unsafe set of states, then $\bar{\mathcal{A}}(t)$ contains all states that would lead to some unsafe state at *some time* within a duration of length $|t|$.

We now first discuss a special case where the full system BRT can be directly reconstructed from subsystem BRTs in Section VI-A, and then present a general method in which a BRT can be obtained via the union of BRSs in Section VI-B.

A. Full System BRTs From Subsystem BRTs

Intuitively, it may seem like the results related to BRSs outlined in Sections IV and V trivially carry over to BRTs, and the relationship between BRSs and BRTs are relatively simple; however, this is only partially true. The results related to BRSs presented so far in this paper only easily carry over for BRTs if $\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2)$. This is formally stated in the following Proposition:

Proposition 3: Suppose (14) holds, that is,

$$\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2)$$

Then, the full-dimensional BRT can be reconstructed from the lower-dimensional BRTs without incurring additional approximation errors. For systems with SCSs as in (5), we have

$$\bar{\mathcal{R}}(t) = \text{proj}^{-1}(\bar{\mathcal{R}}_1(t)) \cup \text{proj}^{-1}(\bar{\mathcal{R}}_2(t)) \quad (40)$$

This is proven by starting the proof of Theorem 1 with Definition 3 for the BRT $\bar{\mathcal{R}}(t)$ instead of Definition 1 for the BRS $\mathcal{R}(t)$.

For systems with subsystem controls that do not share any components, we *in addition* have

$$\bar{\mathcal{A}}(t) = \text{proj}^{-1}(\bar{\mathcal{A}}_1(t)) \cup \text{proj}^{-1}(\bar{\mathcal{A}}_2(t)) \quad (41)$$

This can be proven by starting the proof of Proposition 2 from Definition 4 for the BRT $\bar{\mathcal{A}}(t)$ instead of Definition 2 for the BRS $\mathcal{A}(t)$. Note that (41) does not necessarily hold for systems with shared controls.

B. BRTs From Union of BRSs

If $\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2)$, the BRT cannot be directly reconstructed from lower-dimensional BRTs because when computing BRTs, we lose information about the exact time that a trajectory enters a set. Instead, we provide an alternative method of obtaining the BRT: First compute BRSs, and then take their union over time. We show that $\bar{\mathcal{R}}(t) = \bigcup_{s \in [t, 0]} \mathcal{R}(s)$, and $\bar{\mathcal{A}}(t) = \bigcup_{s \in [t, 0]} \mathcal{A}(s)$ when $\mathcal{A}(s) \neq \emptyset \forall s \in [t, 0]$. These results related to the indirect reconstruction of BRTs are given in Proposition 4 and Theorem 3.

This method is independent of how the BRSs are obtained and whether there is sharing of control components among subsystems, since the union is taken over time of BRSs of the same dimensionality as the BRT. To take advantage of decomposition, one would compute BRSs via decomposition for the cases outlined in Sections IV and V, and then take the union over time of the BRSs to obtain the BRT.

Note that, as mentioned, this alternative method of computing the BRT is needed for the case where $\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2)$; however, it is also applicable for the cases presented in Section VI-A, where $\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2)$, as long as the BRSs can be computed via decomposition; thus there is overlap of applicability between the two BRT computation methods we present.

Proposition 4:

$$\bigcup_{s \in [t, 0]} \mathcal{R}(s) = \bar{\mathcal{R}}(t) \quad (42)$$

Theorem 3:

$$\bigcup_{s \in [t, 0]} \mathcal{A}(s) \subseteq \bar{\mathcal{A}}(t) \quad (43)$$

In addition, if $\forall s \in [t, 0], \mathcal{A}(s) \neq \emptyset$, then

$$\bigcup_{s \in [t, 0]} \mathcal{A}(s) = \bar{\mathcal{A}}(t). \quad (44)$$

Propositions 4 and the first part of Theorem 3 are known [36], but we present them in our paper in greater detail for clarity and completeness. The second part of Theorem 3 is the main new result related to obtaining the BRT from BRSs.

Remark 5: The reason the theorems in Sections IV and V trivially carry over when $\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2)$ is that in this case, any subsystem trajectory that reaches the corresponding subsystem target set implies that the full system trajectory reaches the full system target set.

In contrast, in the case $\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2)$, both subsystem trajectories must be in the corresponding subsystem target sets *at the same time*. Mathematically, recall the definitions of subsystem BRTs in (16):

$$\bar{\mathcal{A}}_i(t) = \{x_i : \forall u(\cdot), \exists s \in [t, 0], \xi_i(s; x_i, t, u(\cdot)) \in \mathcal{T}_i\}$$

$$\bar{\mathcal{R}}_i(t) = \{x_i : \exists u(\cdot), \exists s \in [t, 0], \xi_i(s; x_i, t, u(\cdot)) \in \mathcal{T}_i\}$$

The set of “ s ” during which each subsystem trajectory is in \mathcal{T}_i may not overlap for the different subsystems. In this case, we can still first compute the BRSs in lower-dimensional subspaces, and then convert the BRSs to the BRT using Propositions 4 and Theorem 3.

