Towards Continuous Abstractions of Dynamical and Control Systems*

George J. Pappas and Shankar Sastry
Department of Electrical Engineering and Computer Sciences
University of California at Berkeley
Berkeley, CA 94720
{gpappas,sastry}@eecs.berkeley.edu

Abstract. The analysis and design of large scale systems is usually an extremely complex process. In order to reduce the complexity of the analysis, simplified models of the original system, called *abstractions*, which capture the system behavior of interest are obtained and analyzed. If the abstraction of system can be shown to satisfy certain properties of interest then so does the original complex plant. In hybrid systems, discrete or hybrid abstractions of continuous systems are of great interest. In this paper, the notion of abstractions of continuous systems is formalized.

1 Introduction

Large scale systems such as intelligent highway systems [1] and air traffic control systems [2] result in systems of very high complexity. The design process for large scale systems consists of imposing an overall system architecture as well as designing communication and control algorithms for achieving a desired overall system performance. The merging of discrete communication protocols and continuous control laws results in *hybrid systems* ([3, 4, 5]). The analysis process for complex, hybrid systems consists of proving or verifying that the designed system indeed meets certain specifications. However, both the design and the analysis may be formidable due to the complexity and magnitude of the system.

In the design process, complexity is reduced by imposing a hierarchical structure ([6]) on the system architecture. By imposing this structure, systems of higher functionality reside at higher levels of the hierarchy and are therefore unaware of unecessary details regarding the lower levels of the hierarchy. This structure clearly reduces complexity since every level of the hierarchy is required to know only the necessary infomation in order to successfully complete its function. This may also increase efficiency and performance of the overall system since each level of the hierarchy can focus on its specific function which will be of minimal complexity.

From an analysis perspective, complexity reduction is performed by focusing on the dynamics of interest. For example, aircraft are usually modeled by

* Research supported by the Army Research Office under grants DAAH 04-95-1-0588 and DAAH 04-96-1-0341.

detailed differential equations which describe the behavior of engine dynamics, aerodynamics etc. A desired specification may be that any two aircraft reach their destinations and do not collide with each other. Proving that the system indeed meets the specification may be prohibitingly complex due to the detailed modeled dynamics as well as the large scale of the system.

However, in the above example, it is clear that the specification is not interested in the details of aircraft operation, but only in the relative position of the aircraft. We can therefore reduce the complexity of the analysis by by ignoring certain aspects of system behavior in a manner which is consistent with the behavior of the original system. This is essentially the idea behind system abstraction. Once a system abstraction has been obtained, standard analysis methods are utilized on the abstracted models. For example, verification algorithms of hybrid systems are based on abstracting continuous dynamics by rectangular differential inclusions [7, 8].

Webster's dictionary defines the word abstraction as "*the act or process of separating the inherent qualities or properties of something from the actual physical object or concept to which they belong*". In system theory, the objects are usually dynamical or control systems, the properties are usually the behaviors of certain variables of interest and the act of separation is essentially the act of capturing all interesting behaviors. In summary, Webster's definition can be applied to define the *abstraction of a system to be another system which captures all system behavior of interest.* Behaviors of interest are captured by an *abstracting map* denoted by α. Abstracting maps are provided by the user depending on what information is of interest.

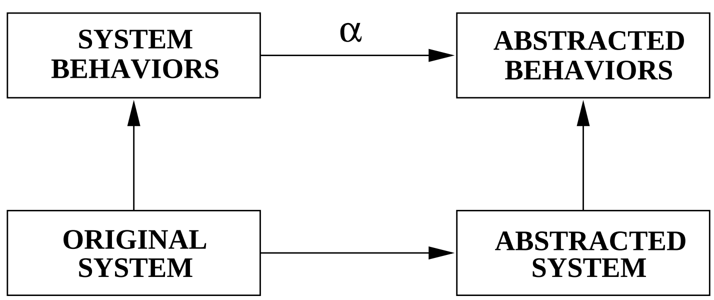

Fig. 1. System Abstractions

Figure 1 displays graphically the broad definition of a system abstraction. The original system may be modeled by ordinary or partial differential equations, discrete event systems or hybrid systems. The original system generates certain behaviors which are simply the system trajectories. The abstracting map α then selects the system behaviors which are of interest. The abstracted system must then be able to reproduce the same set of abstracted behaviors. The abstracted system must reproduce the system behaviors either *exactly* or *approximately*, resulting in exact or approximate abstractions respectively. Classical model reduction techniques can thus be thought of as approximate abstractions in this framework. The abstracted and original system do not have to be similar from a modeling point of view. For example, the original model may be an ordinary differential equation but the abstracted system may be a discrete event system. The main problem then is the following: *Given an original system and an abstracting map, find an abstracted system which generates the abstracted behaviors either exactly or approximately.*

