University of Pennsylvania
ScholarlyCommons

Departmental Papers (ESE)

Department of Electrical & Systems Engineering

6-1-2000

Hierarchically Consistent Control Systems

George J. Pappas
University of Pennsylvania, pappasg@seas.upenn.edu

Gerardo Lafferriere
Portland State University

Shankar Sastry
University of California, Berkeley

Copyright 2000 IEEE. Reprinted from *IEEE Transactions on Automatic Control*, Volume 45, Issue 6, June 2000, pages 1144-1160.
Publisher URL: http://dx.doi.org/10.1109/9.863598

This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of the University of Pennsylvania's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to pubs-permissions@ieee.org. By choosing to view this document, you agree to all provisions of the copyright laws protecting it.

This paper is posted at ScholarlyCommons. http://repository.upenn.edu/ese_papers/87
For more information, please contact repository@pobox.upenn.edu.

Hierarchically Consistent Control Systems

Abstract
Large-scale control systems typically possess a hierarchical architecture in order to manage complexity. Higher levels of the hierarchy utilize coarser models of the system, resulting from aggregating the detailed lower level models. In this layered control paradigm, the notion of hierarchical consistency is important, as it ensures the implementation of high-level objectives by the lower level system. In this paper, we define a notion of modeling hierarchy for continuous control systems and obtain characterizations for hierarchically consistent linear systems with respect to controllability objectives. As an interesting byproduct, we obtain a hierarchical controllability criterion for linear systems from which we recover the best of the known controllability algorithms from numerical linear algebra.

Keywords
abstraction, consistency, controllability algorithms, hierarchical control, continuous time systems, controllability, hierarchical systems, large-scale systems, linear algebra, linear systems, consistency, continuous control systems, linear systems

Comments
Copyright 2000 IEEE. Reprinted from *IEEE Transactions on Automatic Control*, Volume 45, Issue 6, June 2000, pages 1144-1160.
Publisher URL: http://dx.doi.org/10.1109/9.863598

This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of the University of Pennsylvania's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to pubs-permissions@ieee.org. By choosing to view this document, you agree to all provisions of the copyright laws protecting it.

This journal article is available at ScholarlyCommons: http://repository.upenn.edu/ese_papers/87

Hierarchically Consistent Control Systems

George J. Pappas, *Member, IEEE*, Gerardo Lafferriere, and Shankar Sastry, *Fellow, IEEE*

***Abstract*—Large-scale control systems typically possess a hierarchical architecture in order to manage complexity. Higher levels of the hierarchy utilize coarser models of the system, resulting from aggregating the detailed lower level models. In this layered control paradigm, the notion of hierarchical consistency is important, as it ensures the implementation of high-level objectives by the lower level system. In this paper, we define a notion of modeling hierarchy for continuous control systems and obtain characterizations for hierarchically consistent linear systems with respect to controllability objectives. As an interesting byproduct, we obtain a hierarchical controllability criterion for linear systems from which we recover the best of the known controllability algorithms from numerical linear algebra.**

***Index Terms*—Abstraction, consistency, controllability algorithms, hierarchical control.**

I. Introduction

LARGE-SCALE systems such as Intelligent Vehicle Highway Systems [34] and Air Traffic Management Systems [28] are systems of very high complexity. Complexity is typically reduced by imposing a hierarchical structure on the system architecture. In such a structure, systems of higher functionality reside at higher levels of the hierarchy and are therefore unaware of unnecessary lower-level details. The main types of hierarchical structures are classified and described in the visionary work of [23].

Fig. 1 shows a typical two-layer control hierarchy which is frequently used in the quite common planning and control hierarchical systems. Multilayered versions of Fig. 1 are used in both [28] and [34]. In this layered control paradigm, each layer has different objectives. In performing their tasks, the higher level uses a coarser system model than the lower level. One of the main challenges in hierarchical systems is the extraction of a hierarchy of models at various levels of abstraction which are compatible with the functionality and objectives of each layer.

In the literature, the notions of *abstraction* or *aggregation* refer to grouping the system states into equivalence classes. Depending on the cardinality of the resulting quotient space, we may have *discrete* or *continuous* abstractions. With this notion of abstraction, the abstracted system will be defined as the induced quotient dynamics. Discrete abstractions of continuous systems have been considered in [7], [8] as well as [2], [10], [31]. Hierarchical systems for discrete event systems have been formally considered in [6], [35], [36], [38]. In this paper, we focus on *continuous abstractions*. Therefore, our first priority is to have a formal notion of quotient control systems.

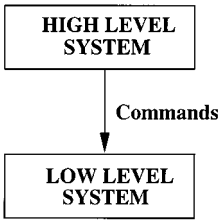

Fig. 1. Two-layer control hierarchy

Problem 1.1: Given a control system

$$\dot{x} = f(x, u), \qquad x \in \mathbb{R}^n \quad u \in \mathbb{R}^m \tag{1.1}$$

and some map $y = h(x)$, *where* h: $\mathbb{R}^n \longrightarrow \mathbb{R}^p$, $p \leq n$, *we would like to define a control system*

$$\dot{y} = g(y, v), \qquad y \in \mathbb{R}^p \quad v \in \mathbb{R}^k \tag{1.2}$$

which can produce as trajectories all functions of the form $y(t) = h(x(t))$, *where* $x(t)$ *is a trajectory of system* (1.1). *That is,* h *maps trajectories of system (1.1) to trajectories of system (1.2).*

The function h is the "quotient map" which performs the state aggregation. System (1.2) will be referred to as the *abstraction* [27] or *macromodel* of the finer *micromodel* (1.1). Note that the control input v of the coarser model (1.2) is not the same input u of system (1.1) and should be thought of as a macroinput. For example, v can be velocity inputs of a kinematic model, whereas u may be force and torque inputs of a dynamic model. This is, therefore, quite different from model-reduction techniques which reduce or aggregate dynamics while using the same control inputs [3], [15]–[18]. The difference between model reduction and abstraction is illustrated in Fig. 2.

We will solve Problem 1.1 by first generalizing the geometric notion of Φ-related vector fields to control systems. A notion of Φ-related control systems would allow us to push forward control systems through quotient maps and obtain well-defined control systems describing the aggregate dynamics. The notion of Φ-related control systems introduced in this paper is more general than the notion of projectable systems defined in [18] and [22] (see Example 3.6), as we will show that given any control system and any surjective map Φ, there always exists another system that is Φ-related to it. Our notion of Φ-related control systems mathematically formalizes the concept of *virtual inputs*

Manuscript received May 12, 1998; revised April 16, 1999. Recommended by Associate Editor M. Krstic. This work was supported by DARPA under Grants F33615-98-C-3614 and F33615-00-C-1707.

G. J. Pappas was with the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA 94720 USA. He is now with the Department of Electrical Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: pappasg@seas.upenn.edu).

G. Lafferriere is with the Department of Mathematical Sciences, Portland State University, Portland, OR 97207 USA (e-mail: gerardo@mth.pdx.edu).

S. Sastry is with the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA 94720 USA (e-mail: sastry@eecs.berkeley.edu).

Publisher Item Identifier S 0018-9286(00)04165-9.

0018–9286/00$10.00 © 2000 IEEE

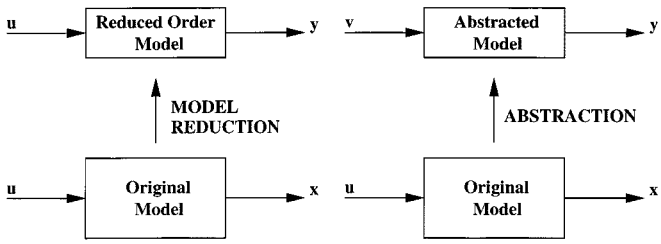

Fig. 2. Model reduction versus abstraction

used in backstepping designs [14]. The fact that the aggregation map sends trajectories of (1.1) to trajectories of (1.2) will enable us to propagate controllability from the micromodel to the macromodel.

Aggregation, however, is not independent of the functionality of the layer at which the abstracted system will be used. Therefore, when an abstracted model is extracted from a more detailed model, one would also like to ensure that certain properties propagate from the macromodel to the micromodel. The properties that are of interest at each layer may include optimality, controllability, stabilizability, and trajectory tracking. If one considers the property of controllability, then one would like to determine conditions under which controllability of the abstracted system (1.2) implies controllability of system (1.1). Obtaining such conditions would ensure that the macromodel is a *consistent abstraction* of the micromodel in the sense that controllability requests from the macromodel are *implementable* by the micromodel. Such conditions will serve as good design principles for hierarchical control systems. Different properties may require different conditions. For example, the notions of consistency [23], dynamic consistency [6] and hierarchical consistency [38] have been defined in order to ensure feasible execution of high-level objectives for discrete event systems. In this paper, we will focus on controllability of linear control systems and characterize consistent linear abstractions. More precisely, we will solve the following problem.

Problem 1.2: Given the linear control system

$$\dot{x} = Ax + Bu, \qquad x \in \mathbb{R}^n \quad u \in \mathbb{R}^m \tag{1.3}$$

characterize linear quotient maps $y = Cx$, so that the abstracted linear system

$$\dot{y} = Fy + Gv, \qquad y \in \mathbb{R}^p \quad v \in \mathbb{R}^k \tag{1.4}$$

is controllable if and only if (iff) system (1.3) is controllable.

In addition to hierarchical control, the above ideas could also be useful in the analysis of complex systems. In order to tackle the complexity involved in verifying that a given large-scale system satisfies certain properties, one tries to extract a simpler but qualitatively equivalent abstracted system. Checking the desired property on the abstracted system should be *equivalent* or *sufficient* to checking the property on the original system. The area of computer aided verification, which must be credited with this notion of abstraction, typically faces problems of exponential complexity and abstractions are frequently used for complexity reduction [9], [13], [21], [30]. Depending on the property, special graph quotients which preserve the property of interest are constructed. More recently, a methodology for constructing finite graph quotients which have equivalent reachability properties with analytic vector fields is presented in [19], [20]. A similar construction which characterizes reachability of a continuous system in terms of an associated discrete system may be found in [8].

In this spirit, and after having characterized consistent linear abstractions, we obtain a hierarchical controllability criterion which has computational and conceptual advantages over the Kalman rank condition and the Popov–Belevitch–Hautus (PBH) tests for large-scale systems. Intuitively, instead of checking controllability of a large-scale system, we construct a sequence of consistent abstractions and then check the controllability of a system, which is much smaller in size. Consistency will then propagate controllability along this sequence of abstractions from the simpler quotient system to the original complex system. The computational advantages of this approach are verified by recovering the best of the known controllability algorithms from numerical linear algebra [11], [12] as a special case of the hierarchical controllability criterion.

The structure of this paper is as follows. In Section II, we review some standard differential geometric concepts and the notion of Φ-related vector fields. Section III generalizes these notions for control systems and establishes the connection between trajectories of Φ-related control systems. In Section IV, we define consistent abstractions and in Section V, we restrict these notions to linear abstractions and characterize consistent linear abstractions. These results are used in Section VI in order to obtain a hierarchical controllability criterion. Finally, Section VII discusses many interesting directions for further research.

II. Φ-Related Vector Fields

We first review some basic facts from differential geometry. The reader may wish to consult numerous books on the subject such as [1], [24], [33]. Let M be a differentiable manifold and T_pM be the tangent space of M at $p \in M$. We denote by $TM = \bigcup_{p \in M} T_pM$ the tangent bundle of M and by π the canonical projection map π: $TM \longrightarrow M$ taking a tangent vector $X_p \in T_pM \subset TM$ to the point $p \in M$.

Now let M and N be smooth manifolds and Φ: $M \longrightarrow N$ be a smooth map. Let $p \in M$ and let $q = \Phi(p) \in N$. We push forward tangent vectors from T_pM to T_qN using the induced push forward map Φ_*: $T_pM \longrightarrow T_qN$. A vector field on a manifold M is a smooth map X: $M \longrightarrow TM$ which assigns to each point p of M a tangent vector in T_pM. Let $I \subseteq \mathbb{R}$ be an open interval containing the origin. An integral curve of a vector field is a smooth curve c: $I \longrightarrow M$ whose tangent at each point is identically equal to the vector field at that point. Therefore, an integral curve satisfies $c' = c_*(1) = X \circ c(t)$ for all $t \in I$ where $c_*(1)$ denotes $c_*(d/dt)$.

An abstraction or aggregation map is a map Φ: $M \longrightarrow N$, which we will assume to be surjective.[1] Given a vector field X on manifold M and a smooth map Φ: $M \longrightarrow N$, not necessarily a diffeomorphism, the push forward of X by Φ_* is generally not a well-defined vector field on N. This leads to the concept of Φ-related vector fields.

Definition 2.1 (Φ-Related Vector Fields): Let X and Y be vector fields on manifolds M and N, respectively, and Φ: $M \longrightarrow N$ be a smooth map. Then X and *Y are Φ-related iff*

$$\Phi_* \circ X = Y \circ \Phi. \tag{2.1}$$

If Φ is not surjective, then X may be Φ-related to many vector fields on N. If Φ is a smooth surjection from M to N, then given a vector field X on a manifold M, the push forward of X by Φ_* is a well-defined vector field on N only if $\Phi_*(X_{p_1}) = \Phi_*(X_{p_2})$ whenever $\Phi(p_1) = \Phi(p_2)$ for any two points p_1,$p_2 \in M$.

