Sensor/Actuator Abstractions for Symbolic Embedded Control Design

Paulo Tabuada

Department of Electrical Engineering,
University of Notre Dame,
Notre Dame, IN 46556
ptabuada@nd.edu
www.nd.edu/∼ptabuada

Abstract. In this paper we consider the problem of developing sensor/actuator abstractions for embedded control design. These abstractions take the form of inequalities relating sensor/actuator characteristics with the continuous dynamics' output. When satisfied, they allow to decouple control design from the choice of sensor/actuators, thus simplifying control design while ensuring implementability.

1 Introduction

The development of control theory has traditionally ignored hardware implementation to focus on the development of a large and important body of theoretical results. Nevertheless, the existing theory is responsible for the wide success of nowadays highly sophisticated and complex controlled systems. The fundamental reason behind this success has been the availability of dedicated computational hardware and sensor/actuators enabling faithful implementations of theoretically developed control laws. However, with the advent of networked embedded control systems we can no longer rely on such assumption. Instead, control algorithms are needed for tiny embedded devices with reduced computational capabilities, low resolution sensors and actuators and strong power limitations. We have thus reached a turning point where we need to rethink the foundations of systems and control theory in order to incorporate the impact of hardware limitations into the behavior achievable by control.

In this paper we take initial steps along this research direction by developing sensor/actuator abstractions for embedded control design. If, on one hand, one would like to have a design theory incorporating implementation details into feedback design, on the other hand, one would also like to restrict such details to the essential minimum. These apparently contradictory objectives can be met by summarizing implementation platform information in a reduced number of parameters. The approach described in this paper captures such platform abstractions in the form of inequalities relating sensor/actuator parameters and the observations of the continuous dynamics, over which specifications are defined. By satisfying these inequalities, it is guaranteed that *any* control design

M. Morari and L. Thiele (Eds.): HSCC 2005, LNCS 3414, pp. 640–654, 2005.

regulating the output behavior of the continuous dynamics can be implemented in a given platform. Alternatively, these inequalities can also be used to define platform requirements sufficient to run such embedded control software. Finally, the introduced inequalities also emphasize several possible tradeoffs between sensor/actuator quantization and saturation characteristics. In particular, we are able to provide answers to the following questions:

Can we determine if a given control design is implementable with certain sensor/actuator quantization characteristics?

Can we compensate poor sensor quantization by good actuator quantization, or vice-versa, in order to implement a given control design?

Can we determine if a given control design is implementable with certain sensor/actuator saturation characteristics?

Can we compensate sensor/actuator quantization with sensor/actuator saturation, or vice-versa?

The sensor/actuator abstractions presented in this paper are developed in the framework of symbolic control that was introduced by the author and coworkers in the sequence of papers [1, 2, 3, 4]. This symbolic approach is based on the existence of finite abstractions (bisimulations) of continuous control systems in several cases of interest including controllable linear systems and flat systems in discrete time. Once these symbolic models are available, it is possible to automatically synthesize (hybrid) controllers enforcing specifications given by regular languages, finite-state machines or temporal logics. Such finite controllers manipulate continuous states and inputs symbolically and allow for simple software implementations.

Recently, there has been an increase in the attention devoted to problems of control with limited resources. Several authors have addressed the problem of control in the presence of limited communication [5, 6, 7, 8] as well as stabilization in the presence of quantization [9, 10]. Closer to the work presented in this paper is the control of systems with quantized inputs. In [11] the effect of input quantization on reachability is analyzed and in [12] input quantization is used as a tool providing a fresh computational perspective of optimal control problems. The presented work differs from quantized control in that the finite symbolic models (bisimulations) used to derive controllers are not obtained by quantizing the inputs. In fact, the objective of this work is precisely to analyze the validity of our symbolic models across a different range of platforms having different quantization but also saturation characteristics. A clear advantage of the proposed approach is the independence of the symbolic model from the implementation platform. Different symbolic approaches to embedded control include maneuver-automata [13] and motion description languages [14, 15] as well as control under limited computational resources [16, 17].

This paper is organized as follows. In Section 2 we describe the models of control systems used throughout the paper and in Section 3 we recall the symbolic approach to embedded control developed by the author and coworkers. Models of sensor/actuator quantization and saturation are introduced in Section 4. We then present abstraction results for sensor quantization in Section 5, actuator quantization in Section 6, sensor saturation in Section 7 and actuator saturation in Section 8. The main contribution summarizing the abstraction results is presented in Section 9 and the paper ends with some discussion of the presented results in Section 10.

2 The Models

2.1 Notation

We introduce some notation required for the remaining paper. When working with vectors $x \in \mathbb{R}^n$ or matrices $A \in \mathbb{R}^{n\times m}$ we shall denote by x^T and A^T the transposed vector and matrix, respectively. The absolute value of a real number α is denoted by $|\alpha|$ while the infinity norm of a vector $x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$ is denoted by $||x||$ and defined as:

$$||x|| = \max_i |x_i| \tag{1}$$

This vector norm induces a norm on matrices when regarded as the representation of linear transformations between normed vector spaces. We shall denote by $||A||$ the matrix norm induced by (1) for any matrix $A \in \mathbb{R}^{n\times m}$. This matrix norm can be computed as:

$$||A|| = \max_i \left(\sum_j |a_{ij}| \right) \tag{2}$$

We will also need some notation to discuss the "size" of sets. Given a set $S \subseteq \mathbb{R}^n$, we denote by $diam(S)$ the diameter of S which is the supremum over the Euclidean distances between every two pairs of points in S. When dealing with a finite collection of sets $S = \{S_i\}_{i\in I}$, we shall use the notation $diam(S)$ to denote $\min_{i\in I} diam(S_i)$.