In this paper, we consider the above problem for exact abstractions of dynamical and control systems. The notion of system abstraction is formalized and we consider the problem of abstracting continuous systems (differential equations and inclusions) by continuous systems. Necessary and sufficient geometric conditions under which one system is an exact abstraction of another with respect to a given abstracting map are derived. Although abstractions of systems may capture all behaviors of interest, they might also allow evolutions which are not feasible by the original system. This results in *overapproximating* the abstracted behaviors. This is due to the information reduction which naturally occurs during the abstraction process and it is the price one has to pay in order to reduce complexity. System abstractions can therefore be ordered based on the "size" of redundant allowable system evolutions leading to a notion of *best abstraction*. Furthermore, we show that certain properties of interest, such as controllability, propagate from the original system to the abstracted system.

The structure of this paper is as follows: In Section 2 we review some facts from differential geometry which will be used throughout the paper. In Section 3 abstracting maps are introduced in order to define system behaviors of interest. A notion of an abstraction of a dynamical system is defined in Section 4 and we discuss when one vector field is an abstraction of another. Section 5 generalizes these notions for control systems. Finally, Section 6 discusses issues of further research.

2 Mathematical Background

We first review some basic facts from differential geometry. The reader may wish to consult numerous books on the subject such as [9, 10, 11].

Let M be a differentiable manifold. The set of all tangent vectors at $p \in M$ is called the tangent space of M at p and is denoted by T_pM. The collection of all tangent spaces of the manifold,

$$TM = \bigcup_{p \in M} T_pM$$

is called the tangent bundle. The tangent bundle has a naturally associated projection map $\pi : TM \longrightarrow M$ taking a tangent vector $X_p \in T_pM \subset TM$ to the point $p \in M$. The tangent space T_pM can then be thought of as $\pi^{-1}(p)$.

The tangent space can be thought of as a special case of a more general mathematical object called a fiber bundle. Loosely speaking, a fiber bundle can be thought of as gluing sets at each point of the manifold in a smooth way.

Definition 1 (Fiber Bundles [12]). 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 $\pi : B \longrightarrow M$ 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 $\Phi_i : \pi^{-1}(O_i) \longrightarrow O_i \times U$ satisfying

$$\pi_o \circ \Phi = \pi$$

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*.

Now let M and N be smooth manifolds and $f : M \longrightarrow N$ be a smooth map. Let $p \in M$ and let $q = f(p) \in N$. We push forward tangent vectors from T_pM to T_qN using the induced push forward map $f_* : T_pM \longrightarrow T_qN$. If $f : M \longrightarrow N$ and $g : N \longrightarrow K$ then

$$(g \circ f)_* = g_* \circ f_*$$

which is essentially the chain rule. A vector field or dynamical system on a manifold M is a continuous map F which places at each point p of M a tangent vector from T_pM. Let $I \subseteq \Re$ be an open interval containing the origin. An integral curve of a vector field is a 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 for all $t \in I$,

$$c' = c_*(1) = X(c)$$

Finally, we have the following definition.

Definition 2 (f-related Vector Fields). Let X and Y be vector fields on manifolds M and N respectively and $f : M \longrightarrow N$ be a smooth map. Then X and Y are *f-related* iff $f_* \circ X = Y \circ f$.

3 Abstracting Maps

Let M be the state space of a system. In abstracting system dynamics, information about the state of the system which is not useful in the analysis process is usually ignored in order to produce a simplified model of reduced complexity. The state $p \in M$ is thus mapped to an abstracted state $q \in N$. It is clear that *complexity reduction requires that the dimension of N should be lower than the dimension of M.*

For example, each state could be mapped to part of the state or to certain outputs of interest. What state information is relevant usually depends on the properties which need to be satisfied. The desired specification, however, could be quite different even for the same system since the functionality of the system may be different in various modes of system operation. It is therefore clear that it is very difficult to intrinsically obtain a system abstraction without any knowledge of the particular system functionality. System functionality determines what state information is of interest for analysis purposes. Given the functionality of the system, a notion of equivalent states is obtained by defining an equivalence relation on the state space. For example, given a dynamic model of some mechanical system one may be interested only in the configuration of the system. In this case, two states are equivalent if the corresponding configurations are the same.