Example 2.2: Consider, for example, the linear vector field

$$\dot{x} = Ax, \qquad x \in \mathbb{R}^n \tag{2.2}$$

and the onto, linear quotient map $y = Cx$. Then, in order to obtain a well-defined quotient vector field

$$\dot{y} = Fy, \qquad y \in \mathbb{R}^m \tag{2.3}$$

by C-relatedness we must have $CAx = FCx$ for all $x \in \mathbb{R}^n$. But for $x \in \mathrm{Ker}(C) = \{x \in \mathbb{R}^n \mid Cx = 0\}$ we must have $CAx = F(Cx) = 0$, and hence $Ax \in \mathrm{Ker}(C)$. Thus, a necessary condition to obtain a well-defined quotient vector field is

$$A\mathrm{Ker}(C) \subseteq \mathrm{Ker}(C). \tag{2.4}$$

It turns out that this is also sufficient for the existence of a unique quotient vector field [37].

The following well-known theorem gives us a condition on the integral curves of two Φ-related vector fields. A proof may be found in [1].

Theorem 2.3 (Integral Curves of Φ-Related Vector Fields): Let X and Y be vector fields on M and N respectively and let Φ: $M \longrightarrow N$ be a smooth map. Then vector fields X and Y are Φ-related iff for every integral curve c of X, $\Phi \circ c$ is an integral curve of Y.

If Σ_X and Σ_Y denote all integral curves of vector fields X and Y, respectively, then Theorem 2.3 simply states that X and Y are Φ-related iff $\Phi(\Sigma_X) \subseteq \Sigma_Y$. Therefore Y overapproximates the collection of curves $\Phi(\Sigma_X)$ and allows redundant evolutions. This is the notion of abstraction of dynamical systems defined in [27]. Instead of checking reachability of vector field X, it is *sufficient* to check it on Y, which is of smaller dimension. If the map Φ is surjective, then under some technical assumptions, it is clear that if X and Y are Φ-related vector fields then $\Phi(\Sigma_X) = \Sigma_Y$. In that case, checking reachability properties of vector field X is *equivalent* to checking reachability on vector field Y.

Even though Φ-relatedness of vector fields is a rather restrictive condition, the above discussion provides the correct conceptual framework for generalizing these concepts to control systems, where due to the freedom of control inputs the equivalent conditions will not be as restrictive.

[1]Note that any map Φ gives rise to an equivalence relation by defining states x and y equivalent if $\Phi(x) = \Phi(y)$. In order for the resulting quotient space to have a manifold structure, the equivalence relation must be regular [1].

III. Control-System Abstractions

In this section, the notions of Section II for vector fields are extended to control systems. We will develop such notions for rather general control systems, since it does not require more effort to do so. In addition, generality will ensure that the concepts of this section do not depend on the particular system structure. We first present a global and coordinate-free description of control systems which is due to Brockett [4], [5] and can also be found in [25]. This global description is based on the notion of fiber bundles, which are defined first.

Definition 3.1 (Fiber Bundles): A fiber bundle is a five-tuple $(B, M, \pi, U, \{O_i\}_{i\in I})$ where B, M, U are smooth manifolds called the total space, the base space and the standard fiber, respectively. The map π: $B \longrightarrow M$ is a surjective submersion and $\{O_i\}_{i\in I}$ is an open cover of M, such that for every $i \in I$, there exists a diffeomorphism Ψ_i: $\pi^{-1}(O_i) \longrightarrow O_i \times U$ satisfying $\pi_o \circ \Psi_i = \pi$, where π_o is the projection from $O_i \times U$ to O_i. The submanifold $\pi^{-1}(p)$ is called the fiber at $p \in M$. If all the fibers are vector spaces of constant dimension, then the fiber bundle is called a vector bundle.

Definition 3.2 (Control Systems): A control system $S = (B, F)$ consists of a fiber bundle π: $B \longrightarrow M$ called the control bundle and a smooth map F: $B \longrightarrow TM$ which is fiber preserving, that is $\pi' \circ F = \pi$, where π': $TM \longrightarrow M$ is the tangent bundle projection.

Essentially, the base manifold M of the control bundle is the state space and the fibers $\pi^{-1}(p)$ can be thought of as the state dependent control spaces. Given the state p and the input, the map F selects a tangent vector from T_pM. The notion of trajectories of control systems is now defined.

Definition 3.3 (Trajectories of Control Systems): A smooth curve c: $I \longrightarrow M$ is called a trajectory of the control system $S = (B, F)$ if there exists a curve c^B: $I \longrightarrow B$ satisfying

$$\pi \circ c^B = c, \qquad c' = c_*(1) = F \circ c^B.$$

In local (bundle) coordinates, Definition 3.3 simply says that a trajectory of a control system is a curve x: $I \to M$ for which there exists a function u: $I \to U$ satisfying, $\dot{x} = F(x, u)$. Note that even though Definition 3.3 assumes c to be smooth, the bundle curve c^B is not necessarily smooth. The definition, therefore, allows nonsmooth control inputs as long as the projection $\pi \circ c^B = c$ is smooth. We are now in a position to define Φ-related control systems in a manner similar to Definition 2.1 for vector fields.

Definition 3.4 (Φ-Related Control Systems): Let $S_M = (B_M, F_M)$ with π_M: $B_M \longrightarrow M$ and $S_N = (B_N, F_N)$ with π_N: $B_N \longrightarrow N$ be two control systems. Let Φ: $M \longrightarrow N$ be a smooth map. Then control systems S_M and S_N are Φ-related iff for every $p \in M$

$$\Phi_* \circ F_M(\pi_M^{-1}(p)) \subseteq F_N(\pi_N^{-1}(\Phi(p))). \tag{3.1}$$

Control system S_N will be referred to as an *abstraction* of control system S_M ([27]). Condition (3.1) states that for each $p \in$

M the left-hand side of (3.1) first takes the input space available at p, and pushes it through F_M to obtain all possible tangent directions of the control system S_M at p. This set of tangent directions is pushed through Φ_* to obtain a set of tangent vectors in $T_{\Phi(p)}N$. In order for S_M and S_N to be Φ-related, this set must be contained in the image under F_N of the input space available at $\Phi(p)$. Note that many control systems S_N may be Φ-related to S_M as the set of tangent vectors on N that must be captured, can be generated using many control parameterizations.

It is easy to show that Φ-relatedness is transitive. Indeed, if Φ_1: $M_1 \to M_2$, Φ_2: $M_2 \to M_3$, S_{M_1} is Φ_1-related to S_{M_2}, and S_{M_2} is Φ_2-related to S_{M_3}, then S_{M_1} is $\Phi_2 \circ \Phi_1$-related to S_{M_3}. It therefore makes sense to consider a sequence of Φ-related systems. In addition, given M, N, a map Φ: $M \to N$ and a system S_M, one can put a partial order on all possible Φ-related systems S_N, where the partial ordering arises from pointwise subset inclusion of the right-hand side of (3.1) (see [27]).

To see that Definition 3.4 is a generalization of Definition 2.1, consider vector fields X_M on M and X_N on N. Then X_M and X_N can be thought of as trivial control systems on M and N respectively by letting $B_M = M$, $B_N = N$, $\pi_M = id_M$, $\pi_N = id_N$, and $F_M = X_M$, $F_N = X_N$, where id_M, id_N are the identity maps on M and N, respectively. Condition (3.1) becomes $\Phi_* \circ X_M(p) = X_N \circ \Phi(p)$, which is Definition 2.1 of Φ-related vector fields.

The following proposition, which is an immediate consequence of Definition 3.4, shows that every control or dynamical system is Φ-related to some control system for any map Φ.

Proposition 3.5: Given any control system $S_M = (B_M, F_M)$ and any smooth map Φ: $M \longrightarrow N$, then there exists a control system $S_N = (B_N, F_N)$ which is Φ-related to S_M. In particular, every vector field X on M is Φ-related to some control system S_N.

Proof: Given S_M, construct S_N by simply letting $B_N = TN$ and F_N: $TN \longrightarrow TN$ equal the identity. Then Condition (3.1) is trivially satisfied. Thus $S_N = (B_N, F_N)$ is Φ-related to S_M. □

The concept of Φ-related control systems is a generalization of the notion of *projectable* control systems defined in [18], [22]. A control system is projectable, essentially, when each vector field corresponding to a fixed input value is Φ-related to some vector field. Definition 3.4, instead of globally pushing a vector field for each fixed value of the control input, takes a pointwise approach by pushing forward all possible tangent directions at a state for all possible inputs available at that state. The following example illustrates that Φ-related control systems are not necessarily projectable.

Example 3.6: Consider the double integrator

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = u$$

with $x_1, x_2, u \in \mathbb{R}$ and the projection $\Phi(x_1, x_2) = x_1$. Using Definition 3.4, we obtain that

$$\dot{x}_1 = x_2$$

is a valid Φ-related system. The double integrator, however, is not projectable in the sense of [22], [18] with respect to this map as for any fixed value of u, the vector field $[x_2 \;\; u]^T$ is not Φ-related to any vector field on $\mathbb{R}$. For the nonlinear control system

$$\dot{x}_1 = f_1(x_1, x_2), \qquad \dot{x}_2 = f_2(x_1, x_2, u)$$

with states x_1, x_2, input u, and the projection $\Phi(x_1, x_2) = x_1$, a Φ-related system is

$$\dot{x}_1 = f_1(x_1, x_2)$$

with state x_1, but where x_2 is now thought of as an input. This is the notion of virtual inputs used in backstepping designs [14]. A more constructive methodology for generating abstractions of linear systems will be presented in Section V.

The following theorem should be thought of as a generalization of Theorem 2.3 for control systems.

Theorem 3.7 (Trajectories of Φ-Related Control Systems): Let $S_N = (B_N, F_N)$ and $S_M = (B_M, F_M)$ be two control systems and Φ: $M \longrightarrow N$ be a smooth map. Then S_M and S_N are Φ-related iff for every trajectory c_M of S_M, $\Phi \circ c_M$ is a trajectory of S_N.

Proof: (Sufficiency) Assume that S_M and S_N are Φ-related, and thus, for all $p \in M$ we have

$$\Phi_* \circ F_M(\pi_M^{-1}(p)) \subseteq F_N(\pi_N^{-1}(\Phi(p))). \tag{3.2}$$

Let c_M: $I \longrightarrow M$ be any trajectory of S_M. We must show that $\Phi \circ c_M$ is a trajectory of S_N. We must therefore find a curve c_N^B: $I \longrightarrow B_N$ such that for all $t \in I$ we have $\pi_N \circ c_N^B(t) = \Phi \circ c_M(t)$ and $(\Phi \circ c_M)'(t) = F_N \circ c_N^B(t)$.

Since c_M: $I \longrightarrow M$ is a trajectory of S_M, by Definition 3.3 there exists a curve c_M^B: $I \longrightarrow B_M$ such that for all $t \in I$ we have $\pi_M \circ c_M^B(t) = c_M(t)$ and $c_M'(t) = F_M \circ c_M^B(t)$. By Φ-relatedness of S_M and S_N, we obtain that for all $t \in I$

$$\begin{aligned} \Phi_* \circ F_M(\pi_M^{-1}(c_M(t))) \subseteq F_N(\pi_N^{-1}(\Phi(c_M(t)))) \Longrightarrow \\ \Phi_* \circ F_M \circ c_M^B(t) \in F_N(\pi_N^{-1}(\Phi(c_M(t)))). \end{aligned} \tag{3.3}$$

Condition (3.3) implies that for each $t \in I$, there must exist at least one element $c_N^B(t) \in \pi_N^{-1}(\Phi(c_M(t)))$ (and thus $\pi_N \circ c_N^B(t) = \Phi \circ c_M(t)$) such that

$$\begin{aligned} \Phi_* \circ F_M \circ c_M^B(t) &= F_N \circ c_N^B(t) \\ \Phi_* \circ c_M'(t) &= F_N \circ c_N^B(t) \\ (\Phi \circ c_M)'(t) &= F_N \circ c_N^B(t). \end{aligned}$$

Therefore $\Phi \circ c_M$ is a trajectory of S_N.

(Necessity) Assume that for every trajectory c_M: $I \longrightarrow M$ of S_M, $\Phi \circ c_M$ is a trajectory of S_N. Now for any point $p \in M$ let

$$Y_{\Phi(p)} \in \Phi_*(F_M(\pi_M^{-1}(p))) \tag{3.4}$$

We must show that $Y_{\Phi(p)} \in F_N(\pi_N^{-1}(\Phi(p)))$. We can write $Y_{\Phi(p)} = \Phi_*(X_p)$ for some (not necessarily unique) tangent vector $X_p \in F_M(\pi_M^{-1}(p))$. Then there exists a trajectory c_M: $I \longrightarrow M$, such that at some $t^* \in I$, we have

$$c_M(t^*) = p \tag{3.5}$$

and

$$c_M'(t^*) = X_p. \tag{3.6}$$

Indeed, a curve c_M satisfying (3.5), (3.6) always exists by the existence theorems for differential equations. To show that c_M is a trajectory, we need to find c_M^B: $I \longrightarrow B_M$ such that $\pi \circ c_M^B = c_M$. Let O be a bundle-trivializing neighborhood of p and Ψ: $\pi^{-1}(O) \longrightarrow O \times U$ the trivializing map. There exists $u \in U$, such that $X_p = F_M \circ \Psi^{-1}(p, u)$. Restricting I if necessary we may assume $c_M(I) \subset O$. We can then define the desired curve by $c_M^B(t) = F_M \circ \Psi^{-1}(c_M(t), u)$.