2.2 Control Systems

In this paper we consider a class of systems which are know to admit finite bisimulations: discrete time linear controllable systems [1, 2]. Even though many of these results carry over to nonlinear flat systems we will restrict our attention to linear systems to make the results more concrete.

Definition 1. *A discrete time linear control system Σ is defined by the following difference equation:*

$$x(t+1) = Ax(t) + Bu(t), \qquad x \in \mathbb{R}^n, \qquad u \in \mathbb{R}^m, \qquad t \in \mathbb{N}$$

where A and B are matrices of appropriate dimensions.

Throughout the remaining paper we will assume that the columns of B are linearly independent. This results in no loss of generality since we can always achieve linear independence by eliminating the inputs associated with linearly dependent columns of B. The Pre operator associated with a linear control system defines the set of all points that can reach in one step a given point $x \in \mathbb{R}^n$:

$$\mathrm{Pre}(x) = \{x' \in \mathbb{R}^n \mid \exists u \in \mathbb{R}^m\,,\ Ax + Bu = x'\}$$

This operator admits the usual extension to sets $S \subseteq \mathbb{R}^n$:

$$\mathrm{Pre}(S) = \bigcup_{x \in S} \mathrm{Pre}(x)$$

For linear systems controllability admits the following simple characterization:

Definition 2. *A discrete time linear control system Σ is said to be controllable when the following matrix has full row rank:*

$$[B|AB|A^2B|\dots|A^{n-1}B] \tag{3}$$

In this case, there are m numbers $k_1, k_2, \dots, k_m$ satisfying $k_i \geq k_{i+1}$, $k_1 + k_2 + \dots + k_m = n$ and the vector space generated by the columns of (3) is also generated by the following basis:

$$\begin{aligned}\mathcal{B} = \{\ & b_1, Ab_1, A^2b_1, \dots, A^{k_1-1}b_1, \\ & b_2, Ab_2, A^2b_2, \dots, A^{k_2-1}b_2, \\ & \vdots \\ & b_m, Ab_m, A^2b_m, \dots, A^{k_m-1}b_m\}\end{aligned}$$

where $b_1, b_2, \dots, b_m$ are the columns of B up to re-ordering. Basis $\mathcal{B}$ induces a natural observation function $H = [H_1^T|H_2^T|\dots|H_m^T]^T : \mathbb{R}^n \to \mathbb{R}^m$ for Σ defined by:

$$H_i x = \begin{cases} 0 & \text{if } x \in \mathcal{B}\backslash\{A^{k_i-1}b_i\} \\ \gamma_i & \text{if } x = A^{k_i-1}b_i \end{cases} \tag{4}$$

for $\gamma_i \in \mathbb{R}^+$ and $i = 1, 2, \dots, m$. The image of the linear map H is then the natural output space of Σ.

The above definition of output map is natural in the sense that it guarantees that the pair (A, H) is observable, that is, the observability matrix:

$$\overline{\mathcal{O}} = \begin{bmatrix} H \\ HA \\ HA^2 \\ \vdots \\ HA^{n-1} \end{bmatrix}$$

has full rank. Full rank of $\overline{\mathcal{O}}$ also implies full rank of the extended observability matrix $\mathcal{O} = [\overline{\mathcal{O}}^T\ (HA^n)^T]^T$. In particular, this implies that $\mathcal{O}$ has a left inverse, simply denoted by $\mathcal{O}^{-1}$. In addition to the extended observability matrix we will also use repeatedly $\gamma_{\max}, \gamma_{\min}$ to denote, respectively, $\max_{i\in\{1,2,\ldots,m\}} \gamma_i$ and $\min_{i\in\{1,2,\ldots,m\}} \gamma_i$.

Specifications for the desired behavior of a controllable linear system Σ are given in terms of a finite number of predicates on the output space $\mathbb{R}^m$ of Σ. These predicates $p \in \mathcal{P}$ are defined by a surjective map $\pi : \mathbb{R}^m \to \mathcal{P}$. Each $p \in \mathcal{P}$ thus defines a set of points in the observation space of Σ by $\{x \in \mathbb{R}^m \mid \pi(x) = p\} = [p]$. We shall say that a point x satisfies predicate p when $\pi(x) = p$, or equivalently $x \in [p]$. In addition, we will abuse language and use the same letter $\mathcal{P}$ to denote the partition of $\mathbb{R}^m$ induced by π and defined by the sets $\{[p]\}_{p\in\mathcal{P}}$.

