Decomposition of Human Motion into Dynamics Based Primitives with Application to Drawing Tasks

D. Del Vecchio[1], R. M. Murray, P. Perona
Division of Engineering and Applied Science
1200 E. California Boulevard mail stop 107-81
California Institute of Technology
Pasadena, CA 91125

Abstract
Using tools from dynamical systems and systems identification we develop a framework for the study of primitives for human motion, which we refer to as *movemes*. The objective is understanding human motion by decomposing it into a sequence of elementary building blocks that belong to a known alphabet of dynamical systems. In this work we address the problem of defining conditions under which collections of signals are well-posed according to a dynamical model class M and then can generate movemes. Based on the assumption of well-posedness, we develop segmentation and classification algorithms in order to reduce a complex activity into the sequence of movemes that have generated it. Using examples we show that the definition of well-posedness can be applied in practice and show analytically that the proposed algorithms are robust with respect to noise and model uncertainty. We test our ideas on data sampled from five human subjects who were drawing figures using a computer mouse. Our experiments show that we are able to distinguish between movemes and recognize them even when they take place in activities containing more than one moveme at a time.

Keywords: identification, parameter estimation, classification, signal segmentation, data acquisition, laboratory experiments

1 Introduction

Building systems that can detect and recognize human actions and activities is an important goal of modern engineering. Applications range from human-machine interfaces to security to entertainment. With the development of information technology we can expect that computer systems will be increasingly embedded in our environment, so that human-machine interaction will need interfaces that are easier to use and more natural. In particular the possibility of interacting with computerized environment without the need for special external equipment is attractive. As humans use their visual system and auditory system to communicate, several works (see for example (Laptev and Lindeberg, 2001; Waldherr *et al.*, 1998) and the earlier work on building human-machine interfaces using vision (Goncalves *et al.*, 1995; Munich and Perona, 1996; Wilson and Bobick, 1995; Yacoob and Davis, 1996; Wellner, 1991)) ask the question of whether it is possible to develop computerized equipment able to communicate with humans in similar way. As described extensively in (Collins *et al.*, 2000) there is also an immediate need for automated surveillance systems in commercial, law enforcement, and military applications. Surveillance cameras are present in banks, stores, and parking lots; it is desirable to develop continuous automated monitoring to alert security officers about suspicious human activity while there is still time to prevent a possible crime. Other applications include video-games and animation where virtual human motion is based on the learning and description of real human motion (see for example (Zordan and Hodgins, 1999) and (Silva *et al.*, 1997)). Another important application is biomechanics (see for example (Pedotti *et al.*, 1989)).

A fundamental problem in detecting and recognizing human action is one of representation. Our point of view is that human activity should be decomposed into building blocks which belong to an "alphabet" of elementary actions; for example the activity "answering the phone" could be decomposed into the sequence "step-step-step-reach-lift", where "step", "reach" and "lift" may not be further decomposed. We refer to these primitives of motion as *movemes*. Our aim is then to build an alphabet of

[1]Corresponding author. Fax: 626/796-8914; e-mail: ddomitilla@cds.caltech.edu

movemes which one can compose to represent and describe human motion similar to the way phonemes are used in speech. In speech recognition phonemes are defined to be the smallest units of sound from which words are composed and hidden Markov models are widely used to separate a stream of speech into discrete phonemes that can be then reassembled into words (Rabiner and Juang, 1993). The word "moveme" intended as primitive of motion was invented by (Bregler and Malik, 1997). They studied periodic or stereotypical motions such as walking or running where the motion is always the same and therefore their movemes, like the phonemes, were repeatable segments of trajectory. (Goncalves *et al.*, 1998) studied motions that were parametrized by an initial condition and a target. They proposed that movemes ought to be parametrized by goal and style parameters. Their moveme models are phenomenological and non-causal. In this paper we attempt to define movemes in terms of causal dynamical systems.

The idea of dynamical primitives of motion has also appeared in neurobiology studies. (Bizzi and Mussa-Ivaldi, 1999) pose the question whether the motor behavior of vertebrates is based on simple units (motor primitives) that can be combined flexibly to accomplish a variety of motor tasks and experiments have provided evidence for a modular organization of the spinal cord in frogs and rats. (Mussa-Ivaldi *et al.*, 1994) ran experiments which showed that the fields induced by the focal activation of the spinal cord follow a principle of vectorial summation, so that a variety of motor control polices can be obtained from a simple linear combination of few control modules. This is suggesting that we could represent elementary actions in terms of vector fields (dynamical systems) and generate more complicated trajectories by properly combining such vector fields. Experimental results in (Kawato, 1999) and (Flanagan and Wing, 1997) support the idea that kinematic and dynamic internal models are utilized in movement planning and control. The internal model hypothesis proposes that the brain acquires an inverse dynamic model of the object to be controlled through motor learning after which motor control can be executed mostly in a feed-forward manner. This way standard motor tasks for which the brain has already learned an inverse dynamic model can be accomplished using a feed-forward control which takes the desired trajectory as input and produces the right motor command to the controlled object. Under normal conditions, the inverse dynamic model calculates motor commands which appropriately compensate the dynamics, so that the realized trajectory is a good reproduction of the desired one. Thus, the role of dynamics in the description of human motion seems to be an important one.

What is the alphabet of movemes? Which are the dynamical models that we should use to represent them? Can a continuous trajectory of a human body be decomposed automatically into its component movemes? To answer these questions we introduce a formal definition of moveme and we set up classification and segmentation problems which in the dynamical systems framework can be appropriately formalized. Standard system identification tools and stability arguments can then be applied to derive analytical error analysis for classification and segmentation algorithms so as to obtain performance estimates in the presence of noise and modeling uncertainties.

This paper is organized as follows. In Section 2 we define *movemes* formally according to a dynamical systems framework and provide a practical well-posedness definition that establishes when sets of actions can define movemes. We introduce the classification problem as a standard problem of pattern recognition (Bishop, 1995; Vapnik, 1995). In Section 3 we introduce the segmentation problem and in Section 4 propose a solution for it. The problem of segmenting data streams originating from different unknown or partially known processes which alternate in time is a general problem of interest to various areas, see for example (Gustafsson, 2000; Lavielle, 1998; Willsky and Jones, 1976). We propose a solution to the problem in our particular scenario in which each one of the segments has been generated from the perturbed version of a linear dynamical system belonging to a finite known set of possible linear models. By using system identification techniques (Ljung, 1999; Söderström and Stoica, 1989) and pattern recognition techniques (Bishop, 1995; Vapnik, 1995) we develop off-line segmentation and classification algorithms and provide an analytical error analysis. In Section 5, we test our ideas on data sampled from five human subjects engaged in drawing houses, cars, ships and suns using a computer mouse. We show how to find movemes from elementary actions (such as reaching a point or drawing a line) and show how to check for well-posedness. We find a set of motions which we refer to as "free

motion" for which the well-posedness test fails, and therefore cannot define movemes. We then apply the segmentation algorithm using the identified movemes to a stream of data of different subjects and observe a performance (correct identification of segmenting points and correct classification of the composing segments) of about 90%. As a final application we show how we can recognize shapes belonging to different categories (ship, car, house) based on an elementary categorization algorithm that uses the segmentation algorithm output for the discrimination. We obtain a categorization error of 5%.