Since c_M is a trajectory of S_M satisfying (3.5), (3.6), then by assumption we have that $\Phi \circ c_M$ is a trajectory of S_N. Therefore, by Definition 3.3, there must exist a curve c_N^B: $I \longrightarrow B_N$, such that for all $t \in I$, we have $\pi_N \circ c_N^B(t) = \Phi \circ c_M(t)$ and $(\Phi \circ c_M)'(t) = F_N \circ c_N^B(t)$. In particular, at $t^* \in I$, we have

$$\begin{aligned}(\Phi \circ c_M)'(t^*) &= F_N \circ c_N^B(t^*)\\ \Phi_* \circ c_M'(t^*) &\in F_N(\pi_N^{-1}(\Phi(c_M(t^*))))\\ Y_p = \Phi_*(X_p) &\in F_N(\pi_N^{-1}(\Phi(p))).\end{aligned}$$

Therefore, at all points $p \in M$, we must have $\Phi_* \circ F_M(\pi_M^{-1}(p)) \subseteq F_N(\pi_N^{-1}(\Phi(p)))$, and thus S_M and S_N are Φ-related. This completes the proof. □

If Σ_{S_M} and Σ_{S_N} denote all trajectories of control systems S_M and S_N, respectively, then Theorem 3.7 simply states that S_M and S_N are Φ-related iff $\Phi(\Sigma_{S_M}) \subseteq \Sigma_{S_N}$. The quotient system therefore overapproximates the abstracted trajectories of the original system which may result in trajectories that the macrosystem S_N may generate but are infeasible in the micromodel S_M.

Theorem 3.7 does not guarantee that the curve $c_N^B(t)$ is a smooth curve. The main obstacle for generating smooth $c_N^B(t)$ is whether the map F_N: $B_N \longrightarrow TN$ is an embedding. An example showing that F_N being only an immersion is not enough can be found in [29]. The following theorem shows that F_N being an injective embedding is sufficient to guarantee smoothness of the $c_N^B(t)$. Note that requiring F_N to be an injective embedding implies that the dimension of the input space is less than the dimension of TN and thus there are no redundant inputs (which covers the cases of interest). In particular, if the control system S_N is affine in the controls then this is equivalent to saying that the "controlled" vector fields are linearly independent at each point. That is, if we write the system in local (bundle) coordinates of B_N and local (vector bundle) coordinates of TN as

$$\dot{x} = f(x) + \sum_{i=1}^{k} g_i(x) u_i$$

then for each x, the vectors $g_1(x), \cdots, g_k(x)$ are linearly independent.

Theorem 3.8 (Control Input Smoothness): Let $S_N = (B_N, F_N)$ and $S_M = (B_M, F_M)$ be two Φ-related control systems where F_N: $B_N \longrightarrow TN$ is an injective embedding. Let c_M: $I \longrightarrow M$ be a trajectory of S_M and assume that the corresponding c_M^B: $I \longrightarrow B_M$ is smooth. Then there exists a smooth curve c_N^B: $I \longrightarrow B_N$ such that for all $t \in I$, $\pi_N \circ c_N^B(t) = \Phi \circ c_M(t)$ and $F_N \circ c_N^B(t) = (\Phi \circ c_M)'(t)$.

Proof: Since S_M and S_N are Φ-related we have $\Phi_* \circ F_M \circ c_M^B(t) \in F_N(\pi_N^{-1}(\Phi(c_M(t))))$ for each $t \in I$. Moreover, since by assumption F_N is an embedding, the space B_N is diffeomorphic to its image under F_N. We can then define

$$c_N^B(t) = F_N^{-1}(\Phi_* \circ F_M \circ c_M^B(t))$$

which is clearly smooth and satisfies the desired properties. □

IV. Consistent Control Abstractions

In general, we are not simply interested in abstracting systems but also propagating properties between the original and abstracted model. In this paper, we focus on various notions of controllability.

Definition 4.1 (Controllability): Let $S = (B, F)$ be a control system on M. For $p \in M$, define $\mathrm{Reach}(p, S)$ to be the set of points $q \in M$ for which there exists a trajectory c: $I \longrightarrow M$ of S, such that for some $t_1, t_2 \in I$, we have $c(t_1) = p$ and $c(t_2) = q$. The control system S is called controllable iff for all $p \in M$, $\mathrm{Reach}(p, S) = M$.

Theorem 3.7 allows us to always propagate the property of controllability from the micromodel to the macromodel for any aggregation map.

Theorem 4.2 (Controllability Propagation): Let control systems $S_M = (B_M, F_M)$ and $S_N = (B_N, F_N)$ be Φ-related with respect to some smooth surjection Φ: $M \longrightarrow N$. Then for all $p \in M$

$$\Phi(\mathrm{Reach}(p, S_M)) \subseteq \mathrm{Reach}(\Phi(p), S_N).$$

Thus, if S_M is controllable then S_N is controllable.

Proof: Consider any $p \in M$ and let $q \in \Phi(\mathrm{Reach}(p, S_M))$. Then there exists $p_1 \in \Phi^{-1}(q)$, with $p_1 \in \mathrm{Reach}(p, S_M)$. Thus, there exists a trajectory c_M of S_M, such that $c_M(t_1) = p$ and $c_M(t_2) = p_1$. By Φ-relatedness, the curve $\Phi \circ c_M$ is a trajectory of S_N which connects $\Phi(c_M(t_1)) = \Phi(p)$ and $\Phi(c_M(t_2)) = \Phi(p_1) = q$. Therefore, $q \in \mathrm{Reach}(\Phi(p), S_N)$.

If S_M is controllable, then for all $p \in M$, we have $\mathrm{Reach}(p, S_M) = M$. But then $\Phi(\mathrm{Reach}(p, S_M)) = \Phi(M) = N = \mathrm{Reach}(\Phi(p), S_N)$. Thus, S_N is controllable. □

Note that Theorem 4.2 is true regardless of the structure of the aggregation map Φ. From a hierarchical perspective, the reverse question is a lot more interesting since it would guarantee that controllability requests are implementable by the lower-level system. In order to arrive at this goal, we define the notions of implementability and consistency. We also give descriptions of those concepts in terms of reachable sets.

Definition 4.3 (Controllability Implementation): Let $S_M = (B_M, F_M)$ and $S_N = (B_N, F_N)$ be two control systems and Φ: $M \longrightarrow N$ be a smooth surjection. Then S_N is implementable[2] by S_M iff whenever there is a trajectory of S_N connecting $q_1 \in N$ and $q_2 \in N$, then there exist $p_1 \in \Phi^{-1}(q_1)$ and $p_2 \in \Phi^{-1}(q_2)$ and a trajectory of S_M connecting p_1 and p_2.

[2]In this paper, we only consider implementation of controllability requests. Thus, implementability will refer to controllability implementation.

Implementability is therefore an existential property. If one thinks of the map Φ as a quotient map, then implementability requires that a reachability request is implementable by at least one member of the equivalence class. It is clear from Definition 4.3 that implementability is transitive, that is, if S_{M_1} is implementable by S_{M_2} with respect to Φ_1, and S_{M_2} is implementable by S_{M_3} with respect to Φ_2, then S_{M_1} is implementable by S_{M_3} with respect to $\Phi_1 \circ \Phi_2$. This is important in hierarchical systems which should consist of a sequence of implementable abstractions. It should be noted that the notion of implementability defined above is related to the notion of between-block controllability, defined in [6], [8].

Proposition 4.4 (Implementation Condition): Consider control systems $S_M = (B_M, F_M)$ and $S_N = (B_N, F_N)$ and a smooth surjection Φ: $M \longrightarrow N$. Then S_N is implementable by S_M iff for all $q \in N$

$$\mathrm{Reach}(q, S_N) \subseteq \Phi(\mathrm{Reach}(\Phi^{-1}(q), S_M)) \tag{4.1}$$

where $\mathrm{Reach}(\Phi^{-1}(q), S_M) = \cup_{p \in \Phi^{-1}(q)} \mathrm{Reach}(p, S_M)$.

Proof: Let $q' \in \mathrm{Reach}(q, S_N)$. By implementability, there exists a trajectory of S_M connecting some $p \in \Phi^{-1}(q)$ to some $p' \in \Phi^{-1}(q')$ and thus $p' \in \mathrm{Reach}(p, S_M)$. But then $q' = \Phi(p') \in \Phi(\mathrm{Reach}(p, S_M)) \subseteq \Phi(\mathrm{Reach}(\Phi^{-1}(q), S_M))$.

Conversely, let $q_2 \in \mathrm{Reach}(q_1, S_N)$ for some $q_1 \in N$. By assumption

$$\begin{aligned} q_2 &\in \Phi(\mathrm{Reach}(\Phi^{-1}(q_1), S_M)) \\ &= \Phi(\cup_{p_1 \in \Phi^{-1}(q_1)} \mathrm{Reach}(p_1, S_M)) \\ &= \cup_{p_1 \in \Phi^{-1}(q_1)} \Phi(\mathrm{Reach}(p_1, S_M)). \end{aligned}$$

But then there must exist at least one $p_1' \in \Phi^{-1}(q_1)$ such that $q_2 \in \Phi(\mathrm{Reach}(p_1', S_M))$ which in turn implies that there exists $p_2' \in \mathrm{Reach}(p_1', S_M)$ with $\Phi(p_2) = q_2$ and thus S_N is implementable by S_M. This completes the proof. □

We will mostly be interested in implementability of Φ-related systems, in which case the above inclusion becomes an equality, by Theorem 4.2.

Implementability may depend on the particular element chosen from the equivalence class $\Phi^{-1}(q)$. In order to make the controllability request well-defined, it would have to be independent of the particular element chosen from the equivalence class. This leads to the important notion of consistency.

Definition 4.5 (Consistency): Let $S_M = (B_M, F_M)$ be a control system on M and let Φ: $M \longrightarrow N$ be a smooth surjection. Then S_M is called consistent with respect to Φ whenever the following holds. If there exists a trajectory of S_M connecting p and q, then for all p' such that $\Phi(p) = \Phi(p')$, there exists a trajectory of S_M connecting p' to some q' with $\Phi(q) = \Phi(q')$.

Note that while implementability is a condition between two systems S_M and S_N, consistency is a condition on a single system with respect to some quotient map Φ. Consistency requires that the ability to reach a particular equivalence class is independent of the chosen element from the initial equivalence class. Notice that $\Phi^{-1}(\Phi(p))$ is the equivalence class of p with respect to Φ.

Proposition 4.6 (Consistency Condition): Consider a control system $S = (B, F)$ on M and a smooth surjection Φ: $M \longrightarrow N$. Then S is consistent with respect to Φ iff for all $p \in M$

$$\Phi(\mathrm{Reach}(\Phi^{-1}(\Phi(p)), S)) = \Phi(\mathrm{Reach}(p, S)). \tag{4.2}$$

Proof: Clearly $\Phi(\mathrm{Reach}(p, S)) \subseteq \Phi(\mathrm{Reach}(\Phi^{-1}(\Phi(p)), S))$ for any $p \in M$. Let $q = \Phi(p')$ with $p' \in \mathrm{Reach}(\Phi^{-1}(\Phi(p)), S)$. There exists $p_0 \in \Phi^{-1}(\Phi(p))$ such that $p' \in \mathrm{Reach}(p_0, S)$. By consistency, since $\Phi(p_0) = \Phi(p)$, there exists $p'' \in \mathrm{Reach}(p, S)$ with $\Phi(p'') = \Phi(p')$. But then $q = \Phi(p'') \in \Phi(\mathrm{Reach}(p, S))$.

Conversely, assume (4.2) holds. Let $q \in \mathrm{Reach}(p, S)$ and $\Phi(p') = \Phi(p)$. Then $\Phi(q) \in \Phi(\mathrm{Reach}(\Phi^{-1}(\Phi(p)), S)) = \Phi(\mathrm{Reach}(p', S))$ and there exists $q' \in \mathrm{Reach}(p', S)$ with $\Phi(q) = \Phi(q')$. □

Consistency does not place any conditions on which element of the final equivalence class the system will be steered to. In some hierarchical systems, this may be acceptable, as the high-level system S_N may be interested in its command having a feasible execution by S_M without being interested about the particular state of S_M, as long as it steers it to the correct equivalence class. This form of generalized output controllability is now defined.

Definition 4.7 (Macrocontrollability): Let $S = (B, F)$ be a control system on M and let Φ: $M \longrightarrow N$ be a smooth surjection. Then S is called macrocontrollable if for all $p \in M$ and any $q \in N$ there exists a trajectory of S connecting p to some $p' \in M$ with $\Phi(p') = q$.

By combining the notions of implementability and consistency, we can propagate some controllability information from the coarser system S_N to the more detailed system S_M.

Proposition 4.8 (Macrocontrollability Propagation): Consider control systems $S_M = (B_M, F_M)$ and $S_N = (B_N, F_N)$ which are Φ-related with respect to the smooth surjection Φ: $M \longrightarrow N$. Assume that S_M is an implementation of S_N, and S_M is consistent. Then S_M is macrocontrollable iff S_N is controllable.

Proof: Let $p \in M$ and $q \in N$ be any points. Let $q_0 = \Phi(p)$. Since S_N is controllable, there exists a trajectory of S_N connecting q_0 and q. Since S_M is an implementation of S_N, there exists a trajectory of S_M connecting some $p_1 \in \Phi^{-1}(q_0)$ and some $p_2 \in \Phi^{-1}(q)$. Moreover, since S_M is also consistent, there is a trajectory of S_M connecting p to some p' with $\Phi(p') = \Phi(p_2) = q$. Therefore, S_M is macrocontrollable. The other direction follows easily from Theorem 4.2. □

In order to propagate full controllability from S_M to S_N, we need a stronger notion of consistency which would be independent from the elements chosen from both the initial and final equivalence class.

Definition 4.9 (Strong Consistency): Let $S_M = (B_M, F_M)$ be a control system on M and Φ: $M \longrightarrow N$ a smooth surjection. Then S_M is called strongly consistent with respect to Φ whenever the following holds. If there exists a trajectory of S_M connecting p and q, then for all p' and for all q' such that $\Phi(p) = \Phi(p')$, $\Phi(q) = \Phi(q')$ there exists a trajectory connecting p' to q'.