C. Proof of Proposition 4

We start with Definition 1 (maximal BRS):

$$\mathcal{R}(t) = \{z : \exists u(\cdot) \in \mathbb{U}, \zeta(0; z, t, u(\cdot)) \in \mathcal{T}\}$$

If some state z is in the union $\bigcup_{s \in [t, 0]} \mathcal{R}(s)$, then there is some $s \in [t, 0]$ such that $z \in \mathcal{R}(s)$. Therefore, the union can be written as

$$\bigcup_{s \in [t, 0]} \mathcal{R}(s) = \{z : \exists s \in [t, 0], \exists u(\cdot), \zeta(0; z, s, u(\cdot)) \in \mathcal{T}\} \quad (45)$$

Suppose $z \in \bigcup_{s \in [t, 0]} \mathcal{R}(s)$, then equivalently

$$\exists s \in [t, 0], \exists u(\cdot) \in \mathbb{U}, \zeta(0; z, s, u(\cdot)) \in \mathcal{T} \quad (46)$$

Using (3), the time-invariance of the system, we can shift the trajectory time arguments by $t - s$ to get

$$\exists s \in [t, 0], \exists u(\cdot) \in \mathbb{U}, \zeta(t - s; z, t, u(\cdot)) \in \mathcal{T} \quad (47)$$

Since $s \in [t, 0] \Leftrightarrow t - s \in [t, 0]$, we can equivalently write

$$\exists s \in [t, 0], \exists u(\cdot) \in \mathbb{U}, \zeta(s; z, t, u(\cdot)) \in \mathcal{T} \quad (48)$$

We can swap the expressions $\exists s \in [t, 0]$ and $\exists u(\cdot) \in \mathbb{U}$ without changing meaning since both quantifiers are the same:

$$\exists u(\cdot) \in \mathbb{U}, \exists s \in [t, 0], \zeta(s; z, t, u(\cdot)) \in \mathcal{T} \quad (49)$$

which is equivalent to $z \in \bar{\mathcal{R}}(t)$ by Definition 3 (maximal BRT). ■

D. Proof of Theorem 3

We first establish $\bigcup_{s \in [t, 0]} \mathcal{A}(s) \subseteq \bar{\mathcal{A}}(t)$. Consider Definition 2 (minimal BRS):

$$\mathcal{A}(t) = \{z : \forall u(\cdot) \in \mathbb{U}, \zeta(0; z, t, u(\cdot)) \in \mathcal{T}\}$$

If some state z is in the union $\bigcup_{s \in [t, 0]} \mathcal{A}(s)$, then $\exists s \in [t, 0]$ such that $z \in \mathcal{A}(s)$. Thus, the union can be written as

$$\bigcup_{s \in [t, 0]} \mathcal{A}(s) = \{z : \exists s \in [t, 0], \forall u(\cdot), \zeta(0; z, s, u(\cdot)) \in \mathcal{T}\} \quad (50)$$

Suppose $z \in \bigcup_{s \in [t, 0]} \mathcal{A}(s)$, then

$$\exists s \in [t, 0], \forall u(\cdot) \in \mathbb{U}, \zeta(0; z, s, u(\cdot)) \in \mathcal{T} \quad (51)$$

Using (3), the time-invariance of the system, we can shift the trajectory time arguments by $t - s$ to get

$$\exists s \in [t, 0], \forall u(\cdot) \in \mathbb{U}, \zeta(t - s; z, t, u(\cdot)) \in \mathcal{T} \quad (52)$$

Since $s \in [t, 0] \Leftrightarrow t - s \in [t, 0]$, we can equivalently write

$$\exists s \in [t, 0], \forall u(\cdot) \in \mathbb{U}, \zeta(s; z, t, u(\cdot)) \in \mathcal{T} \quad (53)$$

Let such an $s \in [t, 0]$ be denoted \bar{s} , then

$$\begin{aligned} \forall u(\cdot) \in \mathbb{U}, \zeta(\bar{s}; z, t, u(\cdot)) &\in \mathcal{T} \\ \Rightarrow \forall u(\cdot) \in \mathbb{U}, \exists s \in [t, 0], \zeta(s; z, t, u(\cdot)) &\in \mathcal{T} \end{aligned} \quad (54)$$

By Definition 4, we have $z \in \bar{\mathcal{A}}(t)$.

Next, given $\forall s \in [t, 0], \mathcal{A}(s) \neq \emptyset$, we show $\bigcup_{s \in [t, 0]} \mathcal{A}(s) \supseteq \bar{\mathcal{A}}(t)$. Equivalently, we show

$$z \notin \bigcup_{s \in [t, 0]} \mathcal{A}(s) \Rightarrow z \notin \bar{\mathcal{A}}(t) \quad (55)$$

First, observe that by the definition of minimal BRS, we have that if any state $\bar{z} \in \mathcal{A}(t)$, then

$$\forall s \in [t, 0], \forall u(\cdot) \in \mathbb{U}, \zeta(s; \bar{z}, t, u(\cdot)) \in \mathcal{A}(s) \quad (56)$$

since otherwise, we would have for some $\bar{s} \in [t, 0]$,

$$\begin{aligned} \exists u(\cdot) \in \mathbb{U}, \zeta(\bar{s}; \bar{z}, t, u(\cdot)) &\notin \mathcal{A}(\bar{s}) \\ \Rightarrow \exists u(\cdot) \in \mathbb{U}, \zeta(0; \zeta(\bar{s}; \bar{z}, t, u(\cdot)), \bar{s}, u(\cdot)) &\notin \mathcal{T} \\ \Leftrightarrow \exists u(\cdot) \in \mathbb{U}, \zeta(0; \bar{z}, t, u(\cdot)) &\notin \mathcal{T} \end{aligned} \quad (57)$$

which contradicts $\bar{z} \in \mathcal{A}(t)$.