2 Dynamical Definition of Moveme

We provide in this section some theoretical background, a formal definition of moveme in the dynamical systems framework, and properties which descend from the proposed definition. We introduce the classification problem and provide a link between the given theoretical notions with properties that classes of actions should have in order to define what we call a well-posed set of movemes.

2.1 Definitions, properties, choice of model class

Let $M(\Theta)$ denote a linear time invariant (LTI) system class parameterized by $\Theta \in E$, E a linear space, and let $\mathcal{U}$ denote a class of inputs. Let $y(t) = Y(M(\Theta)|_{u,x_0})(t)$, for $t \geq t_0$, denote the output of $M(\Theta)$ once parameter $\Theta \in E$, input $u \in \mathcal{U}$, and initial conditions x_0 have been chosen. Let $\theta \in E' \subset E$ be a parameter lying in a subspace of E, and define a map $\Upsilon : E \to E'$. We write $\theta = \Upsilon(\Theta)$ to represent the transformation from $\Theta \in E$ to the reduced set of parameters $\theta \in E'$.

Definition 2.1. Let $M^1 = \{M(\Theta)|\theta \in \mathcal{C}^1\}$ and $M^2 = \{M(\Theta)|\theta \in \mathcal{C}^2\}$ denote two subsets in M with $\mathcal{C}^j \subset E'$ for $j = 1, 2$. M^1 and M^2 are said to be *dynamically independent* if

(i) the class of systems M and the class of inputs $\mathcal{U}$ are such that

$$Y(M(\Theta_1)|_{u_1,x_0})(t) = Y(M(\Theta_2)|_{u_2,x_0})(t), \qquad \forall t \geq t_0$$

if and only if $(\Theta_1, u_1) = (\Theta_2, u_2)$ for $u_1 \in \mathcal{U}$ and $u_2 \in \mathcal{U}$;

(ii) the sets $\mathcal{C}^1$ and $\mathcal{C}^2$ are non empty, bounded, and have trivial intersection, i.e. $\mathcal{C}^1 \bigcap \mathcal{C}^2 = \{\emptyset\}$.

Each of the elements of a set $\mathcal{M}$ of mutually dynamically independent model sets is called a *moveme*.

Definition 2.2. A signal $y(t)$, $t \in (t_0, t_M)$ is said to be *segmentable* if there exists a sequence of times $\tau_0 < \tau_1 < ... < \tau_{l-1} < \tau_l$, $l \geq 2$, with $t_0 = \tau_0$ and $\tau_l = t_M$, such that

$$y(t) = y_i(t - \tau_{i-1}) := Y(M(\Theta_i)|_{u_i,x_{\tau_{i-1}}})(t - \tau_{i-1}), \qquad t \in [\tau_{i-1}, \tau_i]$$

for some u_i, Θ_i, $x_{\tau_{i-1}}$ with $(\Theta_i, u_i) \neq (\Theta_{i-1}, u_{i-1})$ for any i. The sequence $(y_1(t - \tau_0), ..., y_l(t - \tau_{l-1}))$ is referred to as the *segmentation of* $y(t)$.

From these definitions some straightforward properties follow.

Proposition 2.1. *A moveme output* $y^i(t) = Y(M^i(\Theta^*)|_{u^*,x_{t_0}})(t)$, $t \in [t_0, T]$, *is not segmentable.*

Proof. Assume by contradiction that $(y_1(t - t_0), y_2(t - \tau))$ is the segmentation of $y^i(t)$ for some $\tau \in (t_0, T)$. Then from Definition 2.2 $y_1(t - t_0) = Y(M(\Theta_1)|_{u_1,x_{t_0}})(t - t_0)$, $t \in [t_0, \tau]$ and $y_2(t - \tau) = Y(M(\Theta_2)|_{u_2,x_\tau})(t - \tau)$, $t \in [\tau, T]$. Also $y_1(t - t_0) = Y(M^i(\Theta^*)|_{u^*,x_{t_0}})(t - t_0)$, $t \in [t_0, \tau]$ and $y_2(t - \tau) = Y(M^i(\Theta^*)|_{u^*,x_\tau})(t - \tau)$,$t \in [\tau, T]$. Therefore $Y(M(\Theta_1)|_{u_1,x_0})(t - t_0) = Y(M(\Theta^*)|_{u^*,x_0})(t - t_0)$, $Y(M(\Theta_2)|_{u_2,x_\tau})(t - \tau) = Y(M(\Theta^*)|_{u^*,x_\tau})(t - \tau)$ which by (i) of Definition 2.1 implies $(\Theta_1, u_1) = (\Theta^*, u^*)$, $(\Theta_2, u_2) = (\Theta^*, u^*)$ which in turn implies $(\Theta_1, u_1) = (\Theta_2, u_2)$, contradicting Definition 2.2. □

Proposition 2.2. *If* $y(t)$, $t \in [t_0, T]$ *is segmentable, then the segmentation is unique.*

Proof. It is sufficient to prove this for the case in which $y(t)$ is composed of only two segments. Let $(y_1(t-t_0), y_2(t-\tau^*))$ be the segmentation of $y(t)$. Suppose there is a $\tau < \tau^*$ such that $(\tilde{y}_1(t-t_0), \tilde{y}_2(t-\tau))$ is another segmentation, then since $\tau < \tau^*$ we have by Definition 2.2 $\tilde{y}_1(t-t_0) = Y(M(\Theta_1)|_{u_1,x_{t_0}})(t-t_0)$, $t \in [t_0, \tau]$, and $\tilde{y}_2^a(t-\tau) = Y(M(\Theta_1)|_{u_1,x_\tau})(t-\tau)$ for $t \in [\tau, \tau^*]$ while $\tilde{y}_2^b(t-\tau^*) = Y(M(\Theta_2)|_{u_2,x_{\tau^*}})(t-\tau^*)$ for $t \in [\tau^*, T]$ which means by Definition 2.2 that $\tilde{y}_2(t-\tau)$ is segmentable and its segmentation is $(\tilde{y}_2^a(t-\tau), \tilde{y}_2^b(t-\tau^*))$. Therefore $(\tilde{y}_1(t-t_0), \tilde{y}_2(t-\tau))$ is not a segmentation according to Definition 2.2 since $\tilde{y}_2(t-\tau)$ is not output of model M for a choice of parameter, initial conditions and input as the definition establishes. The same argument holds for $\tau > \tau^*$. In the case in which $y(t)$ is composed by more than two segments we can apply the same argument by considering two segments at a time. □

In this paper, we choose our model class M and input u as asymptotically stable linear systems driven by a unit step input with full state output:

$$\begin{aligned} \dot{x} &= Ax + b \\ y &= x \ , \end{aligned} \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$, $x = (x_1, ..., x_n) \in \mathbb{R}^n$, $b \in \mathbb{R}^n$, so that $\Theta = (A|b) \in E = \mathbb{R}^{n\times(n+1)}$ and $\theta = A \in E' = \mathbb{R}^{n\times n}$, with $\Upsilon(A|b) = A$. For such a class of models we make the following assumption.

Assumption 2.1. Given $x(t)$ as the output of model (1) we assume that the initial condition x_0 is such that for any $v \in \mathbb{R}^{n+1}$,

$$v^T\overline{x}(t) = 0, \ t \in [t_1, t_2], \ t_2 > t_1 \quad \Longrightarrow \quad v = 0,$$

where $\overline{x} = (x^T, 1)^T$.