Once a specific equivalence has been chosen, then the quotient space $M/\sim$ is the state space of the abstracted system. In order for the quotient space to have a manifold structure, the equivalence relation must be regular [10]. The surjective map $\alpha : M \longrightarrow M/\sim$ which sends each state $p \in M$ to its equivalence class $[p] \in M/\sim$ is called the quotient map and is the map which sends each state to its abstracted state. In general, we have the following definition.

Definition 3 (Abstracting Maps). Let M and N be given manifolds with $dim(N) \leq dim(M)$. A surjective map $\alpha : M \longrightarrow N$ from the state space M to the abstracted state space N is called an *abstracting map*.

The quotient map is an example of an abstracting map. Other typical abstracting maps could be projection maps, output maps as well as the identity map. Since in this paper we are interested with continuous systems, we will assume that M and N are manifolds and the abstracting maps to be smooth submersions.

4 Abstractions of Dynamical Systems

Once an abstracting map α has been given, then given a vector field X which governs the state evolution on M, then one is interested in obtaining the evolution of the abstracted dynamics. The evolution of a dynamical system is characterized by its integral curves. Let c be any integral curve of X. Then if we push forward the curve c by the abstracting map α we obtain that $\alpha \circ c$ describes the evolution of the abstracted dynamics on N. If we therefore want to abstract the vector field X on M by a vector field Y on N, then $\alpha \circ c$ should be an integral curve of Y. This motivates the following definition.

Definition 4 (Abstractions of Dynamical Systems). Let X and Y be vector fields on M and N respectively and let $\alpha : M \longrightarrow N$ be a smooth abstracting map. Then vector field Y is *an abstraction of vector field X with respect to α* iff for every integral curve c of X, $\alpha \circ c$ is an integral curve of Y.

Therefore if the curve c satisfies

$$c' = c_*(1) = X(c)$$

then it must also be true that

$$(\alpha \circ c)' = (\alpha \circ c)_*(1) = Y(\alpha \circ c)$$

From Definition 4 it is clear that a vector field Y may be an abstraction of some vector field X for some abstracting map α_1, but may not be for another abstracting map α_2.

In building hierarchical models of large scale systems, the system may be modeled at many levels of abstraction. The following proposition shows that abstracting dynamical systems is transitive.

Proposition 5 (Transitivity of Abstractions). *Let X_1, X_2, X_3 be vector fields on manifolds M_1, M_2 and M_3 respectively. If X_2 is an abstraction of X_1 with respect to the abstracting map $\alpha_1 : M_1 \to M_2$ and X_3 is an abstraction of X_2 with respect to abstracting map $\alpha_2 : M_2 \to M_3$ then X_3 is an abstraction of X_1 with respect to abstracting map $\alpha_2 \circ \alpha_1$.*

Proof: Let c be any integral curve of X_1. Since X_2 is an abstraction of X_1, then by definition $\alpha_1(c)$ is an integral curve of X_2. But since X_3 is an abstraction of X_2, $\alpha_2(\alpha_1(c)) = (\alpha_2 \circ \alpha_1)(c)$ is an integral curve of X_3. Thus for any integral curve c of X_1, $(\alpha_2 \circ \alpha_1)(c)$ is an integral curve of X_3. Thus X_3 is an abstraction of X_1 with respect to abstracting map $\alpha_2 \circ \alpha_1$. □

The following theorem shows that Definition 4 is equivalent to saying that the two vector fields are α-related.

Theorem 6. *Vector field Y on N is an abstraction of vector field X on M with respect to the map α if and only if X and Y are α-related.*

Proof: Let vector field Y on N be an abstraction with respect to α of vector field X on M. Then by Definition 4, for any integral curve c of X, $\alpha \circ c$ is an integral curve of Y. Thus

$$\begin{aligned}
&(\alpha \circ c)' = (\alpha \circ c)_*(1) = Y(\alpha \circ c) \Rightarrow \\
&\alpha_* \circ c_*(1) = Y \circ \alpha \circ c \Rightarrow \\
&\alpha_* \circ X(c) = Y \circ \alpha \circ c \Rightarrow \\
&\alpha_* \circ X \circ c = Y \circ \alpha \circ c \Rightarrow \\
&\alpha_* \circ X = Y \circ \alpha
\end{aligned}$$