Definition 4.9 is weaker than the notion of in-block controllability of [6], [8] as it does not restrict the system to remain

within the equivalence class in order to steer from one element to another in the same class.

Proposition 4.10 (Strong Consistency Condition): Consider control system $S = (B, F)$ on M and the smooth surjection Φ: $M \longrightarrow N$. Then S is strongly consistent with respect to Φ iff for all $p \in M$

$$\mathrm{Reach}(p, S) = \Phi^{-1}(\Phi(\mathrm{Reach}(\Phi^{-1}(\Phi(p)), S))). \tag{4.3}$$

Proof: The inclusion $\mathrm{Reach}(p, S) \subseteq \Phi^{-1}(\Phi(\mathrm{Reach}(\Phi^{-1}(\Phi(p)), S)))$ always holds. Let $q \in \Phi^{-1}(\Phi(\mathrm{Reach}(\Phi^{-1}(\Phi(p)), S)))$. Then there exists $q' \in \mathrm{Reach}(\Phi^{-1}(\Phi(p)), S)$ with $\Phi(q') = \Phi(q)$. Let $p' \in \Phi^{-1}(\Phi(p))$ be such that $q' \in \mathrm{Reach}(p', S)$. Since $\Phi(q) = \Phi(q')$ and $\Phi(p) = \Phi(p')$, strong consistency implies $q \in \mathrm{Reach}(p, S)$.

Conversely, assume (4.3) holds. Let $q \in \mathrm{Reach}(p, S)$ and p', q' be such that $\Phi(p') = \Phi(p)$, $\Phi(q') = \Phi(q)$. Then

$$\begin{aligned} q' \in \Phi^{-1}(\Phi(q)) &\subseteq \Phi^{-1}(\Phi(\mathrm{Reach}(p, S))) \\ &\subseteq \Phi^{-1}(\Phi(\mathrm{Reach}(\Phi^{-1}(\Phi(p)), S))) \\ &= \Phi^{-1}(\Phi(\mathrm{Reach}(\Phi^{-1}(\Phi(p')), S))) \\ &= \mathrm{Reach}(p', S). \end{aligned}$$

Therefore, S is strongly consistent. □

Since strong consistency is a more restrictive notion, it is natural that Condition (4.3) is stronger than Condition (4.2) for consistency.

Proposition 4.11 (Controllability Equivalence): Consider control systems $S_M = (B_M, F_M)$ and $S_N = (B_N, F_N)$ which are Φ-related with respect to smooth surjection Φ: $M \longrightarrow N$. Assume that S_M is an implementation of S_N, and S_M is strongly consistent. Then S_N is controllable iff S_M is controllable.

Proof: Let $p_1, p_2 \in M$ any points. Let $q_1 = \Phi(p_1)$ and $q_2 = \Phi(p_2)$. Since S_N is controllable, there exists a trajectory of S_N connecting q_1 and q_2. Since S_M is an implementation of S_N, there exists a trajectory of S_M connecting some $p_1' \in \Phi^{-1}(q_1)$ and some $p_2' \in \Phi^{-1}(q_2)$. Then, since S_M is strongly consistent, there is a trajectory of S_M connecting p_1 to p_2. The other direction is given by Theorem 4.2. □

In this section, we identified the relevant notions for the study of controllability in Φ-related systems. We also described them for arbitrary systems in terms of reachable sets. In the following sections, we will illustrate these notions (see Example 5.7), and give concrete characterizations of these concepts for linear systems. Moreover, we show how to use them to construct explicit Φ-related systems with the desirable properties.

V. Consistent Linear Abstractions

The notion of Φ-related control systems is now specialized for the case of linear time-invariant systems with linear aggregation maps. Consider the linear control systems

$$\begin{aligned} (\Sigma_1) \qquad \dot{x} &= Ax + Bu \\ (\Sigma_2) \qquad \dot{y} &= Fy + Gv \end{aligned}$$

with $x \in \mathbb{R}^n$, $u \in \mathbb{R}^k$, $y \in \mathbb{R}^m$, $v \in \mathbb{R}^l$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times k}$, $F \in \mathbb{R}^{m\times m}$, $G \in \mathbb{R}^{m\times l}$, and the surjective linear aggregation map $y = Cx$. Then by Definition 3.4, Σ_1 and Σ_2 are C-related if for all $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^k$ there exists $v \in \mathbb{R}^l$, such that

$$C(Ax + Bu) = FCx + Gv. \tag{5.1}$$

By Proposition 3.5, given any control system and any map Φ, there always exists another control system which is Φ-related to it. We are interested, however, in a constructive methodology for generating Φ-related systems. The following proposition gives us a systematic way to generate C-related linear abstractions of a linear system with respect to a linear aggregation map $y = Cx$.

Proposition 5.1 (Construction of Linear Abstractions): Consider the linear system

$$(\Sigma_1) \qquad \dot{x} = Ax + Bu$$

and a surjective map $y = Cx$. Let

$$(\Sigma_2) \qquad \dot{y} = Fy + Gv$$

be the system where

$$F = CAC^+, \qquad G = [CB \;\; CAv_1 \;\; \cdots \;\; CAv_r]$$

with $C^+ = C^T(CC^T)^{-1}$, the pseudoinverse of C, and $v_1, \cdots, v_r$ spanning $\mathrm{Ker}(C)$. *Then Σ_1 and Σ_2 are C-related.*

Proof: We need to show that for all $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^k$, there exists $v \in \mathbb{R}^l$, such that

$$C(Ax + Bu) = Fy + Gv$$

or equivalently

$$Gv = CBu + (CA - FC)x.$$

Clearly, CBu belongs in the range of G for all u. Decompose $\mathbb{R}^n = \mathrm{Ker}(C) \oplus \mathrm{Ker}(C)^\perp$. If $x \in \mathrm{Ker}(C)^\perp$, then $C^+Cx = x$, and thus

$$(CA - FC)x = (CA - CAC^+C)x = 0.$$

If $x \in \mathrm{Ker}(C)$ then $(CA - FC)x = CAx$, which also belongs in the range of G. □

It is immediate from Proposition 5.1 that an abstraction of a linear system with respect to a linear aggregation map can be also a linear system. Proposition 5.1 is interesting as it constructively generates for linear systems the so-called *virtual inputs* used in backstepping designs [14]. In particular, if the aggregation map is a projection on some of the states, then the states that are ignored appear as inputs at the abstracted system. As another special case, suppose that $\mathrm{Ker}(C) = Im(B)$. Then we can take as $v_1, \cdots, v_r$ the columns of B. The input vectors for Σ_2 are the images under C of the vectors Av_i, which correspond to the next r vectors in the controllability matrix of Σ_1. The following example illustrates the proposition.

Example 5.2: Consider again the double integrator

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = u$$

and the projection $y = x_1$. So here $A = \left[\begin{smallmatrix} 0 & 1 \\ 0 & 0 \end{smallmatrix}\right]$, $B = [0 \quad 1]^T$, and $C = [1 \quad 0]$. Then $\mathrm{Ker}(C) = span\{[0 \quad 1]^T\}$ and the procedure

of Proposition 5.1 results in $F = 0$, $G = [0 \;\; 1]$. We can reduce further to G= 1 and get

$$\dot{y} = v.$$

Now consider the dynamics of the oscillating vector field

$$\dot{x}_1 = x_2, \qquad \dot{x}_2 = -x_1$$

with the same projection map $y = x_1$. Here $A = \left[\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right]$. Then Proposition 5.1 results in the same control system (or better, differential inclusion)

$$\dot{y} = v.$$

The fact that the coarser system may have control inputs, even though the original one did not, is clearly undesirable. However, as will be shown, this will be taken care of by the notion of consistency.

From linear systems theory we know that for the linear system

$$(\Sigma_1) \qquad \dot{x} = Ax + Bu$$

the reachable space from any $x_0 \in \mathbb{R}^n$ is given by

$$\begin{aligned} \mathrm{Reach}(x_0, \Sigma_1) &= \bigcup_{T \geq 0} e^{AT} x_0 + \mathrm{Reach}(0, \Sigma_1) \\ &= \bigcup_{T \geq 0} e^{AT} x_0 + \mathcal{R}(A, B) \end{aligned} \tag{5.2}$$

where

$$\mathcal{R}(A, B) = \mathrm{Im}[B \; AB \; \cdots \; A^{n-1}B]$$

is the reachable space from the origin. In particular, system Σ_1 is controllable iff $\mathcal{R}(A, B) = \mathbb{R}^n$. As a corollary of Theorem 4.2, we obtain the following result.

Theorem 5.3 (Controllability Propagation for Linear Abstractions): Consider the linear systems

$$\begin{aligned} (\Sigma_1) \qquad \dot{x} &= Ax + Bu \\ (\Sigma_2) \qquad \dot{y} &= Fy + Gv \end{aligned}$$

which are C-related which respect to the surjective map $y = Cx$. Then

$$C\mathcal{R}(A, B) \subseteq \mathcal{R}(F, G)$$

In particular, if Σ_1 is controllable then Σ_2 is controllable.

Proof: Simple application of Theorem 4.2. □

In order to propagate controllability from the linear system Σ_2 to Σ_1, the notions of implementability and consistency were defined in Section IV.

Proposition 5.4 (Implementability Characterization for Linear Systems): Consider two linear systems

$$\begin{aligned} (\Sigma_1) \qquad \dot{x} &= Ax + Bu \\ (\Sigma_2) \qquad \dot{y} &= Fy + Gv \end{aligned}$$

and surjective map $y = Cx$. Then Σ_2 is implementable by Σ_1 iff for all y we have

$$\bigcup_{T \geq 0} e^{FT} y + \mathcal{R}(F, G) \subseteq \bigcup_{T \geq 0} \bigcup_{x \in C^{-1}(y)} Ce^{AT} x + C\mathcal{R}(A, B). \tag{5.3}$$

Proof: Follows from Proposition 4.4 and (5.2). □

The following theorem gives a simple characterization of consistency for linear systems in terms of subspace invariance.

Theorem 5.5 (Consistency Characterization for Linear Systems): The linear system

$$(\Sigma_1) \qquad \dot{x} = Ax + Bu$$

is consistent with respect to the map $y = Cx$ iff

$$A\mathrm{Ker}(C) \subseteq \mathrm{Ker}(C) + \mathcal{R}(A, B) \tag{5.4}$$

Proof: First, notice that for any set $\mathcal{V} \subseteq \mathbb{R}^n$, we have $C^{-1}(C\mathcal{V}) = \mathcal{V} + \mathrm{Ker}(C)$.

Assume (5.4) holds. We must show consistency Condition (4.2), which for linear systems requires, for all x, that $C(\mathrm{Reach}(x + \mathrm{Ker}(C), \Sigma_1)) = C(\mathrm{Reach}(x, \Sigma_1))$, or equivalently

$$C\left(\bigcup_{T \geq 0} e^{AT}(x + \mathrm{Ker}(C)) + \mathcal{R}(A, B)\right) = C\left(\bigcup_{T \geq 0} e^{AT} x + \mathcal{R}(A, B)\right). \tag{5.5}$$

Clearly, $C\,\mathrm{Reach}(x, \Sigma_1) \subseteq C(\mathrm{Reach}(x + \mathrm{Ker}(C), \Sigma_1)$. Condition (5.4) and A-invariance of $\mathcal{R}(A, B)$ imply that for all $T \geq 0$ we have

$$e^{AT}\mathrm{Ker}(C) \subseteq \mathrm{Ker}(C) + \mathcal{R}(A, B)$$

and therefore

$$Ce^{AT}\mathrm{Ker}(C) \subseteq C\mathcal{R}(A, B).$$

This gives the other inclusion, proving consistency.

Conversely, assume that Σ_1 is consistent. Let $x_0 \in \mathrm{Ker}(C)$. From (5.5) with $x = 0$ we get for any $T > 0$ there exists $r \in \mathcal{R}(A, B)$ such that $Ce^{AT}x_0 = Cr$. Therefore, $e^{AT}x_0 = x_0' + r$ for some $x_0' \in \mathrm{Ker}(C)$.

We have therefore shown that for all $T > 0$, $e^{TA}x_0 \in \mathrm{Ker}(C) + \mathcal{R}(A, B)$. By using $de^{tA}/dt = Ae^{tA}$ and taking limits as $T \to 0$, we conclude that $Ax_0 \in \mathrm{Ker}(C) + \mathcal{R}(A, B)$. □

Note that Condition (5.4) is clearly weaker than the well-known condition

$$A\mathrm{Ker}(C) \subseteq \mathrm{Ker}(C) + \mathcal{R}(B)$$

(where $\mathcal{R}(B) = Im(B)$) for $\mathrm{Ker}(C)$ to be a controlled-invariant (or (A,B)-invariant) subspace.

Theorem 5.6 (Strong Consistency Characterization for Linear Systems): The linear system

$$(\Sigma_1) \qquad \dot{x} = Ax + Bu$$

is strongly consistent with respect to the map $y = Cx$ iff

$$\mathrm{Ker}(C) \subseteq \mathcal{R}(A, B). \tag{5.6}$$

Proof: Assume Σ_1 is strongly consistent. Condition 4.3 for linear systems becomes

$$\bigcup_{T\geq 0} e^{AT}x + \mathcal{R}(A, B) = \bigcup_{T\geq 0} e^{AT}(x + \mathrm{Ker}(C)) + \mathcal{R}(A, B) + \mathrm{Ker}(C). \tag{5.7}$$

Using (5.7) with $x = 0$ gives $\mathcal{R}(A, B) \supseteq \mathrm{Ker}(C)$.