This assumption means that the description that model (1) provides for $x(t)$ is minimal in the sense that $x(t)$ cannot also be described by a lower order dynamical system. In fact if $v^T\overline{x}(t) = 0$, $t \in [t_1, t_2]$, $t_2 > t_1$ for some $v \neq 0$ then $x_n(t) = \alpha_0 + \alpha_1 x_i(t) + ... + \alpha_{n-1}x_{n-1}(t)$ for any t, therefore the dynamics can be described just in terms of $x_1(t), ..., x_{n-1}(t)$ and $x_n(t)$ can be derived algebraically. A direct consequence of such an assumption is that we have a one-one correspondence between $x(t)$ and parameters $(A|b)$ of model (1), so that we have the following lemma.

Lemma 2.1. *Let $x(t)$ and $z(t)$ be generated by two LTI systems*

$$\begin{aligned} \dot{x} &= A_1x + b_1 \\ \dot{z} &= A_2z + b_2 \end{aligned}$$

and let Assumption 2.1 hold. Then $z(t) = x(t)$ for all t if and only if $(A_1|b_1) = (A_2|b_2)$.

Proof. ($\Leftarrow$) If $(A_1|b_1) = (A_2|b_2)$ then $z(t) = x(t)$ for all t by uniqueness of solutions.
($\Rightarrow$) If $z(t) = x(t)$ for all t then $\dot{z}(t) = \dot{x}(t)$ for all t, so that $A_1x + b_1 = A_2z + b_2$. This implies $[(A_1|b_1) - (A_2|b_2)]\overline{x}(t) = 0$ for all t, which by Assumption 2.1 (applied to each column) implies $(A_1|b_1) = (A_2|b_2)$. □

This lemma shows that property (i) of Definition 2.1 is satisfied by our choice of M and $\mathcal{U}$. Property (ii) is verified by choosing for example $\mathcal{C}^j$, $j = 1, ..., m$, as balls in $\mathbb{R}^{n\times n}$ with centers $A_c^j \in \mathbb{R}^{n\times n}$, $j = 1, ..., m$, and radii r_j, such that:

$$\begin{aligned} &\mathcal{C}^j = B_{r_j}(A_c^j) \ , && j = 1, ..., m \\ &\mathcal{C}^j \bigcap \mathcal{C}^k = \{\emptyset\}, && j \neq k \end{aligned} \tag{2}$$

where m is the number of movemes and the matrix norm is the Frobenius norm. In what follows we assume that the sets $\mathcal{C}^j$ are described by equation (2). Then we have constructed a set $\mathcal{M} = \{M^1, ..., M^m\}$

of m movemes where $M^k = \{M((A|b))|A \in \mathcal{C}^k\}$, for $k \in \{1, ..., m\}$ and M is in the form given by equation (1).

Given any signal $x(t)$ we can determine a good representative of such a signal in the class of models (1) by minimizing the cost function (see for example (Ljung, 1999)):

$$(\hat{A}|\hat{b}) = \arg\min_{(A|b)} \frac{1}{2} \int_{t_0}^{T} (\dot{x} - (A|b)\overline{x})^T (\dot{x} - (A|b)\overline{x}) dt \tag{3}$$

with $\overline{x} = (x^T, 1)^T$, which gives the least squares estimate of parameters $(\hat{A}|\hat{b})$ so to get the estimate of x in model class (1) as

$$\dot{\hat{x}} = \hat{A}\hat{x} + \hat{b}, \qquad \hat{x}(t_0) = x(t_0).$$

In the case in which $x(t)$ has been generated by (1), by virtue of Assumption 2.1 it is easy to check that (3) leads to $(\hat{A}|\hat{b}) = (A|b)$, so that if $A \in \mathcal{C}^j$, for some $j \in \{1, ..., m\}$ we can classify $x(t)$ as output of moveme M^j just by finding $k \in \{1, .., j, ..m\}$ such that $\hat{A} \in \mathcal{C}^k$. This is equivalent by virtue of (2) to finding $k \in \{1, .., j, ..m\}$ such that $\|\hat{A} - A_c^k\| \leq r_k$, whose solution is unique since the sets $\mathcal{C}^k$ are all not intersecting. Then

$$\arg_{k\in\{1,..,j,..,m\}}\{\|\hat{A} - A_c^k\| \leq r_k\} = \arg_{k\in\{1,..,j,..,m\}}\{\|A - A_c^k\| \leq r_k\} = j$$

The following section addresses the same classification problem in a more general situation in which $x(t)$ has been generated by a perturbed version of system (1).

2.2 Classification Problem

Let the signal $x(t)$ be generated by the perturbed version of (1):

$$\begin{aligned} \dot{x} &= Ax + b + d(t) \\ y &= x \end{aligned} \tag{4}$$

with $A \in \mathcal{C}^j$, for some $j \in \{1, ..., m\}$ and $d(t)$ is a bounded realization of white noise. Under what conditions on A and $d(t)$ can we still classify $x(t)$ as output of moveme M^j? Since $A \in \mathcal{C}^j$, there exists $\delta < r_j$ such that $A = A_c^j + \delta U$ with U a unit norm matrix and A_c^j center of $\mathcal{C}^j$. Then system (4) becomes

$$\begin{aligned} \dot{x} &= (A_c^j + \delta U)x + b + d(t) \\ y &= x \ . \end{aligned} \tag{5}$$

Then the problem of classifying $x(t)$ as output of moveme M^j becomes the same as identifying j in system (5) for some conditions on δ and $d(t)$. In the previous section we showed that if $d(t) = 0$ then we can exactly identify $A_c^j + \delta U$ and then correctly classify $x(t)$. The presence of $d(t)$ induces an estimation error so that $\hat{A}$ will not be equal to $A_c^j + \delta U$, but it is not necessary to achieve equality for our purpose as the following lemma shows.

Lemma 2.2. *Let $x(t)$, $t \in [t_0, T]$ be generated by (5), where A_c^j is the center of $\mathcal{C}^j$ for some $j \in \{1, ..., m\}$ as in (2). Let $\hat{A}$ be the least squares estimate according to (3). There exist positive constants $\overline{d}$ and $\overline{\delta}$ such that if $\delta \leq \overline{\delta}$ and $\|d(t)\| \leq \overline{d}$, then*

$$arg_{k\in\{1,...j,...m\}}\{\|\hat{A} - A_c^k\| \leq r_k\} = j$$

Proof. By equation (3) we have

$$(\hat{A}|\hat{b}) = \left(\int_{t_0}^{T} \dot{x}(t)\ \overline{x}(t)^T dt\right) \left(\int_{t_0}^{T} \overline{x}(t)\ \overline{x}(t)^T dt\right)^{-1}$$

where we can invert $\left(\int_{t_0}^{T} \overline{x}(t)\,\overline{x}(t)^T\right)$ if either $d(t) = 0$ by Assumption 2.1, or $d(t) \neq 0$ by the fact that $d(t)$ is realization of white noise that is uncorrelated in time. Using equation (5), this expression becomes

$$(\hat{A}|\hat{b}) = (A_c^j + \delta U|b) + \left(\int_{t_0}^{T} d(t)\overline{x}(t)^T dt\right)\left(\int_{t_0}^{T} \overline{x}(t)\,\overline{x}(t)^T dt\right)^{-1}$$

so that

$$(\hat{A}|\hat{b}) - (A_c^j|b) = (\delta U|0) + \left(\int_{t_0}^{T} d(t)\overline{x}(t)^T dt\right)\left(\int_{t_0}^{T} \overline{x}(t)\,\overline{x}(t)^T dt\right)^{-1}.$$