Given $z \notin \mathcal{A}(t)$, there exists some control $\bar{u}(\cdot)$ such that $\zeta(0; z, t, \bar{u}(\cdot)) \notin \mathcal{T} = \mathcal{A}(0)$. Moreover, we must have $\forall s \in [t, 0], \zeta(s; z, t, \bar{u}(\cdot)) \notin \mathcal{T}$, since otherwise, we would have $\exists \hat{s}$ such that

$$\begin{aligned} \zeta(\hat{s}; z, t, \bar{u}(\cdot)) &\in \mathcal{T} = \mathcal{A}(0) \\ \Rightarrow z &= \zeta(t; z, t, \bar{u}(\cdot)) \in \mathcal{A}(t - \hat{s}) \end{aligned} \quad (58)$$

which contradicts $z \notin \bigcup_{s \in (t, 0)} \mathcal{A}(s)$.

Using time-invariance of the system dynamics, we have $\forall s \in [t, 0], \zeta(0; z, t - s, \bar{u}(\cdot)) \notin \mathcal{T}$, which is equivalent to $\forall s \in [t, 0], \zeta(0; z, s, \bar{u}(\cdot)) \notin \mathcal{T}$. Therefore, $\exists u(\cdot) \in \mathbb{U}, \forall s \in [t, 0], \zeta(0; z, s, \bar{u}(\cdot)) \notin \mathcal{T} \Leftrightarrow z \notin \bar{\mathcal{A}}(t)$. ■

Remark 6: The way to compute the union of BRSs over a continuous time interval depends on the reachability formulation. In the case of HJ reachability, which is used to obtain numerical examples in this paper, time is discretized to satisfy the Courant-Friedrichs-Lewy (CFL) condition, ensuring the underlying numerical methods are stable. We save sets at each time interval, resulting in a series of look-up tables of dimension n , where n is the number of states; the number of such look-up tables is the number of discrete time steps. As in [21], sets are represented by the zero sublevel sets of functions, so taking the union of sets amounts to taking an element-wise minimum between a finite number of different look-up tables corresponding to each time interval.

Remark 7: When $\exists s \in [t, 0], \mathcal{A}(s) = \emptyset$, it is currently not known whether the union of the BRSs $\mathcal{A}(s)$ will be equal to the BRT $\bar{\mathcal{A}}(t)$ or a proper subset of the BRT $\bar{\mathcal{A}}(t)$. Both

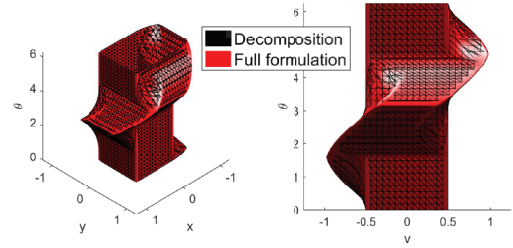


Fig. 7: The BRT computed directly in 3D (red surface) and computed via decomposition in 2D (black mesh). Using our decomposition technique, we first compute the BRSs $\mathcal{A}(s), s \in [-0.5, 0]$, and then obtained the BRT by taking their union.

are possibilities. Finding a weaker condition under which the union of BRSs equals to the BRT is an important future direction that we plan to investigate.

Remark 8: Note that Proposition 4 and Theorem 3 also hold for decoupled control.

A summary of the results from this section can be seen in Table II under Section VI-A, and Table III under Section VI-B.

E. Numerical Results

We now revisit the Dubins Car, whose full system and subsystem dynamics are given in (6) and (7) respectively. Using the target set \mathcal{T} given in (27) and writing \mathcal{T} in the form of (28), we computed the BRT $\bar{\mathcal{A}}(t), t = -0.5$ by first computing $\mathcal{A}(s), s \in [-0.5, 0]$, and then taking their union.

Fig. 7 shows the BRT $\bar{\mathcal{A}}(t), t = -0.5$ computed directly in 3D and via decomposition. Since $\mathcal{A}(s) \neq \emptyset \forall s \in [-0.5, 0]$, reconstruction does not incur additional approximation errors.

VII. NUMERICAL RESULTS: 6D ACROBATIC QUADROTOR

In this section, we show numerical results for the 6D Acrobatic Quadrotor, a system whose BRSs and BRTs were intractable to compute with the HJ formulation to the best of our knowledge. In [39], a 6D quadrotor model used to perform backflips was simplified into a series of smaller models linked together in a hybrid system. The quadrotor has state $z = (p_x, v_x, p_y, v_y, \phi, \omega)$, and dynamics

$$\begin{bmatrix} \dot{p}_x \\ \dot{v}_x \\ \dot{p}_y \\ \dot{v}_y \\ \dot{\phi} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -\frac{1}{m} C_D^v v_x - \frac{T_1}{m} \sin \phi - \frac{T_2}{m} \sin \phi \\ -\frac{1}{m} (mg + C_D^v v_y) + \frac{T_1}{m} \cos \phi + \frac{T_2}{m} \cos \phi \\ \omega \\ -\frac{1}{I_{yy}} C_D^\phi \omega - \frac{l}{I_{yy}} T_1 + \frac{l}{I_{yy}} T_2 \end{bmatrix} \quad (59)$$

where x, y , and ϕ represent the quadrotor's horizontal, vertical, and rotational positions, respectively. Their derivatives represent the velocity with respect to each state. The control inputs T_1 and T_2 represent the thrust exerted on either end of the quadrotor, and the constant system parameters are m for mass, C_D^v for translational drag, C_D^ϕ for rotational drag, g for acceleration due to gravity, l for the length from the quadrotor's center to an edge, and I_{yy} for moment of inertia.