Conversely, assume (5.6) holds. By A-invariance of $\mathcal{R}(A, B)$ we get, for all $T \geq 0$

$$e^{AT}\mathrm{Ker}(C) \subseteq \mathcal{R}(A, B).$$

This gives the inclusion

$$\bigcup_{T\geq 0} e^{AT}x + \mathcal{R}(A, B) \supseteq \bigcup_{T\geq 0} e^{AT}(x + \mathrm{Ker}(C)) + \mathcal{R}(A, B) + \mathrm{Ker}(C).$$

The other inclusion always holds. □

Note that by the A-invariance of $\mathcal{R}(A, B)$, Condition (5.6) is indeed stronger than (5.4). Consistency Conditions (5.4) and (5.6) are rather intuitive. Condition (5.4) essentially says that whatever piece of $\mathrm{Ker}(C)$ is not A-invariant can be compensated by $\mathcal{R}(A, B)$. On the other hand, Condition (5.6) is a form of controllability within the equivalence classes. The trajectories of the system which connect two points of the same equivalence class (as defined by C) are not, however, restricted to remain within the equivalence class. The following example illustrates the notions of implementability and consistency.

Example 5.7: Consider the linear system (without controls) $\dot{x} = Ax$, where

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad C = [1 \quad 0]$$

and the C-related (one-dimensional) system $\dot{y} = Fy + Gv$, where $F = 0$ and $G = 1$. We also have

$$\mathrm{Ker}(C) = \mathrm{span}\left\{\begin{bmatrix} 0 \\ 1 \end{bmatrix}\right\}$$
$$A\mathrm{Ker}(C) = \mathrm{span}\left\{\begin{bmatrix} 1 \\ 0 \end{bmatrix}\right\} \not\subseteq \mathrm{Ker}(C).$$

Therefore, the system Σ_1 is not consistent. To show it is implementable we simply solve the system explicitly. Notice that since $\dot{y} = v$, any two points (of $\mathbb{R}$) can be connected by a trajectory of Σ_2 in arbitrary positive time. Let $y_0, y_f \in \mathbb{R}$. The curve

$$x_1(t) = \frac{y_f - y_0}{T}t + y_0, \qquad x_2(t) = \frac{y_f - y_0}{T}$$

is a trajectory of Σ_1 from $\left[\begin{smallmatrix} y_0 \\ y_f - y_0/T \end{smallmatrix}\right]$ to $\left[\begin{smallmatrix} y_f \\ y_f - y_0/T \end{smallmatrix}\right]$ at time T. Therefore, Σ_2 is implementable by Σ_1. Notice that if $y_f \neq y_0$, there is no trajectory of Σ_1 connecting $\left[\begin{smallmatrix} y_0 \\ 0 \end{smallmatrix}\right]$ to any point x with $Cx = y_f$. The reason is that all the points $\left[\begin{smallmatrix} x_1 \\ 0 \end{smallmatrix}\right]$ are equilibria of Σ_1.

In order to propagate some form of controllability from Σ_2 to Σ_1, we need to check two properties, namely implementability and (strong) consistency. Unfortunately, Condition (5.3) is not easy to check since it involves the explicit integration of the differential equation. However, Condition (5.3), in conjunction with consistency Conditions (5.4) or (5.6), results in checkable characterizations of implementations which are also (strongly) consistent. To achieve this, we will need the following lemma.

Lemma 5.8: Let A $(n \times n)$, C $(m \times n)$, F $(m \times m)$ and G $(m \times l)$ be matrices with $l \leq m$ and G of full rank. If for all $x \in \mathbb{R}^n$ $(CA - FC)x \in \mathcal{R}(F, G)$, then for all $t \geq 0$,

$$(Ce^{tA} - e^{tF}C)x \in \mathcal{R}(F, G). \tag{5.8}$$

In particular, the conclusion holds if A, F, are G are the corresponding matrices for the C-related systems Σ_1 and Σ_2.

Proof: We have the following identity for all $t \geq 0$:

$$Ce^{tA} - e^{tF}C = \sum_{j=0}^{\infty}(CA^j - F^jC)\frac{t^j}{j!}. \tag{5.8}$$

We prove by induction the statement

$$(P_j) \qquad \forall x \in \mathbb{R}^n (CA^j - F^jC)x \in \mathcal{R}(F, G).$$

It is clearly true for $j = 0$ and by hypothesis it is also true for $j = 1$. Assume P_i holds for $i \leq j$. We can write

$$(CA^{j+1} - F^{j+1}C)x = (CA^j - F^jC)Ax + F^j(CA - FC)x.$$

By the inductive hypothesis applied to x and Ax, $(CA^j - F^jC)Ax \in \mathcal{R}(F, G)$ and $(CA - FC)x \in \mathcal{R}(F, G)$. But then $F^j(CA - FC)x \in \mathcal{R}(F, G)$ for all j, since $\mathcal{R}(F, G)$ is F-invariant. Therefore

$$(CA^j - F^jC)Ax + F^j(CA - FC)x \in \mathcal{R}(F, G).$$

By taking the limit in (5.8), we conclude the proof. □

Theorem 5.9 (Implementability and Consistency Characterization): Consider the linear systems

$$(\Sigma_1) \qquad \dot{x} = Ax + Bu$$
$$(\Sigma_2) \qquad \dot{y} = Fy + Gv$$

which are C-related which respect to the surjective map $y = Cx$. Then Σ_2 is implementable by Σ_1 and Σ_1 is consistent iff

$$C\mathcal{R}(A, B) = \mathcal{R}(F, G) \tag{5.9}$$

In addition, Σ_2 is implementable by Σ_1 and Σ_1 is strongly consistent iff

$$\mathcal{R}(A, B) = C^{-1}(\mathcal{R}(F, G)). \tag{5.10}$$

Proof: Assume $C\mathcal{R}(A, B) = \mathcal{R}(F, G)$, and thus $\mathcal{R}(F, G) \subseteq C\mathcal{R}(A, B)$. Now let $x \in \mathrm{Ker}(C)$. By C-relatedness, there exists $v \in \mathbb{R}^l$ such that $CAx = FCx + Gv = Gv$ (using $u = 0$ and since $Cx = 0$). So, $CAx \in R(F, G)$ and by assumption, there is $x_1 \in R(A, B)$, such that $Cx_1 = CAx$. Therefore, $Ax - x_1 \in \mathrm{Ker}(C)$ and $Ax = Ax - x_1 + x_1 \in \mathrm{Ker}(C) + R(A, B)$. Thus

$A\text{Ker}(C) \subseteq \text{Ker}(C) + \mathcal{R}(A, B)$ and Σ_1 is consistent. We must now show that Condition (5.3) holds. Consider any

$$\begin{aligned} y_f &= e^{FT} y_0 + r_F^1 \in \text{Reach}(y_0, \Sigma_2) \\ &= \bigcup_{T \geq 0} e^{FT} y_0 + \mathcal{R}(F, G) \end{aligned}$$

with $r_F^1 \in \mathcal{R}(F, G)$. By Lemma 5.8, we have that $e^{FT} y_0 = Ce^{AT} x_0 + Cr_F^2$ for some $r_F^2 \in \mathcal{R}(A, B)$, and for any x_0 with $y_0 = Cx_0$. But then

$$\begin{aligned} y_f &= Ce^{AT} x_0 + r_F^1 + r_F^2 \\ &= Ce^{AT} x_0 + Cr_A \in \bigcup_{T \geq 0} \bigcup_{x \in C^{-1}(y_0)} Ce^{AT} x + C\mathcal{R}(A, B) \\ &= C(\text{Reach}(C^{-1}(y_0), \Sigma_1)) \end{aligned}$$

for some $r_A \in \mathcal{R}(A, B)$ since $\mathcal{R}(F, G) \subseteq C\mathcal{R}(A, B)$. Therefore Σ_2 is implementable by Σ_1.

For the converse notice that, since the systems are C-related, Proposition 5.3 implies $\mathcal{R}(F, G) \supseteq C\mathcal{R}(A, B)$. Moreover, the implementability Condition (5.3) with $y = 0$ gives

$$\mathcal{R}(F, G) \subseteq \bigcup_{T \geq 0} Ce^{AT} \text{Ker}(C) + C\mathcal{R}(A, B)$$

and the consistency Condition (5.5) with $x = 0$ gives

$$\bigcup_{T \geq 0} Ce^{AT} \text{Ker}(C) \subseteq C\mathcal{R}(A, B).$$

These two combined give $\mathcal{R}(F, G) \subseteq C\mathcal{R}(A, B)$. This concludes the proof of the first equivalence.

Now assume that $\mathcal{R}(A, B) = C^{-1}(\mathcal{R}(F, G))$. Then $C\mathcal{R}(A, B) = \mathcal{R}(F, G)$ and therefore Σ_1 implements Σ_2. Since $0 \in \mathcal{R}(F, G)$ we also have $\text{Ker}(C) \subseteq \mathcal{R}(A, B)$. Therefore, Σ_1 is strongly consistent. If Σ_1 is strongly consistent and implements Σ_2, then Σ_1 is also consistent and therefore must satisfy $C\mathcal{R}(A, B) = \mathcal{R}(F, G)$. Therefore, $\mathcal{R}(A, B) \subseteq C^{-1}(\mathcal{R}(F, G)) = \mathcal{R}(A, B) + \text{Ker}(C)$. By strong consistency $\text{Ker}(C) \subseteq \mathcal{R}(A, B)$, and thus $C^{-1}(\mathcal{R}(F, G)) \subseteq \mathcal{R}(A, B)$. Therefore, $C^{-1}(\mathcal{R}(F, G)) = \mathcal{R}(A, B)$. □

We now have the main ingredients for propagating controllability from the coarser to the more complex model. The following theorem is conceptually similar to [8, Th. 2.2], even though this paper focuses on purely continuous and linear models.

Theorem 5.10 (Consistency and Implementability imply Controllability): Consider the linear systems

$$\begin{aligned} (\Sigma_1) \qquad & \dot{x} = Ax + Bu \\ (\Sigma_2) \qquad & \dot{y} = Fy + Gv \end{aligned}$$

which are C-related system with respect to the surjection $y = Cx$. Assume that Σ_1 implements Σ_2, and Σ_1 is consistent, that is $C\mathcal{R}(A, B) = \mathcal{R}(F, G)$. Then Σ_2 is controllable iff Σ_1 is macrocontrollable. If in addition Σ_1 is strongly consistent, that is $\mathcal{R}(A, B) = C^{-1}(\mathcal{R}(F, G))$, then Σ_1 is controllable iff Σ_2 is controllable.

Proof: Same as the proof of Propositions 4.8 and 4.11. □

Thus, in order to propagate controllability between two linear systems, we have to ensure that the systems are C-related and check either Condition (5.9) or (5.10) depending on the notion of controllability that is needed. It is desirable to have a methodology for constructing C related systems with the desirable properties. Fortunately, for the C-related system constructed in Proposition 5.1, (strong) consistency implies implementability. In order to show this, we will need the following lemma.

Lemma 5.11: Let $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times k}$, and full rank $C \in \mathbb{R}^{m \times n}$ be such that

$$A\text{Ker}(C) \subseteq \text{Ker}(C) + \mathcal{R}(A, B)$$

and let $F = CAC^+$. Then $C\mathcal{R}(A, B)$ is F-invariant, that is

$$FC\mathcal{R}(A, B) \subseteq C\mathcal{R}(A, B).$$

Proof: Let $y = Cx$ for $x \in \mathcal{R}(A, B)$ and consider

$$Fy = CAC^+ y = CAC^+ Cx$$

Decompose $x = x^c + x^n$ where $x^c \in \text{Ker}(C)$ and $x^n \in \text{Ker}(C)^\perp$. Then

$$Fy = CAC^+ C(x^c + x^n) = CAx^n = CA(x - x^c).$$

Since $x \in \mathcal{R}(A, B)$ and $\mathcal{R}(A, B)$ is A-invariant, we get that $CAx \in C\mathcal{R}(A, B)$. By consistency, there exist $z^c \in \text{Ker}(C)$ and $z^r \in \mathcal{R}(A, B)$ such that

$$CAx^c = C(z^c + z^r) = Cz^r. \tag{5.11}$$

Thus CAx^c also belongs in $C\mathcal{R}(A, B)$ resulting in $Fy \in C\mathcal{R}(A, B)$. □

Theorem 5.12 (Consistency Implies Implementability): Consider the linear system

$$(\Sigma_1) \qquad \dot{x} = Ax + Bu$$

which is consistent with respect to the surjective map $y = Cx$. Let

$$(\Sigma_2) \qquad \dot{y} = Fy + Gv$$

be the system where

$$\begin{aligned} F &= CAC^+ \\ G &= [CB \;\; CAv_1 \;\; \cdots \;\; CAv_r] \end{aligned}$$

with C^+ the pseudoinverse of C and $v_1, \cdots, v_r$ spanning $\text{Ker}(C)$. *Then Σ_2 is implementable by Σ_1.*

Proof: By Theorem 5.3 we have that $\mathcal{R}(F, G) \supseteq C\mathcal{R}(A, B)$ and thus we only need to show that $\mathcal{R}(F, G) \subseteq C\mathcal{R}(A, B)$. Let $y_f \in \mathcal{R}(F, G)$. Then

$$y_f = [G \; FG \; \cdots \; F^{m-1} G] x \tag{5.12}$$

for some $x \in \mathbb{R}^{ml}$. By an appropriate partition of $x = [x_1 \; x_2 \; \cdots \; x_m]^T$, we get

$$y_f = Gx_1 + FGx_2 + \cdots + F^{m-1} Gx_m \tag{5.13}$$

It suffices to show that $\mathcal{R}(G) \subseteq C\mathcal{R}(A, B)$ since then, by Lemma 5.11, we get that $\mathcal{R}(FG) \subseteq C\mathcal{R}(A, B), \cdots, \mathcal{R}(F^{m-1}G) \subseteq C\mathcal{R}(A, B)$. Now consider

$$\begin{aligned} y_1 &= Gx_1 = [CB\ CAv_1\ \cdots\ CAv_k]\begin{bmatrix} x_1^1 \\ x_1^2 \end{bmatrix} \\ &= CBx_1^1 + [CAv_1\ \cdots\ CAv_k]x_1^2. \end{aligned} \tag{5.14}$$

Clearly, $CBx_1^1 \in C\mathcal{R}(A, B)$. By consistency, we have

$$A\mathrm{Ker}(C) \subseteq \mathrm{Ker}(C) + \mathcal{R}(A, B) \tag{5.15}$$

and therefore for $i = 1, \cdots, k$

$$Av_i = v_i^c + v_i^r \tag{5.16}$$

for some $v_i^c \in \mathrm{Ker}(C)$ and $v_i^r \in \mathcal{R}(A, B)$. Thus

$$\begin{aligned} CAv_i &= C(v_i^c + v_i^r) = Cv_i^r \\ &= C[B\ AB\ \cdots\ A^{n-1}B]q_i \end{aligned} \tag{5.17}$$

for some vectors q_i of appropriate dimension. But then

$$\begin{aligned} &[CAv_1\ \cdots\ CAv_k]x_1^2 \\ &\quad = C[B\ AB\ \cdots\ A^{n-1}B][q_1\ \cdots\ q_k]x_1^2 \\ &\quad = C[B\ AB\ \cdots\ A^{n-1}B]X_1^2 \end{aligned} \tag{5.18}$$

and thus $\mathcal{R}(G) \in C\mathcal{R}(A, B)$. □

As a result of the above theorem, if we use Proposition 5.1 to construct our abstracted models, then consistency (or strong consistency) is the only condition on the aggregation map that is needed to propagate controllability.