Thus we have

$$\|\hat{A} - A_c^j\| \leq \|(\hat{A}|\hat{b}) - (A_c^j|b)\| \leq \overline{\delta} + \overline{d}\,c$$

where $\overline{\delta}$ and $\overline{d}$ are upper bounds on δ and $d(t)$, and c is a suitable positive constant which exists since $x(t)$ is bounded by the stability properties of the dynamics. Then in order for $\|\hat{A} - A_c^k\| \leq r_k$ to hold for $k = j$ it is sufficient that

$$\|\hat{A} - A_c^j\| \leq \overline{\delta} + \overline{d}\,c \leq r_j \tag{6}$$

which is verified if, for example, $\overline{\delta} = r_j/2$ and $\overline{d} = r_j/(2c)$, which give upper bounds on δ and $d(t)$. Note that the uniqueness of the solution for k comes from the fact that the sets $\mathcal{C}^k$, $\mathcal{C}^j$ for $k \neq j$ are not intersecting as equation (2) guarantees. If such a requirement is not satisfied even when equation (6) holds, then the solution $k \in \{1, ...j, ...m\}$ of $\|\hat{A} - A_c^k\| \leq r_k$ may not be unique, leading to ambiguity in the classification. □

2.3 Well-posedness

As the previous section highlighted, the basic requirement for solving the classification problem is the one of having non intersecting sets in parameter space characterizing the sets of dynamical models M^j, $j = 1, ..., m$. In practice the sets $\mathcal{C}^j$ and $\mathcal{C}^k$, $j \neq k$ may be not defined *a priori* but are derived from finite sets of signals $\mathcal{S}^j$ and $\mathcal{S}^k$, whose characteristics make each element of one set different from each element of the other and therefore we can say that they define two classes of signals. When can these two classes of signals define two movemes M^j, M^k according to Definition 2.1? Let the two classes $\mathcal{S}^j$ and $\mathcal{S}^k$ be composed by signals $s_i^j(t) = Y(M(\Theta_i^j)|_{x_{0i}^j, u_i^j})(t)$, for $s_i^j(t) \in \mathcal{S}^j$, and $s_i^k(t) = Y(M(\Theta_i^k)|_{x_{0i}^k, u_i^k})(t)$, for $s_i^k(t) \in \mathcal{S}^k$. Let $\mathcal{F}_M$ be an estimation procedure establishing a one to one mapping between the signal $Y(M(\Theta)|_{x_0,u})(t)$ and the couple (Θ, u) which exists by virtue of (i) of Definition 2.1 Then we have

$$\begin{aligned} (\Theta_i^k, u_i^k) &= \mathcal{F}_M(s_i^k(t)) \qquad & s_i^k(t) \in \mathcal{S}^k \\ (\Theta_i^j, u_i^j) &= \mathcal{F}_M(s_i^j(t)) \qquad & s_i^j(t) \in \mathcal{S}^j. \end{aligned}$$

Let $f_s : (E \times \mathcal{U}) \to E$ be the selection operator, such that $f_s(\Theta, u) = \Theta$, which selects the first element of the couple (Θ, u). Then define $f_M := \Upsilon \circ f_s \circ \mathcal{F}_M$, which associates to each signal $s(t)$ the corresponding parameter θ lying in $E' \subset E$. We can then write that $\mathcal{C}^j$ is the image of $\mathcal{S}^j$ through f_M and the same for $\mathcal{C}^k$:

$$\begin{aligned} f_M(\mathcal{S}^j) &= \mathcal{C}^j \\ f_M(\mathcal{S}^k) &= \mathcal{C}^k. \end{aligned} \tag{7}$$

Definition 2.3. Classes of signals $\mathcal{S}^j$ and $\mathcal{S}^k$ with elements $s_i^j(t) = Y(M(\Theta_i^j)|_{x_{0i}^j, u_i^j})(t)$, for $s_i^j(t) \in \mathcal{S}^j$, and $s_i^k(t) = Y(M(\Theta_i^k)|_{x_{0i}^k, u_i^k})(t)$, for $s_i^k(t) \in \mathcal{S}^k$, such that the corresponding sets $\mathcal{C}^j$ and $\mathcal{C}^k$ given in (7) are non intersecting, that is $\mathcal{C}^j \bigcap \mathcal{C}^k = \{\emptyset\}$, are said to be *well-posed* classes according to model M.

From this definition it follows immediately that well-posed classes of signals define movemes according to Definition 2.1. In practice we have access to a finite set of signals, $S^j = \{s_1^j(t), ..., s_{n_j}^j(t)\}$ and $S^k = \{s_1^k(t), ..., s_{n_k}^k(t)\}$, which belong to the two classes $\mathcal{S}^j$ and $\mathcal{S}^k$, with $s_i^j(t) = Y(M(\Theta_i^j)|_{x_{0i}^j, u_i^j})(t)$ for $i \in \{1, ..., n_j\}$ and $s_i^k(t) = Y(M(\Theta_i^k)|_{x_{0i}^k, u_i^k})(t)$ for $i \in \{1, ..., n_k\}$. Let $\hat{\mathcal{C}}^j$ and $\hat{\mathcal{C}}^k$ be the images, through f_M, of the sets S^j and S^k respectively. By construction we have $\hat{\mathcal{C}}^k \subset \mathcal{C}^k$ and $\hat{\mathcal{C}}^j \subset \mathcal{C}^j$, so that potentially we can have trivial intersection between $\hat{\mathcal{C}}^j$ and $\hat{\mathcal{C}}^k$, and a no-empty intersection between the sets $\mathcal{C}^k$ and $\mathcal{C}^j$. This creates a problem since if we check Definition 2.3 with $\hat{\mathcal{C}}^j$ and $\hat{\mathcal{C}}^k$, which are the only ones to which we have access, the classes of signals $\mathcal{S}^j$ and $\mathcal{S}^k$ turn out to be well-posed. The

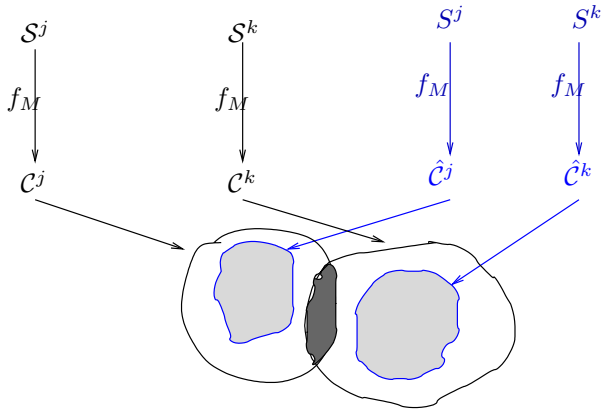

Figure 1: Relation between sets $\hat{\mathcal{C}}^j$ and $\hat{\mathcal{C}}^k$ and $\mathcal{C}^j$ and $\mathcal{C}^k$.

situation is depicted in Figure 1. The issue comes from the fact that we will use the light sets ($\hat{\mathcal{C}}^j$ and $\hat{\mathcal{C}}^k$) for solving the classification problem ignoring the existence of the dark region that is generating signals with undecidable class. Then, one needs to check if $\mathcal{S}^k$ and $\mathcal{S}^j$ are well-posed. The following lemma gives a possible way to check for well-posedness without knowing the sets $\mathcal{C}^j$ and $\mathcal{C}^k$.