We decompose the system into the following subsystems:

$$x_1 = (p_x, v_x, \phi, \omega), \quad x_2 = (p_y, v_y, \phi, \omega) \quad (60)$$

For this example we will compute $\mathcal{A}(t)$ and $\bar{\mathcal{A}}(t)$, which describe the set of initial conditions from which the system

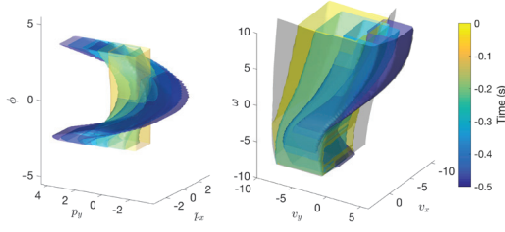


Fig. 8: Left: 3D positional slices of the reconstructed 6D BRSs at $v_x = v_y = 1$, $\omega = 0$ at different points in time. The BRT cannot be seen in this image because it encompasses the entire union of BRSs. Right: 3D velocity slices of the reconstructed 6D BRSs at $x, y = 1.5$, $\phi = 1.5$ at different points in time. The BRT can be seen as the transparent gray surface that encompasses the sets.

may enter the target set despite the best possible control to avoid the target. We define the target set as a square of length 2 centered at $(p_x, p_y) = (0, 0)$ described by $\mathcal{T} = \{(p_x, v_x, p_y, v_y, \phi, \omega) : |p_x|, |p_y| \leq 1\}$. This can be interpreted as a positional box centered at the origin that must be avoided for all angles and velocities. From the target set, we define $l(z)$ such that $l(z) \leq 0 \Leftrightarrow z \in \mathcal{T}$. This target set is then decomposed as follows:

$$\begin{aligned}\mathcal{T}_1 &= \{(p_x, v_x, \phi, \omega) : |p_x| \leq 1\} \\ \mathcal{T}_2 &= \{(p_y, v_y, \phi, \omega) : |p_y| \leq 1\}\end{aligned}$$

The BRS of each 4D subsystem is computed and then recombined into the 6D BRS. To visually depict the 6D BRS, 3D slices of the BRS along the positional and velocity axes were computed. The left image in Fig. 8 shows a 3D slice in (p_x, p_y, ϕ) space at $v_x = v_y = 1, \omega = 0$. The yellow set represents the target set \mathcal{T} , with the BRS in other colors. Shown on the right in Fig. 8 are 3D slices in (v_x, v_y, ω) space at $p_x, p_y = 1.5, \phi = 1.5$ through different points in time. The sets grow darker as time propagates backward. In the case of HJ reachability, time is discretized, and sets are represented via the zero sublevel set of an implicit surface function [21]. Thus, the union in Theorem 3 for obtaining the BRT is computed by taking the element-wise minimum between the stored BRSs at each discretized time step in the time interval. The BRT can be seen in Fig. 8 as the gray surface.

VIII. HANDLING DISTURBANCES

Under the presence of disturbances, the full system dynamics changes from (1) to

$$\frac{dz}{ds} = \dot{z} = f(z, u, d), s \in [t, 0], u \in \mathcal{U}, d \in \mathcal{D} \quad (61)$$

where $d \in \mathcal{D}$ represents the disturbance, with $d(\cdot) \in \mathbb{D}$ drawn from the set of measurable functions.

In addition, we assume that the disturbance function $d(\cdot)$ is drawn from the set of non-anticipative strategies [21], denoted $\Gamma(t)$. We denote the mapping from $u(\cdot)$ to $d(\cdot)$ as $\gamma[u](\cdot)$ as in [21]. The subsystems in (5) are now written as

$$\begin{aligned}\dot{z}_1 &= f_1(z_1, z_c, u, d) \\ \dot{z}_2 &= f_2(z_2, z_c, u, d) \\ \dot{z}_c &= f_c(z_c, u, d)\end{aligned} \quad (62)$$

In general, subsystem disturbances may have shared components. Whether this is the case is very important, as some of the results involving disturbances become stronger when the subsystem disturbances do not have shared components.

Trajectories of the system and subsystems are now denoted $\zeta(s; z, t, u(\cdot), d(\cdot))$, $\xi_i(s; x_i, t, u(\cdot), d(\cdot))$, and satisfy conditions analogous to (2) and (11) respectively. We also need to incorporate the disturbance into the BRS and BRT definitions:

$$\begin{aligned}\bar{\mathcal{A}}(t) &= \{z : \exists \gamma[u](\cdot), \forall u(\cdot), \exists s \in [t, 0], \\ &\quad \zeta(s; z, t, u(\cdot), \gamma[u](\cdot)) \in \mathcal{T}\} \\ \bar{\mathcal{R}}(t) &= \{z : \forall \gamma[u](\cdot), \exists u(\cdot), \exists s \in [t, 0], \\ &\quad \zeta(s; z, t, \gamma[u](\cdot), d(\cdot)) \in \mathcal{T}\} \\ \mathcal{A}(t) &= \{z : \exists \gamma[u](\cdot), \forall u(\cdot), \zeta(0; z, t, u(\cdot), \gamma[u](\cdot)) \in \mathcal{T}\} \\ \mathcal{R}(t) &= \{z : \forall \gamma[u](\cdot), \exists u(\cdot), \zeta(0; z, t, u(\cdot), \gamma[u](\cdot)) \in \mathcal{T}\}\end{aligned} \quad (63)$$

Subsystem BRSs $\mathcal{R}_i, \mathcal{A}_i, i = 1, 2$ are defined analogously.

A. Self-Contained Subsystems

Under the presence of disturbances, the results from Section IV carry over with some modifications. Theorems 1 and 2 need to be changed slightly, and the reconstructed BRS is now an approximation conservative in the right direction.