Theorem 5.13 (Consistency Implies Controllability): Consider the linear system

$$(\Sigma_1) \qquad \dot{x} = Ax + Bu$$

and surjective map $y = Cx$. Let

$$(\Sigma_2) \qquad \dot{y} = Fy + Gv$$

be the C-related system where

$$F = CAC^+, \qquad G = [CB\ \ CAv_1\ \ \cdots\ \ CAv_r]$$

with C^+ the pseudoinverse of C and $v_1, \cdots, v_r$ spanning $\mathrm{Ker}(C)$. *If*

$$A\mathrm{Ker}(C) \subseteq \mathrm{Ker}(C) + \mathcal{R}(A, B)$$

then Σ_2 is macrocontrollable iff Σ_1 is controllable. In particular, if

$$\mathrm{Ker}(C) \subseteq \mathcal{R}(A, B)$$

then Σ_1 is controllable iff Σ_2 is controllable.

Proof: Follows from Theorems 5.10 and 5.12. □

It is interesting to notice what happens to Conditions (5.6) and (5.4) when the linear system is a linear vector field and thus $B = 0$. In that case, Condition (5.4) reduces to

$$A\mathrm{Ker}(C) \subseteq \mathrm{Ker}(C)$$

which, recall from Section II, is the necessary and sufficient condition to obtain a well-defined quotient vector field. Therefore a consistent abstraction of a linear vector field cannot have any control inputs (or cannot be a differential inclusion). Condition (5.6) reduces to

$$\mathrm{Ker}(C) = \{0\}$$

and thus $y = Cx$ must be an invertible linear transformation (since it is already surjective). We will be typically interested in consistent abstractions which are *nontrivial*, in the sense that some state space reduction is performed (thus $\mathrm{Ker}(C) \neq \{0\}$), but the abstracted system is also nontrivial ($\mathrm{Ker}(C) \neq \mathbb{R}^n$).

Corollary 5.14: Consider the assumptions of Theorem 5.13 and assume that $0 < rank(B) < n$. Then a nontrivial, strongly consistent abstraction always exists.

Proof: If $rank(B) > 0$, then we can always find a linear map C such that $\mathrm{Ker}(C) = \mathrm{Im}[B]$. □

Theorem 5.13 and Corollary 5.14 are important as they show that *a consistent abstraction always exists as long as there are control inputs*. In addition, the notions of consistency are important from a hierarchical perspective as they provide good design principles for constructing valid hierarchies. For example, the condition for strong consistency $\mathrm{Ker}(C) \subseteq \mathcal{R}(A, B)$ suggests that in order to ignore dynamics at a higher level [captured by $\mathrm{Ker}(C)$], one would have to ensure the ignored dynamics can be accommodated at the lower level.

As one imposes more restrictions on the matrix C, further properties can be propagated from one system to the other. The following results show conditions under which full trajectories can be implemented by the lower level system.

Theorem 5.15 (Trajectory Implementation): Consider two linear systems

$$\begin{aligned} (\Sigma_1) \qquad \dot{x} &= Ax + Bu \\ (\Sigma_2) \qquad \dot{y} &= Fy + Gv \end{aligned}$$

and the surjective map $y = Cx$. Assume $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$ with $m \leq n$, and $u \in \mathbb{R}^k$ with $k \leq n$. We assume B is of full rank. Let $\mathcal{K} = \mathrm{Ker}(C)$, $\mathcal{B} = \mathrm{Im}[B]$, $\mathcal{G} = \mathrm{Im}[G]$, and let P denote the orthogonal projection from $\mathbb{R}^m$ onto $CA\mathcal{K} + C\mathcal{B}$. We make the following two assumptions:

1) *$CAx = FCx$ for all $x \in \mathcal{K}^\perp$ (the orthogonal complement of $\mathcal{K}$).*
2) *$C^{-1}((I - P)\mathcal{G}) \subseteq \mathcal{B}$.*

Then, for every trajectory $y(\cdot)$ of Σ_2 corresponding to a differentiable control, there exists a trajectory $x(\cdot)$ of Σ_1, such that $y(t) = Cx(t)$ for all t in the domain of $y(\cdot)$.

Proof: Let $y(\cdot)$ be a trajectory of Σ_2 corresponding to the control v. First we define $x_a(t) = C^+y(t)$ where C^+ is the Moore–Penrose pseudo-inverse of C. If $z \in \mathcal{K}$, then

$$z^T x_a(t) = z^T C^T (CC^T)^{-1} y(t) = (Cz)^T (CC^T)^{-1} y(t) = 0.$$

Therefore, $x_a(t) \in \mathcal{K}^\perp$ for all t. Moreover, $\dot{x}_a(t) = C^+\dot{y}(t)$ where $\dot{y}(t) = Fy(t) + Gv(t)$.

Let P denote the orthogonal projection from $\mathbb{R}^m$ onto $CA\mathcal{K} + C\mathcal{B}$. Let D be the restriction of C on $A\mathcal{K} + \mathcal{B}$ and let D^+

be its pseudoinverse. Define $\overline{x}(t) = D^{+}P(Gv(t))$, and therefore, by construction we have that $C\overline{x}(t) = P(Gv(t))$ and $\overline{x}(t) \in A\mathcal{K} + \mathcal{B}$. Thus, there exist $x_b(t) \in \mathcal{K}$ and $b(t) \in \mathcal{B}$, such that $\overline{x}(t) = Ax_b(t) + b(t)$. Since $\overline{x}(t)$ is differentiable, we may choose $x_b(t)$ and $b(t)$ to be differentiable as well (using a suitable pseudoinverse). Let $x(t) = x_a(t) + x_b(t)$. Then $Cx(t) = C(x_a(t) + x_b(t)) = Cx_a(t) = y(t)$, and in addition

$$\begin{aligned} C\dot{x} &= C(\dot{x}_a + \dot{x}_b) = C\dot{x}_a = \dot{y} = Fy + Gv \\ &= FCx_a + Gv = CAx_a + Gv \end{aligned}$$

where the last equality holds by Assumption 1. Set $z(t) = \dot{x}(t) - Ax_a(t) - \overline{x}(t)$. Then for all t, $Cz(t) = C(\dot{x}_a(t) + \dot{x}_b(t)) - CAx_a(t) - C\overline{x}(t) = CAx_a(t) + Gv(t) - CAx_a(t) - P(Gv(t)) = (I - P)Gv(t)$. By Assumption 2, for each t there is $u(t) \in \mathbb{R}^k$, such that $z(t) = Bu(t)$. In fact, we can take $u(t) = B^{+}z(t)$ (here $B^{+} = (B^TB)^{-1}B^T$ since $k \leq n$). Then if we let $x(t) = x_a(t) + x_b(t)$ we get $\dot{x}(t) = Ax(t) + Bu(t)$ and $Cx(t) = Cx_a(t) = y(t)$ for all t. □

Corollary 5.16: Let Σ_1, Σ_2, and C be as in Proposition 5.1. If Ker$(C) \subseteq$ Im$[B]$, *then for every trajectory $y(\cdot)$ of Σ_2 corresponding to a differentiable control there exists a trajectory $x(\cdot)$ of Σ_1 such that $y(t) = Cx(t)$ for all t in the domain of $y(\cdot)$.*

Proof: Set $\mathcal{K} =$ Ker(C), $\mathcal{B} =$ Im$[B]$, and $\mathcal{G} =$ Im$[G]$. Since $C^{+}Cx = x$ for $x \in \mathcal{K}^{\perp}$, Assumption 1 of Theorem 5.15 is satisfied. Now $G = [CB \quad CAv_1 \quad \cdots \quad CAv_r]$, and since P is the orthogonal projection onto $CA\mathcal{K} + C\mathcal{B}$, we get $(I - P)\mathcal{G} = 0$. Then Assumption 2 of Theorem 5.15 reduces to $C^{-1}(0) =$ Ker$(C) \subseteq$ Im$[B]$, which is our assumption. □

VI. Hierarchical Controllability Algorithm

In this section, we will take advantage of the results of Section V in order to analyze the controllability of large scale linear systems. Theorem 5.13 enables us to have a hierarchical controllability criterion, which decomposes the controllability problem into a sequence of smaller problems. Such an approach is numerically more efficient and robust than the standard Kalman rank and PBH eigenvalue tests.

Conceptually, the algorithm starts with the linear system in question, and determines the number of linearly independent input vector fields. If this number is zero, then the system is uncontrollable and the algorithm terminates. If the number of linearly independent inputs is equal to the number of states, then the system is trivially controllable and the algorithm terminates as well. If the number of linearly independent vector fields is less than the number of states but greater than zero, then by Corollary 5.14 we can always find an aggregation matrix C, satisfying the strong consistency condition Ker$(C) \subseteq \mathcal{R}(A, B)$. Since Im$[B \;\; AB \;\; \cdots \;\; A^kB] \subseteq$ Im$[B \;\; AB \;\; \cdots \;\; A^{n-1}B]$ for any $0 \leq k \leq n-1$, from a computational standpoint, we can actually choose any matrix C satisfying Ker$(C) =$ Im$[B \; AB \; \cdots \; A^kB]$ for $0 \leq k \leq n-1$. If $k = 0$, then the abstracted system essentially ignores the directions spanned by the input vector fields (which are trivially controllable). If $k = n-1$, then the matrix C will ignore the whole reachable space.

After a consistent C matrix is determined, the construction of Theorem 5.13 is used in order to obtain a system of smaller dimension with equivalent controllability properties. We recursively apply the same procedure to this new abstracted system. Eventually, by dimension count, either there will be no inputs left and the system will be trivially uncontrollable, or there should be as many linearly independent inputs as number of states in which case controllability follows trivially. Since at each step, the abstractions that are constructed are consistent, then by Theorem 5.13, the outcome of the algorithm at the coarsest level will propagate along this sequence of consistent abstractions to the original complex model.

Algorithm 6.1 (Hierarchical Controllability Algorithm):

1. Given $\dot{x} = Ax + Bu$, $A \in \mathbb{R}^{n\times n}$, $0 \leq k \leq n-1$
2. If $rank(B)$ is
   - 0: System is uncontrollable. Stop
   - n: System is controllable. Stop
3. Find C such that Ker$(C) =$ Im$[B \; AB \; \cdots A^kB]$
4. Obtain new system matrices A, B of the abstracted system using Theorem 5.13
5. Return to 2

The larger k is, the fewer steps the algorithm will need to terminate. On the other hand, as k increases, the amount of computation per step will be higher. Before we discuss computational and implementation aspects of the above algorithm, we will demonstrate its application on various examples.