Lemma 2.3. *Let $y(t) = Y(M(\Theta)|_{u,x_0})(t)$ denote the output of model M for a choice of Θ, u and x_0. Assume to fix u, x_0 and $\Theta|_{E-E'}$, so that $y(t) = Y(M(\theta))(t)$, and let the classes $\mathcal{S}^j$ and $\mathcal{S}^k$ be defined as*

$$\mathcal{S}^j = \{y(t) | y(t) = Y(M(\theta)) \text{ and } g_j(y, \dot{y}, t) = 0,\ h_j(y, \dot{y}, t) \leq 0\}$$
$$\mathcal{S}^k = \{y(t) | y(t) = Y(M(\theta)) \text{ and } g_k(y, \dot{y}, t) = 0,\ h_k(y, \dot{y}, t) \leq 0\}$$

for some functions g_j, g_k, h_j and h_k. Then the classes of signals $\mathcal{S}^j$ and $\mathcal{S}^k$ are well-posed if and only if the system

$$\begin{aligned} y(t) &= Y(M(\theta))(t) \\ g_j(y, \dot{y}, t) &= 0 \\ h_j(y, \dot{y}, t) &\leq 0 \\ g_k(y, \dot{y}, t) &= 0 \\ h_k(y, \dot{y}, t) &\leq 0 \end{aligned} \tag{8}$$

is infeasible.

Proof. ($\Rightarrow$). Let us show that well-posed classes $\mathcal{S}^j$ and $\mathcal{S}^k$ imply infeasibility of (8). According to Definition 2.3 this is equivalent to showing that non-intersecting sets $\mathcal{C}^j$ and $\mathcal{C}^k$ (defined in equation (7)) imply infeasibility of the system of equations (8). Let again $\mathcal{F}_M$ be the one to one mapping between the signal $Y(M(\Theta)|_{x_0,u})(t)$ and the couple (Θ, u) which exists by virtue of (i) of Definition 2.1, and

since input u, initial condition x_0 and $\Theta|_{E-E'}$ have been fixed, $\mathcal{F}_M$ becomes one to one correspondence between $y(t) = Y(M(\theta))(t)$ and θ. Then we can redefine the sets $\mathcal{C}^j$ and $\mathcal{C}^k$ as

$$\mathcal{C}^j = \{\theta | \theta = \mathcal{F}_M(y(t)), \text{ and } y(t) \in \mathcal{S}^j\} \tag{9}$$

and

$$\mathcal{C}^k = \{\theta | \theta = \mathcal{F}_M(y(t)), \text{ and } y(t) \in \mathcal{S}^k \ .\} \tag{10}$$

If (8) is feasible then there exist $y(t)$ such that $y(t) \in \mathcal{S}^j$ and $y(t) \in \mathcal{S}^k$ and also there exist $\theta^* : \ y(t) = M(\theta^*)(t)$, so that by (9) and (10) $\theta^* \in \mathcal{C}^j$ and $\theta^* \in \mathcal{C}^k$, which in turn implies $\mathcal{C}^k \bigcap \mathcal{C}^j \neq \{\emptyset\}$. Then we have shown that trivial intersection of sets $\mathcal{C}^j$ and $\mathcal{C}^k$ defined in (9) and (10) implies infeasibility of (8). ($\Leftarrow$). Let us show now that if classes $\mathcal{S}^j$ and $\mathcal{S}^k$ are not well-posed, then system (8) is feasible. By Definition 2.3, this is equivalent to show that if $\mathcal{C}^k \bigcap \mathcal{C}^j \neq \{\emptyset\}$ then system (8) is feasible. $\mathcal{C}^k \bigcap \mathcal{C}^j \neq \{\emptyset\}$ implies that there exist $\theta^* \in \mathcal{C}^j$ and $\theta^* \in \mathcal{C}^k$ which from (9) and (10) implies that there exist a signal $y^*(t)$ such that $\theta^* = \mathcal{F}_M(y^*(t)), \ y^*(t) \in \mathcal{S}^j$ and $\theta^* = \mathcal{F}_M(y^*(t)), \ y^*(t) \in \mathcal{S}^k$, which means that the signal $y^*(t)$ is both in $\mathcal{S}^j$ and $\mathcal{S}^k$ which implies that it satisfies (8), then (8) is feasible. This completes the proof. □

We introduced Definition 2.3 for practical reasons and its use will be shown in the experiment section (Section 5) with an example. In this section we have introduced the notion of moveme, the definition of segmentability, the classification problem with a possible solution, and the operative definition of well-posedness. In the following section we consider a segmentable signal corrupted by noise: in the noiseless case we saw that the segmentation is unique; under suitable assumptions on noise (such as bounded white noise) we can assume that the segmentation is still unique. Then we introduce the segmentation problem and propose a solution.

3 Problem Statement

Consider the sequence of systems for $i = 0, ..., l$

$$\begin{cases} \dot{x} = (A_i + \delta U_i)x + b_i + d(t) & t \in [\tau_{i-1}, \tau_i) \\ \dot{x} = (A_{i+1} + \delta U_{i+1})x + b_{i+1} + d(t) & t \in (\tau_i, \tau_{i+1}] \end{cases} \tag{11}$$

with $x \in \mathbb{R}^n$, $A_i \in \mathbb{R}^{n \times n}$ an unknown matrix whose value can take place in the set of known Hurwitz matrices $\{A_c^1, ..., A_c^m\}$, which are centers of the sets defined in (2), i.e. $\mathcal{C}^j = B_{r_j}(A_c^j)$ with $\mathcal{C}^j \bigcap \mathcal{C}^k = \{\emptyset\}$ for $j \neq k$, $b_i \in \mathbb{R}^n$ unknown constant vectors, $U_i \in \mathbb{R}^{n \times n}$ norm one matrices (according to Frobenius norm), $\delta \in \mathbb{R}$ modeling uncertainty with $|\delta| \leq \overline{\delta}$, $d(t)$ realization of white noise such that $\|d(t)\| \leq \overline{d}$, τ_i unknown switching times with τ_0 known starting time and τ_l known ending time.

Consider also the related nominal system obtained letting noise and parameter uncertainty to zero:

$$\begin{cases} \dot{x} = A_i x + b_i & t \in [\tau_{i-1}, \tau_i) \\ \dot{x} = A_{i+1} x + b_{i+1} & t \in (\tau_i, \tau_{i+1}] \end{cases} \tag{12}$$

with interconnection condition

$$\frac{\dot{x}(\tau_i^-)^T \dot{x}(\tau_i^+)}{\|\dot{x}(\tau_i^-)\| \|\dot{x}(\tau_i^+)\|} \leq \rho_0 < 1 \ . \tag{13}$$

where we define

$$\dot{x}(\tau_i^-) = \lim_{\tau \to \tau_i^-} \dot{x}(\tau)$$
$$\dot{x}(\tau_i^+) = \lim_{\tau \to \tau_i^+} \dot{x}(\tau)$$

The interconnection condition gives a bound on the discontinuity in the trajectory's derivative at the switching points, which will be required to detect the end of one segment and the beginning of another.