Corollary 3: Suppose that the full system in (1) can be decomposed into the form of (62), then

$$\begin{aligned}\mathcal{T} &= \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \mathcal{R}(t) &\supseteq \text{proj}^{-1}(\mathcal{R}_1(t)) \cup \text{proj}^{-1}(\mathcal{R}_2(t))\end{aligned} \quad (64)$$

To solve this, in the proof of Theorem 1, (24) becomes

$$\begin{aligned}\forall \gamma[u](\cdot), \exists u(\cdot), \quad &\xi_1(0; \bar{x}_1, t, u(\cdot), d(\cdot)) \in \mathcal{T}_1 \vee \\ &\xi_2(0; \bar{x}_2, t, u(\cdot), d(\cdot)) \in \mathcal{T}_2\end{aligned} \quad (65)$$

The expression “ $\forall \gamma[u](\cdot), \exists u(\cdot)$ ” can no longer be distributed, thus making the reconstructed BRS a conservative approximation of the true BRS in the right direction. By conservative in the right direction, we mean that a state z in the reconstructed BRS is guaranteed to be able to reach the target.

Corollary 4: Suppose that the full system in (1) can be decomposed into the form of (62), then

$$\begin{aligned}\mathcal{T} &= \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \mathcal{A}(t) &\subseteq \text{proj}^{-1}(\mathcal{A}_1(t)) \cap \text{proj}^{-1}(\mathcal{A}_2(t))\end{aligned} \quad (66)$$

The proof of Theorem 2 makes the same arguments except that it involves complements of sets instead. Again, the reconstructed BRS is a conservative approximation of the true BRS in the right direction, meaning that a state z outside of the reconstructed BRS is guaranteed to be able to avoid the target.

If the subsystem disturbances have no shared components, then (65) becomes

$$\forall (\gamma_1[u](\cdot), \gamma_2[u](\cdot)), \exists u(\cdot), \quad \xi_1(0; \bar{x}_1, t, u(\cdot), \gamma_1[u](\cdot)) \in \mathcal{T}_1 \vee \\ \xi_2(0; \bar{x}_2, t, u(\cdot), \gamma_2[u](\cdot)) \in \mathcal{T}_2$$

where $\gamma[u](\cdot)$ is written as $(\gamma_1[u](\cdot), \gamma_2[u](\cdot))$.

In this case, the expression “ $\forall (\gamma_1[u](\cdot), \gamma_2[u](\cdot)), \exists u(\cdot)$ ” can be distributed. Therefore, in this case Theorems 1 and 2 still hold.

B. Subsystems with Decoupled Control

For systems with decoupled control, but shared disturbance in the subsystems, results from Section VIII-A still hold since the system dynamics structure is a special case of that in Section VIII-A. In addition, results from Section V hold with some modifications. Propositions 1 and 2 need to be modified, and again the reconstructed BRS is now an approximation conservative in the right direction.

Corollary 5: Suppose that the full system in (1) can be decomposed into the form of (29), with the addition of shared disturbances. Then,

$$\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \mathcal{R}(t) \supseteq \text{proj}^{-1}(\mathcal{R}_1(t)) \cap \text{proj}^{-1}(\mathcal{R}_2(t)) \quad (67)$$

To prove this, we modify Proposition 1 by changing (38) to

$$\forall \gamma[u](\cdot), \exists (u_1(\cdot), u_2(\cdot)) \\ (\eta_1(s; \bar{z}_1, \bar{z}_c, t, u_1(\cdot), \gamma[u](\cdot)), \eta_c(s; \bar{z}_c, t)) \in \mathcal{T}_1 \wedge \\ (\eta_2(s; \bar{z}_2, \bar{z}_c, t, u_2(\cdot), \gamma[u](\cdot)), \eta_c(s; \bar{z}_c, t)) \in \mathcal{T}_2 \quad (68)$$

The expression “ $\forall \gamma[u](\cdot), \exists (u_1(\cdot), u_2(\cdot))$ ” cannot be distributed to lead to a statement analogous to (39). Hence, the forward direction of Proposition 1 does not hold, and conservativeness is introduced.

By the same reasoning, the result of Proposition 2 changes to the following.

Corollary 6:

$$\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \mathcal{A}(t) \subseteq \text{proj}^{-1}(\mathcal{A}_1(t)) \cup \text{proj}^{-1}(\mathcal{A}_2(t)) \quad (69)$$

In both cases, conservative approximations of the BRS can still be obtained. A summary of the results from this section can be seen in Table I under Sections VIII-A & VIII-B.

C. Decomposition of Reachable Tubes

Under disturbances, the results from Section VI carry over with modifications. For reconstruction from other BRTs, the arguments in Proposition 3 do not change. However, in the case where there are overlapping components in the subsystem disturbances, the reconstructed BRTs become conservative approximations:

Corollary 7: Suppose our system has coupled control and disturbance as in (62), then

$$\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \bar{\mathcal{R}}(t) \supseteq \text{proj}^{-1}(\bar{\mathcal{R}}_1(t)) \cup \text{proj}^{-1}(\bar{\mathcal{R}}_2(t)) \quad (70)$$

Corollary 8: Suppose our system has subsystem controls that do not share any components, then

$$\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cup \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \bar{\mathcal{A}}(t) \subseteq \text{proj}^{-1}(\bar{\mathcal{A}}_1(t)) \cup \text{proj}^{-1}(\bar{\mathcal{A}}_2(t)) \quad (71)$$

For Proposition 4, the union of the BRSs now becomes an under-approximation of the BRT in general:

Corollary 9: Suppose our system has coupled control and disturbance as in (62), then

$$\mathcal{T} = \text{proj}^{-1}(\mathcal{T}_1) \cap \text{proj}^{-1}(\mathcal{T}_2) \\ \Rightarrow \bigcup_{s \in [t, 0]} \mathcal{R}(s) \subseteq \bar{\mathcal{R}}(t) \quad (72)$$

To show this, all arguments in the proof of Proposition 4 remain the same, except (45) no longer implies (49). Instead, the implication is unidirectional:

$$\exists s \in [t, 0], \forall \gamma[u](\cdot), \exists u(\cdot), \zeta(0; z, s, u(\cdot), \gamma[u](\cdot)) \in \mathcal{T} \\ \Rightarrow \forall \gamma[u](\cdot), \exists u(\cdot), \exists s \in [t, 0], \zeta(s; z, t, u(\cdot), \gamma[u](\cdot)) \in \mathcal{T} \quad (73)$$

This is due to the switching of the order of the expressions “ $\exists s \in [t, 0]$ ” and “ $\gamma[u](\cdot)$ ”. Therefore, the union of the BRSs becomes an under-approximation of the BRT, a conservatism in the right direction: a state in the under-approximated BRT is still guaranteed to be able to reach the target.