3 Symbolic Control of Continuous Systems

The sensor/actuator abstractions introduced in this paper are developed in the context of symbolic control of continuous systems based on finite bisimulations. This symbolic control methodology has been developed by the author and co-workers in the series of papers [1, 2, 3, 4]. The essence of the approach is the possibility of constructing a finite abstraction (bisimulation) of the continuous dynamics allowing to translate the initial control problem from the continuous to the purely discrete domain. This process was shown to be possible for a reasonable class of control systems including controllable linear systems and discrete time flat systems. The symbolic model is a finite state representation of all the symbolic output behaviors that can be generated by a given system Σ through a map $\pi : \mathbb{R}^m \to \mathcal{P}$ from the output space of Σ to a finite set of symbols $\mathcal{P}$. Standard supervisory control or temporal logic synthesis techniques can then used to obtain a finite supervisor enforcing any regular or ω-regular language specification on the symbolic output of the finite bisimulation. The resulting finite supervisor is then refined to a hybrid controller combining discrete switching logic with continuous state/input information in order to enforce the desired specification on the continuous plant. Throughout the paper we shall use the expression *symbolic controller* to refer to this type of controller. The architecture of the resulting closed loop is displayed in Figure 1 and can be intuitively described as follows. At any time $t \in \mathbb{N}$, the symbolic controller T_c sends a list of possible symbols $\{\sigma_1(t), \sigma_2(t), \ldots, \sigma_k(t)\}$ to the controlled system. Each symbol σ represents a region $[\sigma] \subseteq \mathbb{R}^n$ in the state space of Σ that can/should be reached in the next time step in order to enforce the specification. Since there are several possible symbols, a choice is made by the white box which represents a discrete decision mechanism. We are thus regarding the white box as a discrete control input capturing the nondeterminism inherent to the specification. Once a symbol has been chosen, it is communicated to T_c (in order to update its internal state) and it is enforced by feedback on Σ. Enforcing symbol σ requires selecting an input u such that the pair (x, u) satisfies $(x, u) \in \mathrm{Pre}([\sigma])$. Continuous input u forces the continuous system Σ to jump from the current state x to a new state contained in the set defined by $[\sigma]$, that is $Ax + Bu \in [\sigma]$. Depending on the sensor/actuator characteristics it may or may not be possible to implement a given

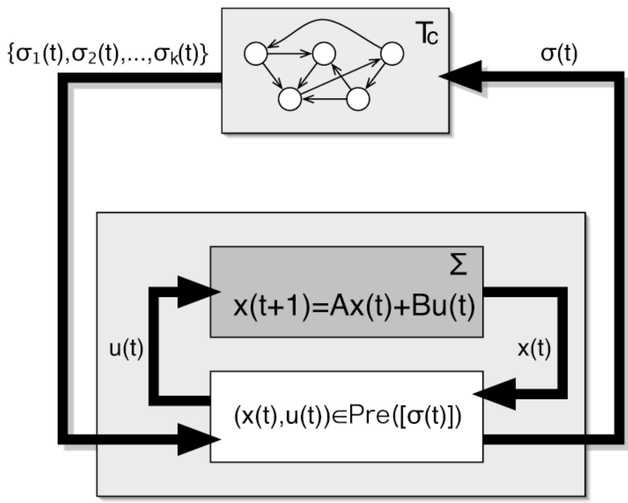

Fig. 1. Feedback interconnection between a symbolic supervisor T_c and a discrete time linear system Σ

symbolic command σ by a pair $(x, u) \in \mathrm{Pre}([\sigma])$. Sensor/actuator characteristics thus limit the behaviors that can be achieved by symbolic control.

To illustrate such limitations consider a control system with output space $\mathbb{R}$ and three predicates neg, zer, pos defined by:

$$\pi(x) = \begin{cases} pos & \text{if } x > 0 \\ zer & \text{if } x = 0 \\ neg & \text{if } x < 0 \end{cases}$$

Consider now the following specifications defined by regular expressions on the labels $\{neg, zer, pos\}$:

$$zer \cdot neg \cdot (neg + pos) \cdot pos^* \qquad\qquad zer \cdot neg \cdot pos \cdot pos^*$$

Even though both specifications use the same predicates it may not be possible to implement them on the same hardware platform. At the third time step, the first specification requires the implementation of a transition to the set associated with $(neg + pos)$ while the second specification requires a transition to the set associated with pos. Actuator saturation may now prevent point $x = -10$ to be controlled to a positive value, while it may still be possible to control it to a negative value such that in a subsequent step it can reach a positive value. In this case the hardware characteristics would allow to implement the first specification but not the second.