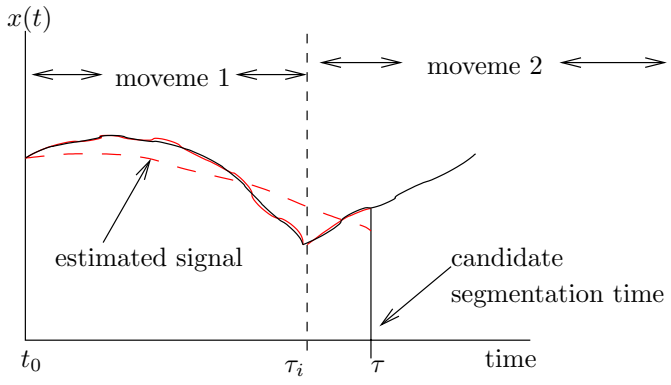

Figure 2: Signal considered for computation of approximation and parametric errors (one component shown) and estimated signal $\hat{x}$ (dashed line).

We wish to obtain sufficient conditions on noise level and parameter uncertainty that allow off-line determination of the sequence of times $\{\tau_1, ..., \tau_{l-1}\}$ and the sequence of matrices $\{A_1, ..., A_l\}$ from the observation of state x. If we have a good guess of the switching times, then we can apply Lemma 2.2 so to solve the classification problem in each interval between two switching times.

We thus focus our attention on the segmentation part of the problem, that is to determine a good estimate for the sequence of times $\{\tau_1, ..., \tau_{l-1}\}$. We use an iterative approach in which at each iteration we look for the maximizer of a function defined on $[t_0, t_M]$ where $t_M = \tau_l$ and t_0 is a starting time which coincides with τ_0 at the first iteration. We want to show the maximizer of such function falls in an interval I around the first switching time encountered after time t_0; moreover this interval should shrink down to the switching point when noise and parameter uncertainty go to zero. To define such a function we define three quantities for system (11): the approximation error, the parametric error, and the transition factor at time τ, where such quantities are computed based on the observation of $x(t)$ of system (11) for $t \in [t_0, \tau]$, $\tau \in (t_0, t_M)$. We begin by computing the least squares estimate for $x(t)$, $t \in [t_0, \tau]$:

$$(\hat{A}|\hat{b}) = \left[\int_{t_0}^{\tau} \dot{x}\, \overline{x}^T dt\right] \left[\int_{t_0}^{\tau} \overline{x}\, \overline{x}^T dt\right]^{-1}, \tag{14}$$

where $\overline{x} := (x^T, 1)^T$. Assuming first that $t_0 = \tau_{i-1}$, the estimate becomes

$$(\hat{A}|\hat{b})(\tau, t_0) = \begin{cases} \left[\int_{t_0}^{\tau_i} (A_i + \delta U_i|b_i)\overline{x}\, \overline{x}^T dt + \right. \\ \qquad \int_{\tau_i}^{\tau} (A_{i+1} + \delta U_{i+1}|b_{i+1})\overline{x}\, \overline{x}^T dt + \\ \qquad \left. \int_{t_0}^{\tau} d(t)\overline{x}^T dt\right] \left[\int_{t_0}^{\tau} \overline{x}\, \overline{x}^T dt\right]^{-1} & \tau_i < \tau < \tau_{i+1} \\ \left[\int_{t_0}^{\tau} (A_i + \delta U_i|b_i)\overline{x}\, \overline{x}^T dt + \right. \\ \qquad \left. \int_{t_0}^{\tau} d(t)\overline{x}^T dt\right] \left[\int_{t_0}^{\tau} \overline{x}\, \overline{x}^T dt\right]^{-1} & \tau < \tau_i \ . \end{cases} \tag{15}$$

The estimates given by (15) generate the system

$$\dot{\hat{x}} = \hat{A}(\tau, t_0)\hat{x} + \hat{b}(\tau, t_0) \ , \qquad \hat{x}(t_0) = x(0). \tag{16}$$

This situation is depicted in Figure 2, where we report the candidate segmentation time τ, the switching time τ_i, the portion of signal under study composed by the sequence of two movemes (solid line), the estimated trajectory (dashed line) obtained by system (16). We define the parametric error at time τ as

$$e_p(\tau, t_0) = \min_{j=1,\dots,m} \|\hat{A}(\tau, t_0) - A_c^j\| \ , \tag{17}$$

where we consider the Frobenius norm to be the matrix norm. The approximation error at time τ is

$$e_a(\tau, t_0) = \frac{1}{\tau - t_0} \int_{t_0}^{\tau} (x - \hat{x})^T (x - \hat{x}) dt \ , \tag{18}$$

and the transition factor is defined as

$$\mathrm{Tr}(\tau) = \tfrac{1}{2} \left(1 - \frac{\dot{x}_{av}(\tau^-)^T \dot{x}_{av}(\tau^+)}{\|\dot{x}_{av}(\tau^-)\| \|\dot{x}_{av}(\tau^+)\|} \right) \tag{19}$$
$$\dot{x}_{av}(\tau^-) := \tfrac{1}{\Delta\tau} \int_{\tau-\Delta\tau}^{\tau} \dot{x}(t) dt, \qquad \dot{x}_{av}(\tau^+) := \tfrac{1}{\Delta\tau} \int_{\tau}^{\tau+\Delta\tau} \dot{x}(t) dt$$

which takes care of local properties of the signal $x(t)$. $\Delta\tau$ is a positive constant depending on perturbation level which will be determined later.

For each time τ we have different values of these three quantities and the idea is to combine them in one function of τ which has the maximizer close to the switching point. The function we choose is

$$W(\tau, t_0) = \frac{\exp(\frac{-e_p(\tau,t_0)^2}{\sigma^2}) \mathrm{Tr}(\tau)}{a + e_a(\tau, t_0)}, \qquad \tau \in (t_0, t_M] \tag{20}$$

where a is an arbitrarily small positive constant to prevent the denominator from being zero. By maximizing function $W(\tau)$ we are looking for the value of τ which has small approximation error (which implies a good guess of dynamical parameters), small parametric error, and a high transition factor. Expression (19) involves integration over time $\Delta\tau$ to attenuate the effect of noise and its expression for system (12) is obtained by letting $\Delta\tau \to 0$. We find that $\mathrm{Tr}(\tau_i) \geq (1 - \rho_0)/2$ and for $\tau \neq \tau_i$, $\mathrm{Tr}(\tau) = 0$. The idea of the transition factor term is to preserve this property as much as possible in the perturbed case so that all the times $\tau_i + \Delta\tau \leq \tau \geq \tau_{i+1} - \Delta\tau$ and $t_0 < \tau \leq \tau_i - \Delta\tau$ are penalized with respect to time τ_i. We also choose to minimize $e_p(\tau, t_0)$ so to reduce the effect of perturbation on the parameter estimates. Alternatively, one could constrain the estimates $\hat{A}$ to lie in a ball around A_i, but we do not know the value of A_i *a priori*, we just know that it belongs to a set of possible values. Therefore we decide to minimize the distance of $\hat{A}$ from the closest point A_j at time τ according to a Gaussian metric.