In contrast to Proposition 4, all the arguments of Theorem 3 hold, since there no change of order of expressions involving existential and universal quantifiers. A summary of the results from this section can be seen in Table II & III under Section VIII-C.

D. Dubins Car with Disturbances

Under disturbances, the Dubins Car dynamics are given by

$$\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v \cos \theta + d_x \\ v \sin \theta + d_y \\ \omega + d_\theta \end{bmatrix} \quad (74) \\ \omega \in \mathcal{U}, \quad (d_x, d_y, d_\theta) \in \mathcal{D}$$

with state $z = (p_x, p_y, \theta)$, control $u = \omega$, and disturbances $d = (d_x, d_y, d_\theta)$. The state partitions are $z_1 = p_x, z_2 = p_y, z_c = \theta$. The subsystems states x_i , controls w_i , and disturbances b_i are

$$x_1 = \begin{bmatrix} \dot{z}_1 \\ \dot{z}_c \end{bmatrix} = \begin{bmatrix} \dot{p}_x \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v \cos \theta + d_x \\ \omega + d_\theta \end{bmatrix} \\ x_2 = \begin{bmatrix} \dot{z}_2 \\ \dot{z}_c \end{bmatrix} = \begin{bmatrix} \dot{p}_y \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v \sin \theta + d_y \\ \omega + d_\theta \end{bmatrix} \quad (75) \\ u_c = \omega = u \\ d_1 = d_x, d_2 = d_y, d_c = d_\theta$$

where the overlapping state is $\theta = z_c$. We assume that each component of disturbance is bounded in some interval centered at zero: $|d_x| \leq \bar{d}_x, |d_y| \leq \bar{d}_y, |d_\theta| \leq \bar{d}_\theta$. The subsystem disturbances b_1 and b_2 have the shared component d_θ .

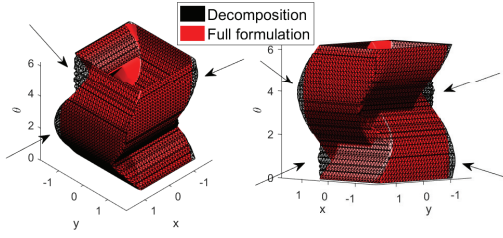


Fig. 9: Minimal BRTs computed directly in 3D and via decomposition in 2D for the Dubins Car under disturbances with shared components. The reconstructed BRT is an over-approximation of the true BRT. The over-approximated regions of the reconstruction are indicated by the arrows.

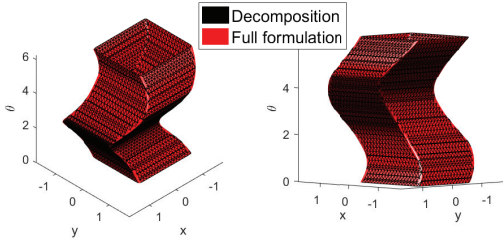


Fig. 10: Minimal BRTs computed directly in 3D and via decomposition in 2D for the Dubins Car under disturbances *without* shared components. In this case, the BRT computed using decomposition matches the true BRT.

Fig. 9 compares the BRT $\bar{A}(t)$, $t = -0.5$ computed directly from the target set in (27), and using our decomposition technique from the subsystem target sets in (28), wherein the union is taken over a discrete number of BRSs over the time interval. For this computation, we chose $\bar{d}_x, \bar{d}_y = 1, \bar{d}_\theta = 5$.

Since there is a shared component in the disturbances, the BRT computed using our decomposition technique becomes an over-approximation of the true BRT. One can see the over-approximation by noting that the black set is not flush against the red set, as marked by the arrows in Fig. 9.

Fig. 10 shows the same computation with $\bar{d}_\theta = 0$, so that subsystem disturbances effectively have no shared components. In this case, one can see that the BRTs computed directly in 3D and via decomposition in 2D are the same.

IX. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a general system decomposition method for efficiently computing BRSs and BRTs in several scenarios. By performing computations in lower-dimensional subspaces, computation burden is substantially reduced, allowing currently tractable computations to be orders of magnitude more faster, and currently intractable computations to become tractable. Unlike related work on computation of BRSs and BRTs, our method can significantly reduce dimensionality without sacrificing any optimality.

Under disturbances, the reconstructed BRSs and BRTs sometimes become slightly conservative approximations which are still useful for providing performance and safety guarantees. To the best of our knowledge, such guarantees for high-dimensional systems are now possible for the first time. Our

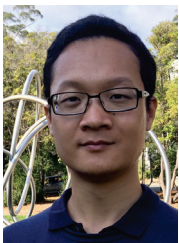
decomposition technique can also be used in combination with other dimensionality reduction or approximation techniques, further alleviating the curse of dimensionality.