But then, by Definition 2, X and Y are α-related. Conversely, let X and Y be α related. Then for any integral curve c of X,

$$\begin{aligned}
&\alpha_* \circ X = Y \circ \alpha \Rightarrow \\
&\alpha_* \circ X \circ c = Y \circ \alpha \circ c \Rightarrow \\
&\alpha_* \circ X(c) = Y(\alpha \circ c) \Rightarrow \\
&\alpha_* \circ c_*(1) = Y(\alpha \circ c) \Rightarrow \\
&(\alpha \circ c)_*(1) = Y(\alpha \circ c)
\end{aligned}$$

and thus $\alpha \circ c$ is an integral curve of Y. Therefore Y is an abstraction of vector field X with respect to α. □

Theorem 6 is important because it allows to check a condition on the vector fields rather than explicitly computing integral curves and verifying Definition 4. However, α-relatedness of two vector fields is a very restrictive condition which limits the cases where one dynamical system is an exact abstraction of another. This forces us to examine approximate abstractions of dynamical systems in our future work.

5 Abstractions of Control Systems

The notions of Section 4 for dynamical systems will be extended to control systems. Control systems can also be thought of as differential inclusions depending on whether one has a design or analysis perspective. We first need to introduce some facts about control systems.

Definition 7 (Control Systems [13]). A *control system* $S = (B, F)$ consists of a fiber bundle $\pi : B \longrightarrow M$ called the control bundle and a smooth map $F : B \longrightarrow TM$ which is fiber preserving and hence satisfies

$$\pi' \circ F = \pi$$

where $\pi' : TM \longrightarrow M$ is the tangent bundle projection.

Essentially, the base manifold M of the control bundle is the state space and the fibers $\pi^{-1}(p)$ are the state dependent control spaces. In a local coordinate chart (V, x), the map F can be expressed as $\dot{x} = F(x, u)$ with $u \in U(x) = \pi^{-1}(x)$.

Definition 8 (Integral Curves of Control Systems). A curve $c : I \longrightarrow M$ is called *an integral curve of the control system* $S = (B, F)$ if there exists a curve $c^B : I \longrightarrow B$ satisfying

$$\begin{aligned} \pi \circ c^B &= c \\ c' = c_*(1) &= F(c^B) \end{aligned}$$

Again in local coordinates, the above definition simply says that $x(t)$ is a solution to a control system if there exists an input $u \in U(x) = \pi^{-1}(x)$ satisfying $\dot{x} = F(x, u)$. We now define abstractions of control systems in a manner similar to dynamical systems.

Definition 9 (Abstractions of Control Systems). Let $S_M = (B_M, F_M)$ with $\pi_M : B_M \longrightarrow M$ and $S_N = (B_N, F_N)$ with $\pi_N : B_N \longrightarrow N$ be two control systems. Let $\alpha : M \longrightarrow N$ be an abstracting map. Then control system S_N is *an abstraction of* S_M *with respect to abstracting map* α iff for every integral curve c_M of S_M, $\alpha \circ c_M$ is an integral curve of S_N.

From Definition 9 it is clear that a control system S_N may be an abstraction of S_M for some abstracting map α_1 but may not be for another abstracting map α_2. It can be easily shown that Definition 9 is transitive. Since the definition of an abstraction is at the level of integral curves, it is clearly difficult to conclude that one control system is an abstraction of another system by directly using Definition 9 since this would require integration of the system. One is therefore interested in easily checkable conditions under which one system is an abstraction of another. The following theorem, provides necessary and sufficient geometric conditions under which one control system is an abstraction of another system with respect to some abstracting map.

Theorem 10 (Conditions for Control System Abstractions). *Let $S_N = (B_N, F_N)$ and $S_M = (B_M, F_M)$ be two control systems and $\alpha : M \longrightarrow N$ be an abstracting map. Then S_N is an abstraction of S_M with respect to abstracting map α if and only if*

$$\alpha_* \circ F_M \circ \pi_M^{-1}(p) \subseteq F_N \circ \pi_N^{-1} \circ \alpha(p) \tag{1}$$

at every $p \in M$.