4 Main Result

Consider the sequence of dynamical systems, for $t \in [\tau_0, \tau_l]$, switching at unknown times $\{\tau_1, ..., \tau_{l-1}\}$ defined as

$$\begin{cases} \dot{x} = (A_i + \delta U_i)x + b_i + d(t) & t \in [\tau_{i-1}, \tau_i) \\ \dot{x} = (A_{i+1} + \delta U_{i+1})x + b_{i+1} + d(t) & t \in (\tau_i, \tau_{i+1}] \end{cases} \tag{21}$$

with $x \in \mathbb{R}^n$, the matrices $A_i \in \mathbb{R}^{n \times n}$ are unknown matrices, $U_i \in \mathbb{R}^{n \times n}$ are unit norm matrices, and the vectors $b_i \in \mathbb{R}^n$ are unknown constant vectors. We make a number of assumptions on the nominal system and its perturbation:

Assumption 4.1. The i^{th} segment is of class j, with j unknown. In formulas we have $(A_i + \delta U_i) \in \mathcal{C}^j$, and $A_i = A_c^j \in \{A_c^1, ..., A_c^m\}$ for some j and the set of known Hurwitz matrices, $\{A_c^1, ..., A_c^m\}$, is such that $\mathcal{C}^j = B_{r_j}(A_c^j)$ with $\mathcal{C}^j \bigcap \mathcal{C}^k = \{\emptyset\}$ for $j \neq k$. (Figure 3 gives an intuitive idea of the sets $\mathcal{C}^j$ and $\mathcal{C}^k$.)

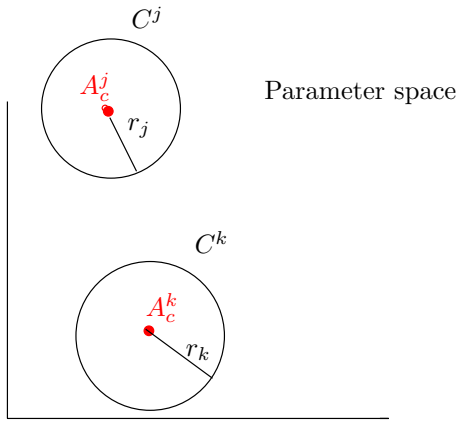

Figure 3: Intuitive picture in 2D space of sets $\mathcal{C}^j$ and $\mathcal{C}^k$.

Assumption 4.2. $\delta \in \mathbb{R}$ represents modeling uncertainty with $|\delta| \leq \overline{\delta}$, and $d(t)$ is realization of white noise such that $\|d(t)\| \leq \overline{d}$.

Assumption 4.3. The nominal system

$$\begin{cases} \dot{x} = A_i x + b_i & t \in [\tau_{i-1}, \tau_i) \\ \dot{x} = A_{i+1} x + b_{i+1} & t \in (\tau_i, \tau_{i+1}] \end{cases}$$

satisfies the *interconnection condition*

$$\frac{\dot{x}(\tau_i^-)^T \dot{x}(\tau_i^+)}{\|\dot{x}(\tau_i^-)\| \|\dot{x}(\tau_i^+)\|} \leq \rho_0 < 1 \ .$$

Assumption 4.4. The state $x(t)$ of the nominal system is such that

$$v^T \overline{x}(t) = 0, \ t \in [t_1, t_2], \ t_2 > t_1 \quad \Longrightarrow \quad v = 0 \ .$$

where $\overline{x} = (x^T, 1)^T$.

We give the following theorem.

Theorem 4.1. *Consider the sequence of dynamical systems given in (21) subject to Assumptions 4.1 to 4.4. Let the function $W(\tau, t_0)$ be defined as*

$$W(\tau, t_0) = \frac{\exp(\frac{-e_p(\tau,t_0)^2}{\sigma^2}) \mathrm{Tr}(\tau)}{a + e_a(\tau, t_0)}, \qquad \tau \in (t_0, t_M]$$

for $t_0 = \tau_{i-1}$ and $t_M = \tau_l$. Then there exist bounds δ^* and d^* such that if $\overline{\delta} \leq \delta^*$ and $\overline{d} \leq d^*$ the potential function $W(\tau)$ admits its global maximizer $\hat{\tau}_i$ for $\hat{\tau}_i \in I = [\tau_i - \Delta\tau, \tau_i + \Delta\tau^+]$ where I contracts to τ_i as $\overline{\delta} \to 0$ and $\overline{d} \to 0$. Moreover the estimated class $\hat{j}$ of the segment in $[t_0, \hat{\tau}_i]$ is equal the class of i^{th} segment generated by system (21).

Note that the theorem states that we can find j which means that we can determine A_c^j and therefore A_i. Before proving the theorem we give some intermediate results whose proofs are reported in the appendix. In what follows we omit the dependence on t_0.

4.1 Intermediate results and proof of the theorem

To prove Theorem 4.1 we make use of a sequence of lemmas, whose proofs are given in the appendix.

Lemma 4.1. *Consider expression (15), with $\delta = 0$ and $d(t) = 0$ for all t, for system (12). There exists $k_1 > 0$ such that*

$$\|(\hat{A}|\hat{b}) - (A_i|b_i)\|^2 \geq k_1(\tau - \tau_i)^2 \ , \qquad \tau_i < \tau < \tau_{i+1} \tag{22}$$

This lemma establishes that the closer τ is to the switching time τ_i the smaller the lower bound of the parameter estimation error for the nominal system.

Lemma 4.2. *Consider system (12) with estimates (15) with $\delta = 0$ and $d(t) = 0$ for all t. For the estimated states generated by system (16) and $e_a(\tau)$ as defined in (18), there exists $k_2 > 0$ such that*

$$e_a(\tau) \geq k_2\|(\hat{A}|\hat{b}) - (A_i|b_i)\|^2 \ , \qquad \tau_i < \tau < \tau_{i+1} \tag{23}$$

This lemma states that for the nominal system (12) the approximation error lower bound increases as the parameter estimates become far from the parameters A_i, b_i. These two lemmas hold for the quantities computed for the nominal system. In order to link such quantities with the ones computed for the perturbed system (21), we give two additional results: the first one establishes how far two system's states are from each other when the two systems differ because of parameter differences and the presence of noise, the second result explicitly links parameter and approximation errors for nominal and perturbed systems through the aid of the first result.

Lemma 4.3. *Let A and A_1 be Hurwitz matrices and consider the pair of systems*

$$\dot{x} = Ax + b \tag{24}$$

$$\dot{z} = A_1 z + b_1 + d(t) \tag{25}$$

with x and z in $\mathbb{R}^n$, $A,\ A_1 \in \mathbb{R}^{n\times n}$, b and b_1 constant vectors in $\mathbb{R}^n$, $\|d(t)\| \leq \overline{d}$ and $\|(A|b)-(A_1|b_1)\| \leq \overline{\delta}$. Then if $x(0) = z(0)$ there exist $k_3 > 0$ and $k_4 > 0$ such that

$$\|x - z\|^2 \leq k_3\overline{\delta} + k_4\overline{d} \qquad \forall t \geq 0 \tag{26}$$

Lemma 4.4. *Let $e_p(\tau)$ and $e_a(\tau)$ denote parametric errors and approximation errors given in expressions (17) and (18) for the sequence of dynamical systems (21). Let $e_p^0(\tau)$ and $e_a^0(\tau)$ denote parametric errors and approximation errors for the related nominal system (12). Then there exist constants $k_p > 0$ and $k_a > 0$ such that*

$$e_p^0(\tau) - \Delta \leq e_p(\tau) \leq e_p^0(\tau) + \Delta \tag{27}$$

$$e_a^0(\tau) - \varepsilon \leq e_a(\tau) \leq e_a^0(\tau) + \varepsilon \tag{28}$$

with $\Delta = k_p(\overline{d} + \overline{d}^2 + \overline{d}^3 + \overline{\delta} + \overline{\delta}^2 + \overline{\delta}^3)$ and $\varepsilon = k_a(\overline{d} + \overline{d}^2 + \overline{d}^3 + \overline{d}^4 + \overline{\delta} + \overline{\delta}^2 + \overline{\delta}^3 + \overline{\delta}^4 + \overline{\delta}^6)$.