We are currently extending our decomposition technique to other scenarios, including more representations of full-dimensional sets in lower-dimensional subspaces and more families of system dynamics. In addition, we look forward to combining our technique with other related techniques such as reinforcement learning and machine learning, automating the system decomposition process, and demonstrating our theory in hardware experiments.

REFERENCES

- [1] M. Chen, S. Herbert, and C. J. Tomlin, "Exact and efficient Hamilton-Jacobi-based guaranteed safety analysis via system decomposition," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2017.
- [2] C. Zhao, U. Topcu, N. Li, and S. Low, "Design and Stability of Load-Side Primary Frequency Control in Power Systems," *IEEE Trans. Autom. Control*, vol. 59, no. 5, pp. 1177–1189, May 2014.
- [3] R. Dobbe, D. Arnold, S. Liu, D. Callaway, and C. Tomlin, "Real-Time Distribution Grid State Estimation with Limited Sensors and Load Forecasting," in *Int. Conf. on Cyber-Physical Syst.*, Apr. 2016, pp. 1–10.
- [4] R. Ghosh and C. J. Tomlin, "Lateral Inhibition through Delta-Notch Signaling: A Piecewise Affine Hybrid Model," in *Proc. ACM Int. Conf. Hybrid Syst.: Computation and Contr.*, 2001, pp. 232–246.
- [5] R. Ghosh and C. Tomlin, "A query-based technique for interpreting reachable sets for hybrid automaton models of protein feedback signaling," in *Proc. Amer. Contr. Conf.*, 2005, pp. 4417–4422.
- [6] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, "Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment," in *Guidance, Navigation and Contr. Conf. and Exhibit*, Aug. 2007.
- [7] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *Intelligent Vehicles Symp.*, Jun. 2015, pp. 1094–1099.
- [8] S. Kong, S. Gao, W. Chen, and E. Clarke, "dReach: δ -Reachability Analysis for Hybrid Systems," in *Proc. Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems*, 2015, pp. 200–205.
- [9] P. S. Duggirala, S. Mitra, M. Viswanathan, and M. Potok, "C2E2: A Verification Tool for Stateflow Models," in *Proc. Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems*, 2015, pp. 68–82.
- [10] G. Frehse, C. Le Guernic, A. Donzé, S. Cotton, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, and O. Maler, "SpaceEx: Scalable Verification of Hybrid Systems," in *Proc. Int. Conf. on Computer Aided Verification*, 2011, pp. 379–395.
- [11] A. Kurzhanski and P. Varaiya, "Ellipsoidal techniques for reachability analysis: internal approximation," *Syst. & Contr. Lett.*, vol. 41, no. 3, pp. 201–211, Oct. 2000.
- [12] —, "On Ellipsoidal Techniques for Reachability Analysis. Part II: Internal Approximations Box-valued Constraints," *Optimization Methods and Software*, vol. 17, no. 2, pp. 207–237, Jan. 2002.
- [13] J. N. Maidens, S. Kaynama, I. M. Mitchell, M. M. K. Oishi, and G. A. Dumont, "Lagrangian methods for approximating the viability kernel in high-dimensional systems," *Automatica*, vol. 49, no. 7, pp. 2017–2029, Jul. 2013.
- [14] X. Chen, E. Ábrahám, and S. Sankaranarayanan, "Flow*: An Analyzer for Non-linear Hybrid Systems," in *Proc. Conf. Computer Aided Verification*, 2013, pp. 258–263.
- [15] M. Althoff, "An Introduction to CORA 2015," in *Proc. Workshop on Applied Verification for Continuous and Hybrid Systems*, 2015.
- [16] A. Majumdar, R. Vasudevan, M. M. Tobenkin, and R. Tedrake, *Convex optimization of nonlinear feedback controllers via occupation measures*. Springer Berlin Heidelberg, Aug. 2014, vol. 33, no. 9, pp. 1209–1230.
- [17] T. Dreossi, T. Dang, and C. Piazza, "Parallelotope Bundles for Polynomial Reachability," in *Proc. ACM Int. Conf. Hybrid Syst.: Computation and Contr.*, 2016, pp. 297–306.
- [18] P. Nilsson and N. Ozay, "Synthesis of separable controlled invariant sets for modular local control design," in *American Control Conference*, Jul. 2016, pp. 5656–5663.
- [19] M. Althoff and B. H. Krogh, "Reachability Analysis of Nonlinear Differential-Algebraic Systems," *IEEE Trans. Automat. Contr.*, vol. 59, no. 2, pp. 371–383, Feb. 2014.