Two different approaches can be taken towards the study of symbolic control implementability. Specific results for a particular control design can be given or sufficient, but conservative, results applying to *any* control design can be developed. In this paper we consider sufficient implementability conditions ensuring that *any* control design can be implemented. Since these are only sufficient conditions, specific information regarding the control design will have to be used in order to assert implementability. However, such sufficient conditions provide a valuable working assumption allowing to decouple control design considerations from hardware implementation details. The essence of our approach consists in the following observation:

Since any discrete time controllable linear system Σ admits a finite bisimulation with respect to any choice of predicates on the output space[1]*, ensuring that arbitrary symbolic commands can be implemented on the hardware platform is sufficient to ensure that any control design based on the same predicates is implementable on the hardware platform.*

Our sufficient abstractions will then relate sensor/actuator characteristics with the nature of the predicates defined on the observation space. At the technical level our results rely on several facts related to the existence of finite bisimulations of controllable linear systems [1, 2, 3, 4]. For the purpose of this paper, however, it is sufficient to recall the following:

Given a finite set of predicates $\mathcal{P} = \{p_1, p_2, \ldots, p_l\}$ and a surjective map $\pi : \mathbb{R}^m \to \mathcal{P}$ defined on the output space of controllable discrete time linear system Σ, the symbolic commands σ issued by discrete supervisor T_c correspond to subsets of $\mathbb{R}^n$ defined by the following equalities:

$$p_1 = \pi \circ Hx \tag{5}$$

$$p_2 = \pi \circ H(Ax + Bu_1) \tag{6}$$

$$p_3 = \pi \circ H(A^2x + ABu_1 + Bu_2) \tag{7}$$

$$\vdots$$

$$p_{k_1} = \pi \circ H(A^{k_1-1}x + A^{k_1-2}Bu_1 + \ldots + Bu_{k_1-1}) \tag{8}$$

for some $u_1, u_2, \ldots, u_{k_1-1} \in \mathbb{R}^m$ and $p_1, p_2, \ldots, p_{k_1} \in \mathcal{P}$.

In other words, each symbolic command represents a set defined by the existence of a sequence of predicates $p_1, p_2, \ldots, p_{k_1} \in \mathcal{P}$ and a sequence of inputs $u_1, u_2, \ldots, u_{k_1-1} \in \mathbb{R}^m$ such that the current state satisfies predicate p_1, the next state satisfies predicate p_2, the following state satisfies p_3 and so on.

4 Sensor/Actuator Models

In this paper we are mainly interested in two characteristics of sensors/actuators: quantization and saturation. We model a sensor as a map S from $\mathbb{R}$ to some space (finite or infinite) of measurements M. For simplicity of presentation we will assume the existence of a sensor S_i for each state x_i. The complete state measurement is therefore given by the vector $[S_1(x_1)\ S_2(x_2)\ \ldots\ S_n(x_n)]^T$. We will also assume that $M \subseteq \mathbb{R}$ as this allows to model an ideal sensor by the indentity map on $\mathbb{R}$. Quantization is described by the number Δ_S defining how state values are transformed into measurements:

[1] See Definition 2 for output space.

$$S(x) = \begin{cases} \Gamma_S & \text{if } x \geq \Gamma_S + \frac{1}{2}\Delta_S \\ \Gamma_S - 1 & \text{if } x \in [\Gamma_S - \frac{1}{2}\Delta_S, \Gamma_S + \frac{1}{2}\Delta_S[\\ & \vdots \\ 2\Delta_S & \text{if } x \in [\frac{3}{2}\Delta_S, \frac{5}{2}\Delta_S[\\ \Delta_S & \text{if } x \in [\frac{1}{2}\Delta_S, \frac{3}{2}\Delta_S[\\ 0 & \text{if } x \in [-\frac{1}{2}\Delta_S, \frac{1}{2}\Delta_S[\\ -\Delta_S & \text{if } x \in [-\frac{3}{2}\Delta_S, -\frac{1}{2}\Delta_S[\\ -2\Delta_S & \text{if } x \in [-\frac{5}{2}\Delta_S, -\frac{3}{2}\Delta_S[\\ & \vdots \\ -\Gamma_S + 1 & \text{if } x \in [-\Gamma_S - \frac{1}{2}\Delta_S, -\Gamma_S + \frac{1}{2}\Delta_S[\\ -\Gamma_S & \text{if } x < -\Gamma_S - \frac{1}{2}\Delta_S \end{cases} \tag{9}$$

A sensor S thus maps sets of length Δ_S into its mid-point and saturates with the value Γ_S or $-\Gamma_S$ when the threshold $\Gamma_S + \frac{1}{2}\Delta_S$ or $-\Gamma_S - \frac{1}{2}\Delta_S$ is reached, respectively. The number Γ_S characterizes the saturation of the sensor. A sensor with quantization Δ_S and saturation Γ_S will be called a (Δ_S, Γ_S)-sensor. Given a set of sensors $S_1, S_2, \ldots, S_n$ used to measure the state we will simply refer to the quantization of such set by $\Delta_S = \max_{i\in\{1,2,\ldots,n\}} \Delta_{S_i}$ and we will refer to the saturation of the set by $\Gamma_S = \min_{i\in\{1,2,\ldots,n\}} \Gamma_{S_i}$.

Actuators are similarly described. They are modeled by a map A from an output space $O \subseteq \mathbb{R}$ to $\mathbb{R}$. Actuators are also described by quantization Δ_A and saturation Γ_A. The map A has the same form as (9) but Δ_S and Γ_S are now Δ_A and Γ_A, respectively.

5 Sensor Quantization

In this section we address the effects of sensor quantization on implementability of control designs. In particular, we answer the following question:

How should sensor quantization be related to the predicates $p \in \mathcal{P}$ in order to implement a given design?