Proof: Before we proceed with the proof, we remark that condition (1) can be visualized using the following diagram,

$$\begin{array}{ccc} M & \xrightarrow{\alpha} & N \\ \pi_M^{-1}\downarrow & & \downarrow\pi_N^{-1} \\ B_M & & B_N \\ F_M\downarrow & & \downarrow F_N \\ TM & \xrightarrow{\alpha_*} & TN \end{array} \tag{2}$$

Then condition (1) states that in the above diagram the set of tangent vectors produced in the direction ($M \stackrel{\pi_M^{-1}}{\to} B_M \stackrel{F_M}{\to} TM \stackrel{\alpha_*}{\to} TN$) is a subset of the tangent vectors produced in the direction ($M \stackrel{\alpha}{\to} N \stackrel{\pi_N^{-1}}{\to} B_N \stackrel{F_N}{\to} TN$).

We begin the proof, by first showing that if $\alpha_* \circ F_M \circ \pi_M^{-1} \subseteq F_N \circ \pi_N^{-1} \circ \alpha$ at every point $p \in M$ then F_N is an abstraction of F_M. We will prove the contrapositive. Assume that F_N is not an abstraction of F_M. Then there exists an integral curve c_M of F_M such that $\alpha \circ c_M$ is not an integral curve of F_N. Therefore for all curves $c_N^B : I \longrightarrow B_N$ such that $\pi_N \circ c_N^B = \alpha \circ c_M$ we have that at some point $t^* \in I$

$$(\alpha \circ c_M)'(t^*) \neq F_N(c_N^B(t^*))$$

But since this is true for all curves c_N^B satisfying $\pi_N \circ c_N^B(t^*) = \alpha \circ c_M(t^*)$ and since π_N is a surjection we have

$$\begin{aligned} &(\alpha \circ c_M)'(t^*) \notin F_N(\pi_N^{-1}(\alpha \circ c_M(t^*))) \Rightarrow \\ &(\alpha \circ c_M)'(t^*) \notin F_N \circ \pi_N^{-1} \circ \alpha \circ c_M(t^*) \Rightarrow \\ &\alpha_* \circ c_{M*}(t^*)(1) \notin F_N \circ \pi_N^{-1} \circ \alpha \circ c_M(t^*) \Rightarrow \\ &\alpha_* \circ F_M \circ c_M^B(t^*) \notin F_N \circ \pi_N^{-1} \circ \alpha \circ c_M(t^*) \end{aligned} \tag{3}$$

for some curve $c_M^B : I \longrightarrow B_M$ such that $\pi_M \circ c_M^B = c_M$. But then $c_M^B(t^*) \in \pi_M^{-1}(c_M(t^*)) = \pi_M^{-1} \circ c_M(t^*)$. Therefore, there exists a tangent vector $Y_{\alpha(c_M(t^*))} \in T_{\alpha(c_M(t^*))}N$, namely

$$Y_{\alpha(c_M(t^*))} = \alpha_* \circ F_M \circ c_M^B(t^*)$$

such that

$$Y_{\alpha(c_M(t^*))} \in \alpha_* \circ F_M \circ \pi_M^{-1} \circ c_M(t^*)$$

since $c_M^B(t^*) \in \pi_M^{-1} \circ c_M(t^*)$ but

$$Y_{\alpha(c_M(t^*))} \notin F_N \circ \pi_N^{-1} \circ \alpha \circ c_M(t^*)$$

by condition (3). But then we have that at $c_M(t^*) \in M$,

$$\alpha_* \circ F_M \circ \pi_M^{-1}(c_M(t^*)) \not\subseteq F_N \circ \pi_N^{-1} \circ \alpha(c_M(t^*)) \tag{4}$$

Conversely, we now prove that if F_N is an abstraction of F_M then $\alpha_* \circ F_M \circ \pi_M^{-1} \subseteq F_N \circ \pi_N^{-1} \circ \alpha$. We will use contradiction. Assume that F_N is an abstraction of F_M but at some point $p \in M$ we have $\alpha_* \circ F_M \circ \pi_M^{-1}(p) \not\subseteq F_N \circ \pi_N^{-1} \circ \alpha(p)$. Then there exists tangent vector $Y_{\alpha(p)} \in T_{\alpha(p)}N$ such that

$$Y_{\alpha(p)} \in \alpha_* \circ F_M \circ \pi_M^{-1}(p) \tag{5}$$

$$Y_{\alpha(p)} \notin F_N \circ \pi_N^{-1} \circ \alpha(p) \tag{6}$$

Since $Y_{\alpha(p)} \in \alpha_* \circ F_M \circ \pi_M^{-1}(p)$, we can write $Y_{\alpha(p)} = \alpha_*(X_p)$ for some (not necessarily unique) tangent vector $X_p \in F_M \circ \pi_M^{-1}(p)$. But since $X_p \in F_M \circ \pi_M^{-1}(p)$ then there exists an integral curve $c_M : I \longrightarrow M$ such that at some $t^* \in I$ we have