Example 6.2: Consider the linear system

$$\begin{aligned} \dot{x} &= \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} u \\ &= A_1x + B_1u. \end{aligned} \tag{6.1}$$

Since there is one linearly independent input field, we can find a consistent abstraction satisfying

$$\text{Ker}(C_1) = \text{Im}[B_1] \subseteq Im[B_1 \; A_1B_1 \; A_1^2B_1].$$

For example, we can choose

$$C_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The construction of Theorem 5.13 then results in

$$A_2 = C_1A_1C_1^{+} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \qquad B_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{6.2}$$

Since B_2 is nonzero and the number of linearly independent inputs is strictly less than the number of states, we can obtain another consistent abstraction by choosing $C_2 = [1 \quad 0]$. The resulting abstraction is

$$A_3 = C_2A_2C_2^{+} = 0, \qquad B_3 = 1. \tag{6.3}$$

At this point, the number of inputs is equal to the number of states and thus the pair (A_3, B_3) is trivially controllable. By

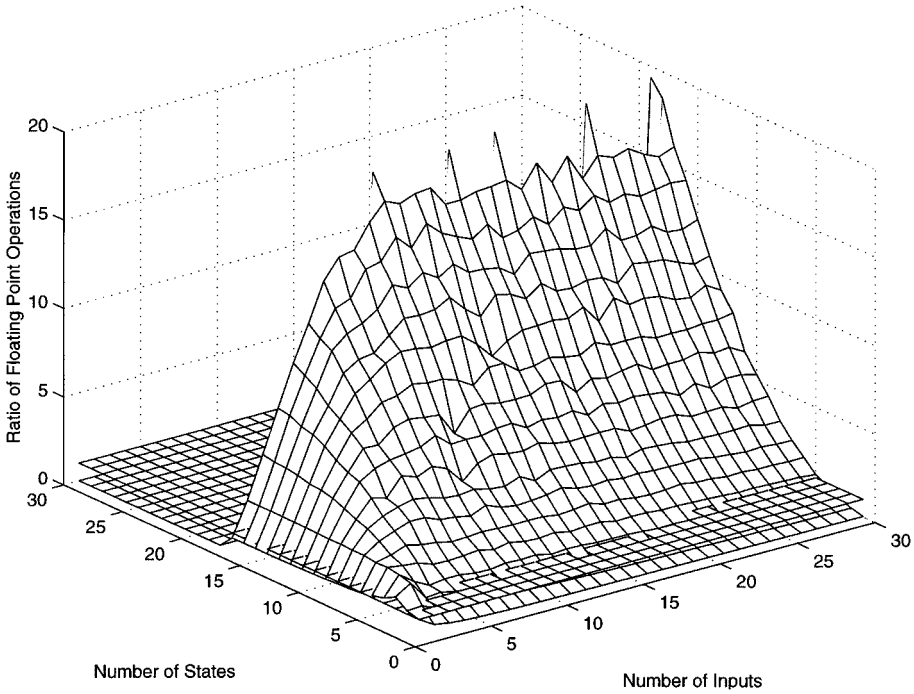

Fig. 3. Comparison of Algorithm 6.4 and the Kalman rank condition.

consistency, the pairs (A_2, B_2) and (A_1, B_1) are also controllable.

There is a much more intuitive explanation of the sequence of steps taken above. Note that the system started with the pair (A_1, B_1) and in the first iteration, we essentially removed the dynamics of x_2 (second row) from equation (6.1) since they have direct connection to the input u. This results in the pair (A_2, B_2), where x_2 can now be thought of as an input. We re-apply the above procedure by now removing the dynamics of x_3 [second row of (6.2)] since they can be directly controlled by the new controls. This results in the pair (A_3, B_3) which is trivially controllable.

Example 6.3: Consider the linear system

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u = A_1 x + B_1 u. \quad (6.4)$$

A consistent abstraction results by choosing the aggregation matrix

$$C_1 = [-1 \quad 1]$$

resulting in

$$A_2 = C_1 A_1 C_1^+ = 0, \qquad B_2 = 0. \quad (6.5)$$

Therefore, by Theorem 5.13, the pairs (A_2, B_2) and (A_1, B_1) are both uncontrollable.

In the case where we select $k = 0$ in Algorithm 6.1, then we choose matrices C satisfying $\mathrm{Ker}(C) = \mathrm{Im}[B]$. In this particular case $CB = 0$, and in addition, the columns of B span $\mathrm{Ker}(C)$. From a computational standpoint, it is advantageous to actually choose a matrix C, which not only satisfies $\mathrm{Ker}(C) = \mathrm{Im}[B]$, but is also a projection to $\mathrm{Im}[B]^{\perp}$. This reduces some of the computations of Theorem 5.13 and results in the following variation of Algorithm 6.1.

Algorithm 6.4 (Hierarchical Controllability Algorithm):

1. Given $\dot{x} = Ax + Bu$, $A \in \mathbb{R}^{n \times n}$.
2. If $rank(B)$ is
   - 0: System is uncontrollable. Stop
   - n: System is controllable. Stop
3. Find matrix C such that $\mathrm{Ker}(C) = \mathrm{Im}[B]$
4. Let $A := CAC^+$, $B := CAB$
5. Return to 2

Intuitively, Algorithm 6.4 starts with the system in question and, since $\mathrm{Im}[B]$ is in the controllable region, it chooses an abstraction matrix C which essentially projects the system in a direction which is orthogonal to the space spanned by B. Thus the macroinputs of the first abstraction are spanned by CAB, which are the first order Lie brackets of the original system, *projected on the orthogonal complement of* Im *[B]*. Similarly, the second abstraction will have as input vector fields the second-order Lie brackets projected on the orthogonal complement of both $\mathrm{Im}[B]$ and $\mathrm{Im}[AB]$.[3] Because of this selection of inputs at each abstraction layer, we simply have to add the dimension of the span of the input vector fields at each abstraction layer in order to obtain the dimension of the controllability subspace. From the above discussion, it is also clear that if the system is uncontrollable, then the algorithm computes the uncontrollable part of the system since at each iteration we are projecting on the space orthogonal to parts of the controllable space. The sequence of abstracting maps can then be used in a straightforward manner

[3]Clearly, macroinputs being projections of Lie brackets will be useful in developing a nonlinear version of this theory.

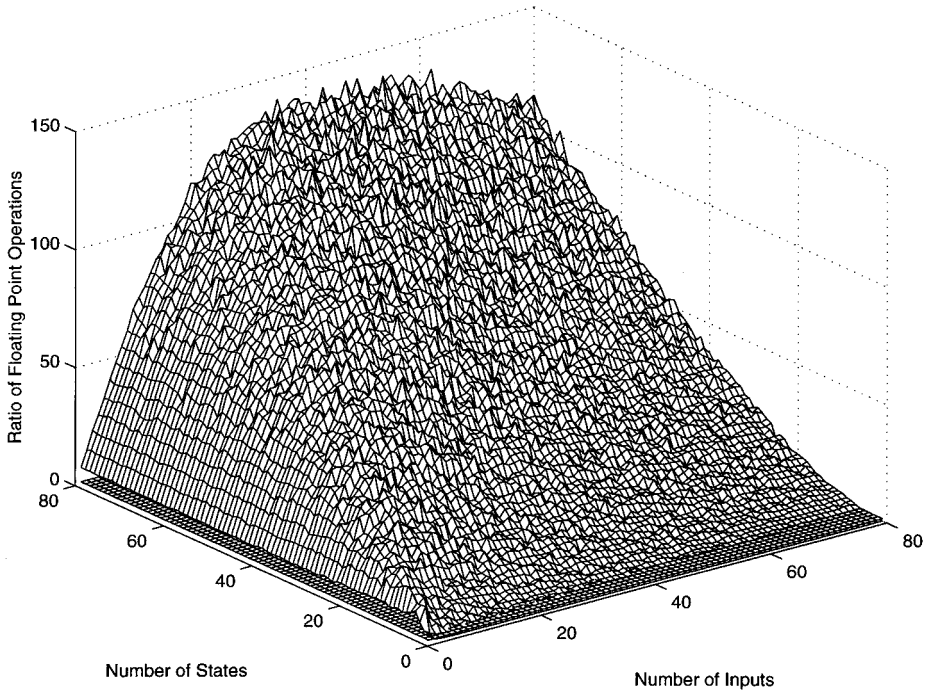

Fig. 4. Comparison of Algorithm 6.4 and the PBH test.

in order to decompose the system into controllable and uncontrollable subsystems.

We now focus on the implementation issues of Algorithms 6.1 and 6.4. For simplicity, we consider Algorithm 6.4; Algorithm 6.1 can be treated in a similar manner. From a computational perspective, the two main problems for implementing Algorithm 6.4 are: first, the construction of a consistent aggregation matrix C satisfying $\mathrm{Ker}(C) = \mathrm{Im}[B]$, and second, given such a matrix, to perform the computations required for the construction of a consistent abstraction. In order to tackle the first problem, we perform a singular value decomposition on the matrix B. The $n \times m$ $(n \geq m)$ matrix B with rank r is decomposed as

$$B = U\Sigma V^T = [U_1\ U_2] \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} = U_1 \Sigma_r V_1^T \quad (6.6)$$

where Σ_r is the $r \times r$ matrix of nonzero singular values. From the above decomposition we immediately obtain that $\mathrm{Ker}(C) = \mathrm{Im}[B] = \mathrm{Im}[U_1]$ and we can therefore choose the abstracting map $C = U_2^T$. In addition, $C^+ = U_2$, and therefore the singular value decomposition gives us, for free, the pseudoinverse calculation. Similar constructions are used in the implementation of Algorithm 6.1. Of course, singular value decompositions are computationally expensive. If speed of computation is of great interest, then QR-type decompositions could be used instead of singular value decompositions in order to accelerate the algorithm. However, as is typical in such cases, this may result in a less robust algorithm. The Matlab code that implements Algorithms 6.1 and 6.4 can be found in the Appendix.

Various experimental, comparative studies were performed on a Matlab platform. Given the dimension of the state and input space, random A, B matrices were generated, and their controllability was checked using the Kalman rank condition, the PBH test and Algorithm 6.4. Floating point operations were measured for each test, and the following ratios:

$$\text{Ratio} = \frac{\text{Floating Point Operations of Kalman or PBH Test}}{\text{Floating Point Operations of Algorithm 6.4}}$$

are plotted as a function on state and input dimension in Figs. 3 and 4. The plane with ratio equal to one is also plotted. Whenever the unreliable Kalman rank test fails to recognize a controllable system, the ratio is set to zero. Note from Fig. 3, that the Kalman rank test is more efficient for very low dimensional systems but Algorithm 6.4 is up to 15 times faster for most systems. In addition, the Kalman condition fails to be reliable for systems with more than approximately 15 states. Fig. 4 compares the PBH test with Algorithm 6.4. Even though the PBH test is more reliable than the Kalman rank condition, it is significantly slower than Algorithm 6.4 (up to 150 times for some systems). In addition, it is well known (see [26]) that the PBH test is very sensitive to parameter perturbations due to eigenvalue calculations.

The computational and conceptual advantages of Algorithm 6.4 are verified by the fact that Algorithm 6.4 is identical to the controllability algorithm of [11], derived from a purely numerical analysis perspective. In [11], the above algorithm is shown to be numerically stable and is a stabilized version of the realization algorithm of [32] (Matlab command CTRBF). Fig. 5 compares Algorithm 6.4 with the more general Algorithm 6.1 with $k = 1$. Fig. 5 clearly shows that it may be advantageous to use Algorithm 6.1 with $k = 1$ only in cases where the state dimension is much larger than the input dimension.

The hierarchical framework developed in this paper places a geometric and conceptual framework on the best known

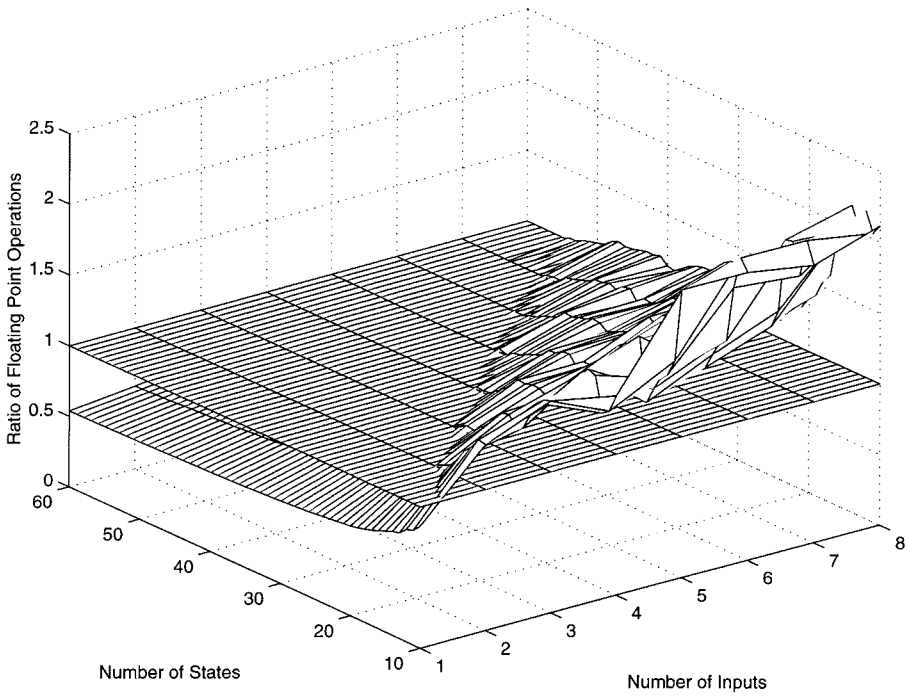

Fig. 5. Comparison of Algorithm 6.4 and Algorithm 6.1 with $k = 1$

controllability algorithm from numerical linear algebra. This is strong evidence that hierarchical decompositions of control problems are indeed reducing the complexity of control algorithms. It is therefore worthwhile pursuing this direction of research for more general classes of systems (nonlinear), as well as for other properties of interest (stabilizability, optimality, trajectory tracking).

VII. Conclusions: Issues for Further Research

In this paper, we considered a notion of control-system abstractions which are typically used in hierarchical and multi-layered systems. This was achieved by generalizing the notion of Φ-related vector fields to control systems. This notion is more general than the notion of projectable control systems [18], [22] and, in addition, mathematically formalizes the concept of virtual inputs used in backstepping designs [14]. The notions of implementability and consistency were then defined in order to propagate controllability from the abstracted system to the more detailed one. These notions were completely characterized for linear systems, and the easily checkable conditions allowed us to construct a hierarchical controllability algorithm for linear systems.

There are many directions for further future research. The results of Section V enable the development of an open loop backstepping methodology which, given a sequence of consistent abstractions would recursively generate the actual control input, by first generating a control input for the abstracted system, and then recursively refine it as one adds more modeling detail. Nonlinear analogs of the results of Section V, will provide a hierarchical controllability algorithm for nonlinear systems which may be more efficient and robust from a symbolic computation point of view. Many other properties are also of interest and will be investigated both for linear and nonlinear control systems. For example, obtaining consistent abstractions for nonlinear systems with respect to stabilizability would essentially classify all backsteppable systems. Other properties of interest include trajectory tracking, optimality and the proper propagation of state and input constraints. The framework presented in this paper provides a suitable platform for such studies.