Proposition 1. *Let Σ be a discrete time controllable linear system and $\mathcal{P}$ a finite set of predicates on the output space of Σ. If the following inequality is satisfied:*

$$diam(\mathcal{P}) > ||\mathcal{O}||\Delta_S \tag{10}$$

then every symbolic controller enforcing a specification defined over the symbolic output $\mathcal{P}$ is implementable with (Δ_S, ∞)-sensors and $(0, \infty)$-actuators.

Before proving this result we make some remarks regarding inequality (10). As it was intuitively expected, increasing sensor quantization has the unpleasant effect of increasing also the diameter of the observation predicates. Therefore if a certain minimum diameter for the predicates is required to express certain properties, an upper bound on sensor quantization is also being enforced. Furthermore, the linear relation between $diam(\mathcal{P})$ and Δ_S is characterized by the observability properties of Σ as defined by $||\mathcal{O}||$. In fact, a less conservative estimate for the bound on the diameter of $\mathcal{P}$ is given by $diam(\mathcal{P}) > ||HA^n||\Delta_S$ as can be seen from (18) in the proof of Proposition 1. However, the extended observation matrix captures, in a single object, all the continuous dynamics information required for all the abstractions presented in this paper.

Proof. We first consider the case where $m = 1$, that is, Σ only has one input. In this case, the set represented by a predicate $p_i \in \mathcal{P}$ is of the form $[\alpha_i, \beta_i]$, $[\alpha_i, \beta_i[$, $]\alpha_i, \beta_i]$ or $]\alpha_i, \beta_i[$ for $\alpha_i, \beta_i \in \mathbb{R} \cup \{\infty\}$. For simplicity we will only consider the case $[\alpha_i, \beta_i]$ since the same argument applies to the remaining cases. Let σ be a symbolic command issued by T_c. As discussed in Section 3, each such command is associated with a subset of $\mathbb{R}^n$ defined by points $y \in \mathbb{R}^n$ satisfying:

$$
\begin{aligned}
p_1 &= \pi \circ Hy && (11)\\
p_2 &= \pi \circ HAy = \pi \circ H(Ay + Bu_1) && (12)\\
p_3 &= \pi \circ HA^2y = \pi \circ H(A^2y + ABu_1 + Bu_2) && (13)\\
&\vdots\\
p_{k_1} &= \pi \circ HA^{n-1}y = \pi \circ H(A^{k_1-1}y + A^{k_1-2}Bu_1 + \ldots + Bu_{k_1-1}) && (14)
\end{aligned}
$$

for some $u_1, u_2, \ldots, u_{k_1-1} \in \mathbb{R}$ and $p_1, p_2, \ldots, p_{k_1} \in \mathcal{P}$. If a point x will move to $Ax + Bu = y \in [\sigma]$, then by replacing y with $Ax + Bu$ in equations (11) through (14) and using (4), we see that the only constraint involving u is given by:

$$\pi \circ H(A^{k_1}x + A^{k_1-1}Bu) = p_{k_1} \tag{15}$$

Denoting by $\widehat{x}$ the quantized value of x, we have $x = \widehat{x} + d$ with $||d|| \leq \frac{\Delta_S}{2}$. We can therefore rewrite (15) in terms of $\widehat{x}$ which leads to the following equation that has to be satisfied for all $||d|| \leq \frac{\Delta_S}{2}$:

$$HA^{k_1}\widehat{x} + HA^{k_1}d + HA^{k_1-1}Bu \in [\alpha_{k_1}, \beta_{k_1}] \tag{16}$$

A sufficient condition for solvability of the above equation is:

$$HA^{k_1}\widehat{x} + HA^{k_1-1}Bu \in [\alpha_{k_1} + |HA^{k_1}d|, \beta_{k_1} - |HA^{k_1}d|]$$

Since this equation can always be solved for u, provided that the right hand side is a nonempty set, we must have:

$$\beta_{k_1} - |HA^{k_1}d| > \alpha_{k_1} + |HA^{k_1}d| \Leftrightarrow diam(p_{k_1}) = \beta_{k_1} - \alpha_{k_1} > 2|HA^{k_1}d| \tag{17}$$

Furthermore, as:

$$
\begin{aligned}
2|HA^{k_1}d| &\leq 2||HA^{k_1}||||d|| && (18)\\
&= 2||\mathbf{n}^T\mathcal{O}||||d||\\
&\leq 2||\mathbf{n}||||\mathcal{O}||||d||\\
&\leq ||\mathcal{O}||\Delta_S
\end{aligned}
$$

where $\mathbf{n}$ denotes the vector $(0, 0, \ldots, 0, 1) \in \mathbb{R}^n$, we conclude that if:

$$diam(\mathcal{P}) > ||\mathcal{O}||\Delta_S$$

holds, then (17) also holds and a transition from any $x \in X$ to some $Ax + Bu = y \in Y$ can be implemented.

We now consider the general case. Since we can always under-approximate a set associated with $p \in \mathcal{P}$ by the Cartesian product of m sets of the form $[\alpha, \beta]$, that is $\Pi_{j=1}^{n}[\alpha_j, \beta_j]$, and since by (4) the observation function decouples the influence of each input channel, we can apply the previous argument to each of the m input channels obtaining m conditions of the form $diam(\mathcal{P}) > ||\mathcal{O}||\Delta_S$. □

6 Actuator Quantization

We now turn to the effects of actuator quantization and the following related question:

How should sensor and actuator quantization be related to the predicates $p \in \mathcal{P}$ in order to implement a given design?