- [20] E. N. Barron, "Differential games with maximum cost," *Nonlinear Anal.*, vol. 14, no. 11, pp. 971–989, Jun. 1990.
- [21] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Trans. Autom. Control*, vol. 50, no. 7, pp. 947–957, 2005.
- [22] K. Margellos and J. Lygeros, "Hamilton-Jacobi Formulation for Reach-Avoid Differential Games," *IEEE Trans. Autom. Control*, vol. 56, no. 8, pp. 1849–1861, Aug. 2011.
- [23] O. Bokanowski and H. Zidani, "Minimal Time Problems with Moving Targets and Obstacles," *IFAC Proc. Volumes*, vol. 44, no. 1, pp. 2589–2593, Jan. 2011.
- [24] J. Darbon and S. Osher, "Algorithms for overcoming the curse of dimensionality for certain Hamilton-Jacobi equations arising in control theory and elsewhere," *Res. Math. Sci.*, vol. 3, no. 1, p. 19, Dec. 2016.
- [25] M. R. Hafner and D. Del Vecchio, "Computation of safety control for uncertain piecewise continuous systems on a partial order," in *Proc. Conf. Decision and Contr.*, 2009, pp. 1671–1677.
- [26] S. Coogan and M. Arcak, "Efficient finite abstraction of mixed monotone systems," in *Proc. ACM Int. Conf. Hybrid Syst.: Computation and Contr.*, 2015, pp. 58–67.
- [27] I. Mitchell, "Scalable calculation of reach sets and tubes for nonlinear systems with terminal integrators: a mixed implicit explicit formulation," in *Proc. ACM Int. Conf. Hybrid Syst.: Computation and Contr.*, 2011, pp. 103–112.
- [28] J. F. Fisac, M. Chen, C. J. Tomlin, and S. S. Sastry, "Reach-avoid problems with time-varying dynamics, targets and constraints," in *Proc. ACM Int. Conf. Hybrid Syst.: Computation and Contr.*, 2015, pp. 11–20.
- [29] I. M. Mitchell and C. J. Tomlin, "Overapproximating Reachable Sets by Hamilton-Jacobi Projections," *J. Scientific Computing*, vol. 19, no. 1–3, pp. 323–346, 2003.
- [30] M. Chen, S. Herbert, and C. J. Tomlin, "Fast reachable set approximations via state decoupling disturbances," in *IEEE Conf. Decision and Contr.*, Dec. 2016, pp. 191–196.
- [31] S. Kaynama and M. Oishi, "Schur-based decomposition for reachability analysis of linear time-invariant systems," in *Proc. IEEE Conf. on Decision and Control*, Dec. 2009, pp. 69–74.
- [32] —, "A Modified Riccati Transformation for Decentralized Computation of the Viability Kernel Under LTI Dynamics," *IEEE Trans. on Automatic Control*, vol. 58, no. 11, pp. 2878–2892, Nov. 2013.
- [33] F. M. Callier and C. A. Desoer, "The System Representation $R=[A,B,C,D]$, Part II," in *Linear System Theory*. Springer, 1991, pp. 103–139.
- [34] S. S. Sastry, "Linearization by State Feedback," in *Nonlinear Systems: Analysis, Stability, and Control*. Springer-Verlag, 1999, ch. 9, p. 384.
- [35] E. A. Coddington and N. Levinson, *Theory of ordinary differential equations*. Tata McGraw-Hill Education, 1955.
- [36] I. M. Mitchell, "Comparing Forward and Backward Reachability as Tools for Safety Analysis," in *Proc. ACM Int. Conf. Hybrid Syst.: Computation and Contr.*, 2007, pp. 428–443.
- [37] P. P. Varaiya, "On the Existence of Solutions to a Differential Game," *SIAM J. Contr.*, vol. 5, no. 1, pp. 153–162, Feb. 1967.
- [38] P. Zhao, S. Mohan, and R. Vasudevan, "Control Synthesis for Nonlinear Optimal Control via Convex Relaxations," Oct. 2016. [Online]. Available: <http://arxiv.org/abs/1610.00394>
- [39] J. H. Gillula, G. M. Hoffmann, Haomiao Huang, M. P. Vitus, and C. J. Tomlin, "Applications of hybrid reachability analysis to robotic aerial vehicles," *Int. J. of Robotics Res.*, vol. 30, no. 3, pp. 335–354, Mar. 2011.



Mo Chen is a postdoctoral researcher in the Aeronautics and Astronautics Department at Stanford University with Marco Pavone. He completed his PhD in the Electrical Engineering and Computer Sciences Department at the University of California, Berkeley with Claire Tomlin in 2017, and received his BSc in Engineering Physics from the University of British Columbia in 2011. His research interests include multi-agent systems, safety-critical systems, and control theory. Mo received the 2017 Eli Jury Award and the 2016 Demetri Angelakos Memorial

Achievement Award. Mo will be joining Simon Fraser University in Burnaby, BC, Canada as an Assistant Professor in the School of Computing Science in Fall 2018.



ence Foundation Graduate Research Fellowship and the Berkeley Chancellor's Fellowship.

Sylvia L. Herbert is a PhD student in the Electrical Engineering and Computer Science Department at the University of California, Berkeley. She completed her BS and MS in Mechanical Engineering at Drexel University in 2014. Her research interests include the theory and application of new methods for the efficient navigation of dynamic systems with theoretical safety guarantees, as well as the application of human-like cognitive processes to robotic algorithms for improved performance and human-robot interaction. Sylvia received the National Science



Mahesh S. Vashishtha received his B.A. degree in Computer Science from the University of California, Berkeley in 2017. His research interests include safety-critical systems and control theory. He is currently a software engineer on the Display Ads team at Google.



control theory frameworks to improve the overall controller performance, especially when the system is operating in an uncertain environment.

Somil Bansal is currently a Ph.D. candidate in Electrical Engineering and Computer Sciences at the University of California, Berkeley. He completed his B.Tech. in Electrical Engineering from Indian Institute of Technology, Kanpur, and an M.S. in Electrical Engineering and Computer Sciences from UC Berkeley in 2012 and 2014, respectively. His research interests include Hamilton-Jacobi reachability theory and its applications to multi-agent autonomous systems. He is also interested in exploring how machine learning tools can be used within the



2010 held the Tage Erlander Professorship of the Swedish Research Council at KTH in Stockholm.

Claire J. Tomlin is the Charles A. Desoer Professor of Engineering in Electrical Engineering and Computer Sciences at the University of California, Berkeley. She was an Assistant, Associate, and Full Professor in Aeronautics and Astronautics at Stanford from 1998 to 2007, and in 2005 joined Berkeley. Claire works in the area of control theory and hybrid systems, with applications to air traffic management, UAV systems, energy, robotics, and systems biology. She is a MacArthur Foundation Fellow (2006) and an IEEE Fellow (2010), and in