$$c_M(t^*) = p \tag{7}$$

$$c'_M(t^*) = X_p \tag{8}$$

To see that such a curve exists assume that such an integral curve does not exist. But then for all curves c_M satisfying (7,8) and for all curves $c_M^B : I \longrightarrow B_M$ such that $\pi_M \circ c_M^B = c_M$ we have that

$$c'_M(t^*) \neq F_M(c_M^B) \Rightarrow X_p \neq F_M(c_M^B) \tag{9}$$

But since this is true for all such c_M^B we obtain

$$X_p \notin F_M(\pi_M^{-1}(c_M(t^*)))$$

which is clearly a contradiction. Therefore, an integral curve satisfying (7,8) always exists.

We know that F_N is an abstraction of F_M. Therefore by definition, for every integral curve c_M of F_M, $\alpha \circ c_M$ must be an integral curve of F_N. Let c_M be the integral curve satisfying (7,8). Then it must be true that

$$(\alpha \circ c_M)' = F_N(c_N^B)$$

for some $c_N^B : I \longrightarrow B_N$ such that $\pi_N \circ c_N^B = \alpha \circ c_M$. But at $t^* \in I$ we have that

$$(\alpha \circ c_M)'(t^*) = \alpha_* \circ c_{M*}(t^*)(1) = \alpha_*(X_p) = Y_{\alpha(p)}$$

But by condition (6), $Y_{\alpha(p)} \notin F_N \circ \pi_N^{-1} \circ \alpha(p)$ and therefore for all curves c_N^B satisfying $\pi_N \circ c_N^B = \alpha \circ c_M$ we get

$$(\alpha \circ c_M)'(t^*) = Y_{\alpha(p)} \notin F_N(c_N^B(t^*))$$

But then $\alpha \circ c_M$ is not an integral curve of F_N which is a contradiction since we assumed that F_N is an abstraction of F_M with respect to the abstracting map α. Therefore, at all points $p \in M$ we must have $\alpha_* \circ F_M \circ \pi_M^{-1} \subseteq F_N \circ \pi_N^{-1} \circ \alpha$. This completes the proof. □

Theorem 10 is the analogue of Theorem 6 for control systems. However, unlike Theorem 6 which required the α-relatedness of two vector fields, Theorem 10 does not require the commutativity of diagram 2. This is actually quite fortunate since, as the following corollaries of Theorem 10 show, *every* control and dynamical system is abstractable by another control system.

Corollary 11 (Abstractable Control Systems). *Every control system $S_M = (B_M, F_M)$ is abstractable by a control system S_N with respect to any abstracting map $\alpha : M \longrightarrow N$.*

Proof: Simply let $B_N = TN$ and $F_N : TN \longrightarrow TN$ equal the identity. Then condition (1) is trivially satisfied. Thus $S_N = (B_N, F_N)$ is an abstraction of S_M. □

As a subcollorary of Corollary 11 we have.

Corollary 12 (Abstractable Dynamical Systems). *Every dynamical system on M is abstractable by a control system with respect to any abstracting map $\alpha : M \longrightarrow N$.*

Proof: Every vector field X can be thought of a trivial control system $S_M = (B_M, F_M)$ where $B_M = M \times \{0\}$ and F_M is equal to $X \circ \pi$. Then Corollary 11 applies. □

Corollary 12 states the fact that *any dynamical system can be exactly abstracted at the cost of nondeterminism.* In local coordinates, Corollaries 11 and 12 simply state the fact that the behavior of any system can be abstracted by a differential inclusion $\dot{x} \in \Re^n$ where x are the local coordinates of interest and n is the dimension of manifold N. However, such an abstraction may not be useful in proving properties. Therefore, it is clear that there is a notion of order among abstractions of a given system.

If one considers fiber subbundles Δ of the tangent bundle TN where at each $q \in N$,

$$\Delta(q) = F_N \circ \pi_N^{-1}(q) \subseteq T_q N \tag{10}$$

for a control system $S_N = (B_N, F_N)$ then Theorem 10 essentially allows us to think of abstractions of a given system $S_M = (B_M, F_M)$ as subbundles $\Delta \subseteq TN$ that satisfy at each point $p \in M$,

$$\alpha_* \circ F_M \circ \pi_M^{-1}(p) \subseteq \Delta(\alpha(p)) \tag{11}$$

and therefore capture all possible tangent directions in which the abstracted dynamics may evolve. Note that Δ is not needed to be a distribution or to have any vector space structure.