Proposition 2. *Let Σ be a discrete time controllable linear system and $\mathcal{P}$ a finite set of predicates on the output space of Σ. If the following inequality is satisfied:*

$$diam(\mathcal{P}) > ||\mathcal{O}||\Delta_S + \gamma_{\max}\Delta_A \tag{19}$$

then every symbolic controller enforcing a specification defined on the symbolic output $\mathcal{P}$ is implementable with (Δ_S, ∞)-sensors and (Δ_A, ∞)-actuators.

Equation (19) shows that actuation quantization further contributes to limit the diameter of the predicates. However, we also see that when $diam(\mathcal{P})$ has been fixed by some particular design, several different combinations of sensor/actuator quantization can be used in the implementation.

Proof. We use the notation of the proof of Proposition 1 and start by considering the single input case, that is $m = 1$. To implement a symbolic command σ, the following equation must have a solution in u for every d satisfying $||d|| \leq \frac{\Delta_S}{2}$ (see (16)):

$$HA^{k_1}\widehat{x} + HA^{k_1-1}Bu \in [\alpha_n, \beta_n] - HA^{k_1}d \tag{20}$$

To solve (20) for u it is sufficient to have:

$$\gamma u \in [\alpha_{k_1} + ||\mathcal{O}||\frac{\Delta_S}{2} - HA^{k_1}\widehat{x}, \beta_{k_1} - ||\mathcal{O}||\frac{\Delta_S}{2} - HA^{k_1}\widehat{x}] \tag{21}$$

since $|HA^{k_1}d| \leq ||\mathcal{O}||\frac{\Delta_S}{2}$, as shown in the proof of Proposition 1. Using now $u = z\Delta_A$ with $z \in \mathbb{Z}$ and solving for z we obtain:

$$z \in \frac{1}{\gamma\Delta_A}[\alpha_{k_1} + ||\mathcal{O}||\frac{\Delta_S}{2} - HA^{k_1}\widehat{x}, \beta_{k_1} - ||\mathcal{O}||\frac{\Delta_S}{2} - HA^{k_1}\widehat{x}]$$

Since z is an integer, the previous equation is satisfied only when the right-hand side interval has length greater than 1, that is:

$$\frac{1}{\gamma \Delta_A}\left(\beta_{k_1} - ||\mathcal{O}||\frac{\Delta_S}{2} - \alpha_{k_1} - ||\mathcal{O}||\frac{\Delta_S}{2}\right) > 1$$

which can be rewritten as:

$$diam(p_{k_1}) - ||\mathcal{O}||\Delta_S > \gamma \Delta_A$$

and as it must be satisfied for every $p \in \mathcal{P}$, leads to (19).

We now consider now the multi input case. As in the proof of Proposition 1 we under-approximate a set associated with a predicate $p \in \mathcal{P}$ by a Cartesian product of sets of the form $[\alpha_i, \beta_i]$ and use the previous argument for each of the m input channels. We thus obtain a set of sufficient inequalities of the form

$$diam(p_{k_i}) - ||\mathcal{O}||\Delta_S > \gamma_i \Delta_A$$

which are satisfied by taking γ_i to be $\gamma_{\max}$. □

7 Sensor Saturation

Having discussed quantization effect in the previous sections we now turn to the effects of saturation. The motivation for the results to be presented comes from the following question:

How should sensor saturation be related to the predicates $p \in \mathcal{P}$ in order to implement a given design?

Proposition 3. *Let Σ be a discrete time controllable linear system and $\mathcal{P}$ a finite set of predicates on the output space of Σ. If for every predicate $p \in \mathcal{P}$ the following inclusion holds:*

$$[p] \subseteq \left\{ z \in \mathbb{R}^m \mid ||z|| \leq \Gamma_S / ||\mathcal{O}^{-1}|| \right\} \tag{22}$$

then every symbolic controller enforcing a specification defined on the symbolic output $\mathcal{P}$ is implementable with (α, Γ_S)-sensors and (β, ∞)-actuators for any $\alpha, \beta \in \mathbb{R}_0^+$.

The previous result shows that by properly restricting the output predicates we can achieve implementability, with respect to sensor saturation, independently of the symbolic controller design. A less conservative approach would require inclusion (3) to hold, not for every $p \in \mathcal{P}$, but only for the predicates appearing in the behavior enforced by a particular choice for discrete supervisor T_c. This option, even though less conservative, would no longer be independent of symbolic controller design as it requires knowledge of T_c.

In any case, the effect of sensor actuation is decoupled from the effect of sensor or actuator quantization. This implies that we cannot trade quantization by saturation or vice-versa.