Finally, another direction which is of great interest from a hybrid systems perspective, is to obtain consistent, discrete and hybrid abstractions of continuous systems. A very interesting problem, however, remains the construction of finite and consistent state space partitions, given a continuous control system. An algorithm for constructing finite reachability-preserving quotients of vector fields is proposed in [19], [20], and [39].

Appendix
MATLAB Implemetation of Algorithms 6.1 and 6.4

```
function [controllable]=HCA(A,B,k,tol)
%****************************
%  Controllability Algorithms 6.1 and 6.4
%
%  Required Inputs: System Matrices A, B,
    Integer 0 ≤ k ≤ n − 1
    (k = 0 is Algorithm 6.4)
%  Optional Inputs: Tolerance threshold
  tol (used for rank computation)
%****************************
n=size(A,1);
if nargin == 3
  tol = n*norm(A,1)*eps;
end r = rank(B,tol);
```

```
  %*** Dimension of input space
while ((n > r) & (r > 0)),
  %*** If inputs exist and are less than
  states
    l = min(k, n - 1);
  %*** Ignore Lie brackets higher than n-1
    W = B;
  %*** Compute [B AB ...A^kB]
    for j = 1 : l,
      W = [B A * W];
    end
    [U,S,V] = svd(W);
  %*** Obtain consistent matrix C
    m = rank(S,tol);
    U1 = U(:,1:m) ;
    U2 = U(:,(m+1):n) ;
    C = U2';
    B = C*A*U1;
  %***Obtain consistent abstraction
    A = C*A*C';
    n = size(A,1)
  %*** Dimension of abstracted system
    r = rank(B,tol);
  %*** Dimension of macroinputs
end
if (n==r) controllable=1;
elseif (r==0) controllable=0;
end
```

REFERENCES

[1] R. Abraham, J. Marsden, and T. Ratiu, *Manifolds, Tensor Analysis and Applications*, ser. Applied Mathematical Sciences. New York: Springer-Verlag, 1988.

[2] P. J. Antsaklis, J. A. Stiver, and M. Lemmon, "Hybrid system modeling and autonomous control systems," in *Hybrid Systems*. ser. Lecture Notes in Computer Science, R. L. Grossman, A. Nerode, A. P. Ravn, and H. Rischel, Eds. New York: Springer-Verlag, 1993, vol. 736, pp. 366–392.

[3] M. Aoki, "Control of large scale dynamic systems by aggregation," *IEEE Trans. Automat. Cont.*, vol. 13, pp. 246–253, June 1968.

[4] R. Brockett, "Control theory and analytical mechanics," in *Geometric Control Theory*, C. Martin and R. Hermann, Eds. Brookline, MA: Math. Sci. Press, 1977, pp. 1–46.

[5] ——, "Global descriptions of nonlinear control problems; Vector bundles and nonlinear control theory,", manuscript, 1980.

[6] P. Caines and Y. J. Wei, "The hierarchical lattices of a finite state machine," *Syst. Contr. Lett.*, vol. 25, pp. 257–263, 1995.

[7] ——, "Hierarchical hybrid control systems," in *Control Using Logic Based Switching*. ser. Lecture Notes in Control and Information Sciences, S. Morse, Ed. New York: Springer-Verlag, 1996, vol. 222, pp. 39–48.

[8] ——, "Hierarchical hybrid control systems: A lattice theoretic formulation," *IEEE Trans. Automat. Cont.*, vol. 43, pp. 501–508, Apr. 1998.

[9] P. Cousot and R. Cousot, "Systematic design of program analysis framework," in *Proc. 6th ACM Symp. Principles of Programming Languages*, 1979.

[10] J. E. R. Cury, B. H. Krogh, and T. Niinomi, "Synthesis of supervisory controllers for hybrid systems based on approximating automata," *IEEE Trans. Automat. Contr.*, vol. 43, pp. 564–568, Apr. 1998.

[11] P. M. Van Dooren, "The generalized eigenstructure problem in linear system theory," *IEEE Trans. Automat. Contr.*, vol. AC-26, pp. 111–129, Jan. 1981.

[12] G. H. Golub and C. F. van Loan, *Matrix Computations*, third ed. Baltimore, MD: The John Hopkins University Press, 1996.

[13] T. A. Henzinger and H. Wong-Toi, "Linear phase-portrait approximations for nonlinear hybrid systems," in *Hybrid Systems III*, R. Alur, T. A. Henzinger, and E. D. Sontag, Eds. New York: Springer-Verlag, 1996, pp. 377–388.

[14] M. Kristic, I. Kanellakopoulos, and P. Kokotovic, "Adaptive and learning systems for signal processing, communications and control," in *Nonlinear and Adaptive Control Design*. New York: Wiley, 1995.

[15] C. P. Kwong, "Optimal chained aggregation for reduced order modeling," *Int. J. Contr.*, vol. 35, no. 6, pp. 965–982, 1982.

[16] ——, "Disaggregation, approximate disaggregation, and design of suboptimal control," *Int. J. Contr.*, vol. 37, no. 4, pp. 843–854, 1983.

[17] C. P. Kwong and C. F. Chen, "A quotient space analysis of aggregated models," *IEEE Trans. Automat. Contr.*, vol. AC-27, pp. 203–205, Feb. 1982.

[18] C. P. Kwong and Y. K. Zheng, "Aggregation on manifolds," *Int. J. Syst. Sci.*, vol. 17, no. 4, pp. 581–589, 1986.

[19] G. Lafferriere, G. J. Pappas, and S. Sastry, "Hybrid systems with finite bisimulations," in *Hybrid Systems V*. ser. Lecture Notes in Computer Science, P. Antsaklis, W. Kohn, M. Lemmon, A. Nerode, and S. Sastry, Eds. New York: Springer-Verlag, 1998, vol. 1567, pp. 186–203.

[20] ——, "Subanalytic stratifications and bisimulations," in *Hybrid Systems: Computation and Control*. ser. Lecture Notes in Computer Science, T. Henzinger and S. Sastry, Eds. New York: Springer-Verlag, 1998, vol. 1386, pp. 205–220.

[21] C. Loiseaux, S. Graf, J. Sifakis, A. Bouajjani, and S. Bensalem, "Property preserving abstractions for the verification of concurrent systems," in *Formal Methods in Systems Design*. Norwell, MA: Kluwer, 1995, vol. 6, pp. 1–35.

[22] L. S. Martin and P. E. Crouch, "Controllability on principal fiber bundles with compact structure group," *Syst. Contr. Lett.*, vol. 5, no. 1, pp. 35–40, 1984.

[23] M. D. Mesarovic, "Theory of hierarchical, multilevel, systems," in *Mathematics in Science and Engineering*. New York: Academic, 1970, vol. 68.

[24] J. R. Munkres, *Analysis on Manifolds*. Reading, MA: Addison-Wesley, 1991.

[25] H. Nijmeijer and A. J. van der Schaft, *Nonlinear Dynamical Control Systems*. New York: Springer-Verlag, 1990.

[26] C. C. Paige, "Properties of numerical algorithms related to computing controllability," *IEEE Trans. Automat. Contr.*, vol. AC-26, pp. 111–129, Jan. 1981.

[27] G. J. Pappas and S. Sastry, "Toward continuous abstractions of dynamical and control systems," in *Hybrid Systems IV*. ser. Lecture Notes in Computer Science, P. Antsaklis, W. Kohn, A. Nerode, and S. Sastry, Eds. New York: Springer-Verlag, 1997, vol. 1273, pp. 329–341.

[28] G. J. Pappas, C. Tomlin, J. Lygeros, D. N. Godbole, and S. Sastry, "A next generation architecture for air traffic management systems," in *Proc. 36th IEEE Conf. Decision and Control*, San Diego, CA, Dec. 1997, pp. 2405–2410.

[29] G. J. Pappas, G. Lafferriere, and S. Sastry, "Hierarchically consistent control systems," in *Proc. 37th IEEE Conf. Decision and Control*, Tampa, FL, December 1998, pp. 4336–4341.

[30] A. Puri and P. Varaiya, "Decidability of hybrid systems with rectangular differential inclusions," *Computer Aided Verification*, pp. 95–104, 1994.

[31] J. Raisch and S. D. O'Young, "Discrete approximations and supervisory control of continuous systems," *IEEE Trans. Automat. Contr.*, vol. 43, pp. 569–573, Apr. 1998.

[32] H. H. Rosenbrock, *State Space and Multivariable Theory*. New York: Wiley, 1970.

[33] M. Spivak, *A Comprehensive Introduction to Differential Geometry*: Publish or Perish, 1979.

[34] P. Varaiya, "Smart cars on smart roads: Problems of control," *IEEE Trans. Automat. Contr.*, vol. AC-38, pp. 195–207, Feb. 1993.

[35] K. C. Wong and W. M. Wonham, "Hierarchical control of discrete-event systems," *Discrete Event Dynam. Syst.*, vol. 6, pp. 241–273, 1995.

[36] ——, "Hierarchical control of timed discrete-event systems," *Discrete Event Dynam. Syst.*, vol. 6, pp. 275–306, 1995.

[37] W. M. Wonham, "Linear multivariable control: A geometric approach," in *Applications of Mathematics*. New York: Springer-Verlag, 1985, vol. 10.

[38] H. Zhong and W. M. Wonham, "On the consistency of hierarchical supervision in discrete-event systems," *IEEE Trans. Automat. Contr.*, vol. 35, pp. 1125–1134, Oct. 1990.

[39] G. Lafferrier, G. J. Pappas, and S. Sastry, "O-minimal hybrid systems," *Math. Contr. Signals Syst.*, vol. 13, pp. 1–21, Mar. 2000.

George J. Pappas (S'91—M'98) received the B.S. degree in computer and systems engineering in 1991 and the M.S. degree in computer and systems engineering in 1992, both from Rensselaer Polytechnic Institute, Troy, NY. In December 1998, he received the Ph.D. degree from the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley.

He is currently an Assistant Professor in the Department of Electrical Engineering, University of Pennsylvania, where he also holds a secondary appointment in the Department of Computer and Information Sciences. Previously he was a Postdoctoral Researcher with the University of California at Berkeley and the University of Pennsylvania. In 1994, he was a Graduate Fellow at the Division of Engineering Science, Harvard University, Cambridge, MA. His research interests include hybrid systems, hierarchical control systems, nonlinear control systems, geometric control theory with applications to flight management and air traffic management systems, robotics, and unmanned aerial vehicles.

Dr. Pappas is the recipient of the 1999 Eliahu Jury Award for Excellence in Systems Research from the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley. He was also a finalist for the Best Student Paper Award at the 1998 IEEE Conference on Decision and Control.

Gerardo Lafferriere received the *Licenciado* degree in mathematics from the University of La Plata, Argentina, in 1977, and the Ph.D. degree in mathematics from Rutgers University, New Brunswick, NJ, in 1986.

From 1986 to 1990, he worked as a Research Scientist at the Robotics Laboratory, the Courant Institute, NY. Since 1990, he has been with the Department of Mathematical Sciences, Portland State University, Portland, OR, where he is currently an Associate Professor. He spent the 1997–1998 academic year as a Visiting Associate Research Engineer at the Electronics Research Laboratory, University of California at Berkeley. His research interests are in nonlinear control theory, hybrid systems, and robotics.

S. Shankar Sastry (F'98) received the Ph.D. degree in 1981 from the University of California at Berkeley.

He was on the faculty of the Massachusetts Institute of Technology (MIT) from 1980 to 1982 and Harvard University, Cambridge, MA, as a Gordon McKay Professor in 1994. He is currently a Professor of Electrical Engineering and Computer Sciences and Director of the Electronics Research Laboratory at the University of California at Berkeley. He has held visiting appointments at the Australian National University, Canberra, the University of Rome, Scuola Normale, the University of Pisa, the CNRS Laboratory LAAS in Toulouse (poste rouge), and as a Vinton Hayes Visiting Fellow at the Center for Intelligent Control Systems, MIT. His areas of research are nonlinear and adaptive control, robotic telesurgery, control of hybrid systems, and biological motor control. He is a co-author (with M. Bodson) of *Adaptive Control: Stability, Convergence and Robustness* (Englewood Cliffs, NJ: Prentice-Hall, 1989) and co-author (with R. Murray and Z. Li) of *A Mathematical Introduction to Robotic Manipulation* (Boca Raton, FL: CRC Press, 1994), and the author of *Nonlinear Control: Analysis, Stability and Control* (New York: Springer-Verlag, 1999). He has co-edited (with P. Antsaklis, A. Nerode, and W. Kohn) *Hybrid Control II*, *Hybrid Control IV*, and *Hybrid Control V* (Springer Lecture Notes in Computer Science, 1995, 1997, and 1999), and co-edited (with Henzinger) *Hybrid Systems Computation and Control* (Springer-Verlag Lecture Notes in Computer Science, 1998) and (with Baillieul and Sussmann) *Essays in Mathematical Robotics* (Springer Verlag IMA Series).

Dr. Sastry was an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, *IEEE Control Magazine*, and the *Journal of Mathematical Systems, Estimation and Control*, and is currently an Associate Editor of the *IMA Journal of Control and Information*, the *International Journal of Adaptive Control and Signal Processing*, and the *Journal of Biomimetic Systems and Materials*. He received the President of India Gold Medal in 1977, the IBM Faculty Development Award for 1983–1985, the National Science Foundation Presidential Young Investigator Award in 1985, and the Eckman Award of the of the American Automatic Control Council in 1990.