It is clear from (10,11) that if Δ is an abstraction of a control system S_M then so is any superset of Δ, say $\bar{\Delta}$ and thus $\bar{\Delta}$ is also an abstraction. But if $\Delta \subset \bar{\Delta}$ then a straightforward application of Theorem 1 shows that $\bar{\Delta}$ is an abstraction of Δ with respect to the identity map $i : N \longrightarrow N$. Therefore, any integral curve of Δ is also an integral curve of $\bar{\Delta}$ but not vice versa. But since Δ has captured all evolutions of S_M which are of interest, $\bar{\Delta}$ can only contain more redundant evolutions which are not feasible by S_M. It is therefore clear that Δ is a more desirable abstraction than $\bar{\Delta}$. This raises a notion of order among abstractions.

Let $S_M = (B_M, F_M)$ be a control system and an abstracting map $\alpha : M \longrightarrow N$ be given. Let control systems $S_{N_1} = (B_{N_1}, F_{N_1})$ and $S_{N_2} = (B_{N_2}, F_{N_2})$ be abstractions of S_M with respect to α. Define at each $q \in N$,

$$\Delta_1(q) = F_{N_1} \circ \pi_{N_1}^{-1}(q) \subseteq T_q N$$
$$\Delta_2(q) = F_{N_2} \circ \pi_{N_2}^{-1}(q) \subseteq T_q N$$

Then we say that S_{N_1} is a better abstraction than S_{N_2}, denoted $S_{N_1} \preceq S_{N_2}$ iff at each point $p \in M$ we have

$$\Delta_1(\alpha(p)) \subseteq \Delta_2(\alpha(p)) \tag{12}$$

It is clear that $\preceq$ is a partial order among abstractions since the order is essentially set inclusion at each fiber. The following Theorem shows that the resulting lattice has a diamond-like structure since there is a unique minimal and maximal element.

Theorem 13 (Structure of Abstractions). *The partial order $\preceq$ has a unique maximal and minimal element.*

Proof: It is easy to see that the unique maximal abstraction is given by $\bar{S} = (TN, i)$ where i is the identity map from TN to TN.

From condition (11) it is clear that it is clear that if

$$\Delta_1(q) = F_{N_1} \circ \pi_{N_1}^{-1}(q) \subseteq T_q N$$
$$\Delta_2(q) = F_{N_2} \circ \pi_{N_2}^{-1}(q) \subseteq T_q N$$

are abstractions of a control system S_M with respect to α then so is the control system $\Delta = \Delta_1 \cap \Delta_2$ where the intersection of the two bundles is defined at each fiber. It is therefore clear that the unique minimal element of $\preceq$ is given by the intersection of all abstractions of S_M. But the intersection of all abstractions can be seen from condition (11) to be the subbundle that satisfies

$$\underline{\Delta}(\alpha(p)) = \alpha_* \circ F_M \circ \pi_M^{-1}(p) \tag{13}$$

for every $p \in M$. □

Therefore the best abstraction results in diagram (2) being commutative.

Once a system abstraction has been obtained, it is useful to propagate properties of interest from the original system to the abstracted system. For control systems, one of those properties is controllability.

Definition 14 (Controllability). Let $S = (B, F)$ be a control system. Then S is called controllable iff given any two points $p_1, p_2 \in M$, there exists an integral curve c such that for some $t_1, t_2 \in I$ we have $c(t_1) = p_1$ and $c(t_2) = p_2$.

Theorem 15 (Controllable Abstractions). *Let control system $S_N = (B_N, F_N)$ be an abstraction of $S_M = (B_M, F_M)$ with respect to some abstracting map α. If S_M is controllable then S_N is controllable.*

Proof: Let q_1 and q_2 be any two points on N. Then let $p_1 \in \alpha^{-1}(q_1)$ and $p_2 \in \alpha^{-1}(q_2)$ be any two points on M. Since F_M on B_M is controllable then there exists an integral curve c_M such that $c_M(t_1) = p_1$ and $c_M(t_2) = p_2$. The curve $\alpha \circ c_M$ satisfies $\alpha \circ c_M(t_1) = q_1$ and $\alpha \circ c_M(t_2) = q_2$. But since F_N is an abstraction of F_M, then $\alpha \circ c_M$ is an integral curve of F_N on B_N. Therefore, the abstracted system is controllable. □

Other properties, such as local accesibility, also propagate. Stability, however, does not propagate since the abstracted system allows redundant evolutions which could be unstable.