Proof. Given the sensor saturation characteristics, only the states belonging to $\Pi_{i=1}^{n}[-\Gamma_{S_i}, \Gamma_{S_i}]$ can be measured. Conservatively, we further restrict the set of observable states to $[-\Gamma_S, \Gamma_S]^n \subseteq \Pi_{i=1}^{n}[-\Gamma_{S_i}, \Gamma_{S_i}]$. In order to implement symbolic commands issued by T_c, it is sufficient to guarantee that every state trajectory of the controlled behavior remains within the set $[-\Gamma_S, \Gamma_S]^n$. Consider now a symbolic command σ issued by T_c. Such command is associated with a set $[\sigma] \subseteq \mathbb{R}^n$ defined by equations (5) through (8). If the current state x, satisfying

$$Hx = \mathbf{z} \tag{23}$$

will jump to a state $x \in [\sigma]$, then we can compactly write (23) and (5) through (8) as:

$$\mathcal{O}x = Z$$

with:

$$Z = \begin{bmatrix} \mathbf{z} \\ \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_{k_1} \end{bmatrix}$$

for $\mathbf{z}_i \in [p_i]$, $i = 1, 2, \ldots, k_1$. Since $\mathcal{O}$ admits a left inverse by construction, $x = \mathcal{O}^{-1}Z$ and:

$$||x|| = ||\mathcal{O}^{-1}Z|| \leq ||\mathcal{O}^{-1}||||Z||$$

Furthermore, $\mathbf{z}_i \in [p]$ and the assumption $[p] \subseteq \left\{z \in \mathbb{R}^m \mid ||z|| \leq \Gamma_S/||\mathcal{O}^{-1}||\right\}$ implies $||Z|| \leq \Gamma_S/||\mathcal{O}^{-1}||$ from which we conclude $||x|| \leq \Gamma_S$, thus ensuring that controlled trajectories remain in the set $[-\Gamma_S, \Gamma_S]^n$, as desired. □

8 Actuator Saturation

The last considered effect is actuator saturation motivated by the following question:

How should sensor and actuator saturation be related to the predicates $p \in \mathcal{P}$ in order to implement a given design?

Proposition 4. *Let Σ be a discrete time controllable linear system and $\mathcal{P}$ a finite set of predicates on the output space of Σ. If for every predicate $p \in \mathcal{P}$ inclusion (22) holds and:*

$$\Gamma_A > \frac{\Gamma_S}{\gamma_{\min}}\left(1 + ||\mathcal{O}||\right) \tag{24}$$

then every symbolic controller enforcing a specification defined on the symbolic output $\mathcal{P}$ is implementable with (α, Γ_S)-sensors and (β, Γ_A)-actuators for any $\alpha, \beta \in \mathbb{R}_0^+$.

As expected, actuator saturation scales linearly with observation saturation. This is natural since an input making the continuous system jump in one step between two maximally distant points inside the sensor range may be required. This lower bound on input saturation can be reduced by requiring, as part of the specification, that only δ length jumps can be taken. In this case we can replace Γ_S by δ in above expression (24) to reduce the lower bound on Γ_A. Once again we see that saturation effects can be decoupled from quantization effects.

Proof. As usual we treat the single input case first. From the proof of Proposition 2 we know that solvability of (21) for u is a sufficient condition for implementability. It then suffices to ensure that γu can reach the following lower bound:

$$\gamma u > \alpha_{k_1} - ||\mathcal{O}||\frac{\Delta_S}{2} - HA^{k_1}\widehat{x}$$

for all possible values of α_{k_1} and $\widehat{x}$. Since both α_{k_1} and $\widehat{x}$ are bounded by Γ_S, it follows that if $\Gamma_A > \frac{\Gamma_S}{\gamma_{\min}}(1 + ||\mathcal{O}||)$ we can choose $u \in [-\Gamma_A, \Gamma_A]$ such that:

$$\begin{aligned}\gamma u &> \Gamma_S + ||\mathcal{O}||\Gamma_S \\ &\geq \alpha_{k_1} - HA^{k_1}\widehat{x} \\ &\geq \alpha_{k_1} - ||\mathcal{O}||\frac{\Delta_S}{2} - HA^{k_1}\widehat{x}\end{aligned}$$

thus obtaining the desired sufficient condition.

Following the same argument we obtain for the multi-input case m inequalities of the form $\Gamma_A > \frac{\Gamma_S}{\gamma_i}(1 + ||\mathcal{O}||)$ which are enforced by taking γ_i to be $\gamma_{\min}$. □

9 Main Result

For convenience we summarize Propositions 1,2,3 and 4 in the following theorem:

Theorem 1. *Let Σ be a discrete time controllable linear system and $\mathcal{P}$ a finite set of predicates on the output space of Σ. If the following inequalities hold:*

$$diam(\mathcal{P}) > ||\mathcal{O}||\Delta_S + \gamma_{\max}\Delta_A \tag{25}$$

$$\Gamma_A > \frac{\Gamma_S}{\gamma_{\min}}\big(1 + ||\mathcal{O}||\big) \tag{26}$$

and for every predicate $p \in \mathcal{P}$ the following inclusion also holds:

$$[p] \subseteq \Big\{z \in \mathbb{R}^m \mid ||z|| \leq \Gamma_S/||\mathcal{O}^{-1}||\Big\} \tag{27}$$

then every symbolic controller enforcing a specification defined on the symbolic output $\mathcal{P}$ is implementable with (Δ_S, Γ_S)-sensors and (Δ_A, Γ_A)-actuators.

Theorem 1 collects, in the form of inequalities, the abstractions developed in this paper. These inequalities represent simple and intuitive conditions for implementability: the sets $[p]$ have to be large enough to accommodate the errors introduced by sensor and actuator quantization as described by (25); sensor saturation limits the range of states that can be measured and the symbolic outputs $p \in \mathcal{P}$ must represent sets $[p]$ reflecting such state limitations as described by (27); and actuator saturation has to permit arbitrary jumps between states that can be measured as described by (26). In all these equalities the system dynamics plays a fundamental role defined by the presence of the extended observation matrix in all the inequalities. A large value for $||\mathcal{O}||$ poses additional limitations on the relation between sensing, actuation and output predicates since it implies an increase of the "size" of the sets $[p]$, an increase on actuator saturation and a reduction on the "size" of the output space that can be used to define predicates.

10 Discussion

The sensor/actuator abstractions presented in this paper are clearly conservative and can be improved in several different ways. However, improving the equalities in Theorem 1 would require embedding quantization and saturation information into control design. As with any design problem, the right level of abstraction depends on the particular problem being solved. When the presented abstractions fail to hold, determining implementability of a given design requires a deeper analysis of the effects of the implementation platform in the given design. However, the presented results are still useful as a working assumption for the early design phases decoupling control requirements from hardware requirements.

There several other hardware requirements that should also be addressed and have not been discussed in this paper. These include (real-time) computational capabilities and power consumption among others. Developing similar abstractions to capture the influence of these properties on embedded control design in currently being addressed by the author.

References

1. Tabuada, P., Pappas, G.J.: Finite bisimulations of controllable linear systems. In: Proceedings of the 42nd IEEE Conference on Decision and Control, Hawaii (2003)
2. Tabuada, P.: Flatness and finite bisimulations in discrete time. In: Proceedings of the Sixteenth International Symposium on Mathematical Theory of Networks and Systems, Leuven, Belgium (2004)
3. Tabuada, P., Pappas, G.J.: From discrete specifications to hybrid control. In: Proceedings of the 42nd IEEE Conference on Decision and Control, Hawaii (2003)
4. Tabuada, P., Pappas, G.J.: Linear Time Logic control of linear systems. IEEE Transaction on Automatic Control (2004) Under review, available at `www.nd.edu/~ptabuada`.
5. Hristu, D., Morgansen, K.: Limited communication control. Systems and Control Letters **37** (1999) 193 – 205
6. Wong, W.S., Brockett, R.: Systems with finite communication bandwidth constraints II: Stabilization with limited information feedback. IEEE Transactions on Automatic Control **44** (1999) 1049–1053

7. Hespanha, J., Ortega, A., Vasudevan, L.: Towards the control of linear systems with minimum bit-rate. In: Proceedings of the Int. Symp. on the Mathematical Theory of Networks and Systems, Notre Dame, Indiana (2002)
8. Tatikonda, S., Mitter, S.: Control under communication constraints. IEEE Transactions on Automatic Control **49** (2004) 1056–1068
9. Elia, N., Mitter, S.: Stabilization of linear systems with limited information. IEEE Transactions on Automatic Control **46** (2001) 1384–1400
10. Liberzon, D.: Hybrid feedback stabilization of systems with quantized signals. Automatica **39** (2003) 1543–1554
11. Bicchi, A., Marigo, A., Piccoli, B.: On the rechability of quantized control systems. IEEE Transaction on Automatic Control (2002)
12. Pancanti, S., Leonardi, L., Pallottino, L., Bicchi, A.: Optimal control of quantized linear systems. In Tomlin, C., Greenstreet, M.R., eds.: Hybrid Systems: Computation and Control. Lecture Notes in Computer Sience. Springer-Verlag (2002) 351–363
13. Frazzoli, E.: Explicit solutions for optimal maneuver-based motion planning. In: Proceedings of the 42nd IEEE Conference on Decision and Control, Maui, Hawaii (2003)
14. Hristu-Varsakelis, D., Egerstedt, M., Krishnaprasad, P.: On the structural complexity of the motion description language mdle. In: Proceedings of the 42nd IEEE Conference on Decision and Control, Maui, Hawaii (2003)
15. Austin, A., Egerstedt, M.: Mode reconstruction for source coding and multi-modal control. In Alur, R., Pappas, G.J., eds.: Hybrid Systems: Computation and Control. Volume 2993 of Lecture Notes in Computer Science. Springer-Verlag (2004) 36–49
16. Krogh, B., Maler, O., Mahfoudh, M.: On control with bounded computational resources. In Damm, W., Olderog, E.R., eds.: Formal Techniques in Real-Time and Fault-Tolerant Systems, 7th International Symposium, FTRTFT 2002. Volume 2469 of Lecture Notes in Computer Science. Springer-Verlag (2002) 147–164
17. Palopoli, L., Pinello, C., Vincentelli, A.S., Elghaoui, L., Bicchi, A.: Synthesis of robust control systems under resource constraints. In Tomlin, C., Greenstreet, M.R., eds.: Hybrid Systems: Computation and Control. Lecture Notes in Computer Sience. Springer-Verlag (2002) 337–350