6 Conclusions - Issues for Further Research

In this paper, preliminary results on abstracting dynamical and control systems have been presented. A notion of system abstraction has been defined and necessary and sufficient conditions under which one system is an exact abstraction of another have been obtained. Furthermore, a notion of order among abstractions was introduced by ordering the conservativeness of the given abstractions. Finally, desirable system properties were found to propagate from original models to abstracted models.

Issues for further research include, on the theoretical front, extending these results to approximate, discrete and hybrid abstractions. Approximate abstractions approximate integral curves of a given system with some guaranteed margin of error. Discrete abstractions are generated when the codomain of the abstracting map is a finite set. In that case the abstracted behaviors are timed sequences of events and we seek an automaton reproducing these sequences. However, there are many problems in obtaining vector field conditions similar to those of Theorem 10. Some of these issues are topological and have been addressed in the literature [14, 15]. Furthermore, a more challenging problem is the following: *Given* *an original system and a property of interest, find an abstracting map which preserves this property.* Then checking the property at the abstracted system ensures us that original system has the property as well. This is very important for analysis purposes. For example, given a stable system the abstracting map could be a Lyapunov function. Then checking some properties on the Lyapunov function ensures that the original system is stable.

Finally, the developments presented in this paper will be applied to various applications of interest. Particular applications of interest include aircraft and automobile models.

References

1. Pravin Varaiya. Smart cars on smart roads: problems of control. *IEEE Transactions on Automatic Control*, AC-38(2):195–207, 1993.
2. S. Sastry, G. Meyer, C. Tomlin, J. Lygeros, D. Godbole, and G. Pappas. Hybrid control in air traffic management systems. In *Proceedings of the 1995 IEEE Conference in Decision and Control*, pages 1478–1483, New Orleans, LA, December 1995.
3. Robert L. Grossman, Anil Nerode, Anders P. Ravn, and Hans Rischel, editors. *Hybrid Systems.* Lecture Notes in Computer Science. Springer-Verlag, 1993.
4. Panos Antsaklis, Wolf Kohn, Anil Nerode, and Shankar Sastry, editors. *Hybrid Systems II.* Lecture Notes in Computer Science. Springer-Verlag, 1995.
5. R. Alur, T.A. Henzinger, and E.D. Sontag, editors. *Hybrid Systems III.* Lecture Notes in Computer Science 1066. Springer-Verlag, 1996.
6. Peter Caines and Y.J. Wei. The hierarchical lattices of a finite state machine. *Systems and Control Letters*, 25:257–263, 1995.
7. Anuj Puri and Pravin Varaiya. Decidability of hybrid systems with rectangular differential inclusions. In *Computer Aided Verification*, pages 95–104, 1994.
8. T.A. Henzinger and H. Wong-Toi. Linear phase-portrait approximations for nonlinear hybrid systems. In R. Alur, T.A. Henzinger, and E.D. Sontag, editors, *Hybrid Systems III*, Lecture Notes in Computer Science 1066, pages 377–388. Springer-Verlag, 1996.
9. Michael Spivak. *A Comprehensive Introduction to Differential Geometry.* Publish or Perish, 1979.
10. Abraham R., Marsden J., and Ratiu T. *Manifolds, Tensor Analysis and Applications.* Applied Mathematical Sciences. Springer-Verlag, 1988.
11. James R. Munkres. *Analysis on Manifolds.* Addison-Wesley, 1991.
12. Nijmeijer H. and van der Schaft A.J. *Nonlinear Dynamical Control Systems.* Springer-Verlag, 1990.
13. Roger Brockett. Global descriptions of nonlinear control problems; vector bundles and nonlinear control theory. manuscript, 1980.
14. A. Nerode and W. Kohn. Models for hybrid systems: Automata, topologies, controllability, observability. In Robert L. Grossman, Anil Nerode, Anders P. Ravn, and Hans Rischel, editors, *Hybrid System*, pages 317–356. Springer Verlag, New York, 1993.
15. Michael Branicky. Topology of hybrid systems. In *Proceedings of the 32nd IEEE Conference in Decision and Control*, pages 2309–2314, San Antonio, TX, December 1993.