Let us consider now the transition factor given in expression (19) for system (21). Our aim is to find a possible value of the averaging time $\Delta\tau$ as function of noise level and parameter uncertainty such that for $\tau_i + \Delta\tau \leq \tau \leq \tau_{i+1} - \Delta\tau$ for each i the transition factor becomes smaller and smaller as the perturbation decreases and reaches zero when we have no perturbation at all.

Lemma 4.5. *Let the transition factor be given by (19) for system (11). There exist positive constants c_1 and c_2 such that if*

$$\Delta\tau = -c_1 \ln\left(\frac{1-2\beta}{1-\beta}\right) \tag{29}$$

then the transition factor is such that

$$\mathrm{Tr}(\tau) \leq c_2\beta \qquad \tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau, \tag{30}$$

$$\mathrm{Tr}(\tau) \geq \frac{1-\rho_0-\varphi}{2} \qquad \tau = \tau_i, \tag{31}$$

for all i, where β and φ are perturbation dependent quantities and go to zero as the perturbation goes to zero.

Proof of Theorem 4.1. The proof proceeds in three steps: we show that the function W given in (20) achieves smaller values in $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau$ than the one at $\tau = \tau_i$. Then we show that such value is larger also than the one that W achieves at times $\tau_i + \Delta\tau \leq \tau < \tau_{i+1} - \Delta\tau$ and, finally, at times $\tau > \tau_{i+1} - \Delta\tau$. Let us first show that $W(\tau_i) > W(\tau)$ for $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau$. In fact

$$W(\tau_i) = \frac{\exp(\frac{-e_p(\tau_i)^2}{\sigma^2})\mathrm{Tr}(\tau_i)}{a + e_a(\tau_i)} \geq \frac{\exp(\frac{-\Delta^2}{\sigma^2})(1-\rho_0-\varphi)/2}{a+\varepsilon}$$

by virtue of Lemma 4.5, and Lemma 4.4 and by the fact that $e_a^0 = 0$ and $e_p^0 = 0$ for $\tau \leq \tau_i$. For $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau$ we have

$$W(\tau) = \frac{\exp(\frac{-e_p(\tau)^2}{\sigma^2})\mathrm{Tr}(\tau)}{a + e_a(\tau)} \leq \frac{\mathrm{Tr}(\tau)}{a} \leq \frac{c_2\beta}{a}$$

by virtue of Lemma 4.5, and Lemma 4.4 again and by the fact that $e_a^0 = 0$ and $e_p^0 = 0$ for $\tau \leq \tau_i$. Therefore in order to show that $W(\tau_i) > W(\tau)$ for $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau$, it suffices to show

$$\frac{\exp(\frac{-\Delta^2}{\sigma^2})(1-\rho_0-\varphi)/2}{a+\varepsilon} > \frac{c_2\beta}{a},$$

which is the same as requiring

$$\frac{2c_2\beta}{a} e^{\Delta^2/\sigma^2}(a+\varepsilon) + \varphi \leq 1 - \rho_0 \ . \tag{32}$$

Inequality (32) imposes conditions on the perturbation amplitude once ρ_0 has been fixed. In fact in the perturbed case if the noise and parameter uncertainties are too big even if in the nominal case the local signal properties would clearly give evidence of a transition, the corrupted signal could not maintain such local properties that would be hidden by perturbation.

Consider now the case $\tau_i + \Delta\tau \leq \tau \leq \tau_{i+1} - \Delta\tau$. For such times, from Lemma 4.1 and 4.2, we have $e_a^0(\tau) \geq k(\tau - \tau_i)^2$ for a suitable $k > 0$, then plugging this relation into the left side of (28) we obtain $e_a(\tau) \geq k(\tau-\tau_i)^2 - \varepsilon$. Using this relation in the expression of $W(\tau)$ we have

$$W(\tau) \leq \frac{\mathrm{Tr}(\tau)}{a + k(\tau-\tau_i)^2 - \varepsilon} \leq \frac{c_2\beta}{a + k(\tau-\tau_i)^2 - \varepsilon}$$

where we have used (30). Since also

$$W(\tau_i) \geq \frac{\exp(\frac{-\Delta^2}{\sigma^2})(1-\rho_0-\varphi)/2}{a+\varepsilon}$$

if we require

$$\frac{\exp(\frac{-\Delta^2}{\sigma^2})(1-\rho_0-\varphi)/2}{a+\varepsilon} > \frac{c_2\beta}{a+k(\tau-\tau_i)^2-\varepsilon},$$

we find

$$(\tau-\tau_i)^2 > \max\{\Delta\tau^2, \frac{\varepsilon-a}{k} + \frac{2c_2\beta(\varepsilon+a)e^{\Delta^2/2}}{k(1-\rho_0-\varphi)}\} := (\Delta\tau^+)^2 \ . \tag{33}$$

For such times $W(\tau)$ cannot have a maximizer, therefore the maximizer can occur only for $(\tau-\tau_i) \leq \Delta\tau^+$ which tends to zero as the perturbation tends to zero because when $\varepsilon \to 0$ and $\delta \to 0$, also $\beta \to 0$ and $\Delta\tau \to 0$ by Lemma 4.5, so that $(\Delta\tau^+)^2 = \max\{0, -a/k\} = 0$. In presence of perturbation, $(\Delta\tau^+)$ gives a measure of the uncertainty on τ_i for $\tau > \tau_i$, the left uncertainty is determined by $\Delta\tau$ only.

We finally show that $W(\tau_i) > W(\tau)$ for $\tau \geq \tau_{i+1} - \Delta\tau$. By definition of e_a from (18) we have $e_a(\tau) = \frac{1}{\tau-t_0}\int_{t_0}^{\tau}(\tilde{x})^T(\tilde{x})dt$, where $\tilde{x} = x - \hat{x}$ and $\hat{x}$ is generated by system (16). Then we can rewrite e_a as $e_a(\tau) = \frac{1}{\tau-t_0}\left(\int_{t_0}^{\tau_i}(\tilde{x})^T(\tilde{x})dt + \int_{\tau_i}^{\tau_i+1}(\tilde{x})^T(\tilde{x})dt + ... + \int_{\tau_{i+m}}^{\tau}(\tilde{x})^T(\tilde{x})dt\right)$, and by applying Lemma 4.2 to each integral we have for suitable h_i's, $e_a(\tau) \geq h_1\|(A_i,b_i)-(\hat{A},\hat{b})\|^2 + h_2\|(A_{i+1},b_{i+1})-(\hat{A},\hat{b})\|^2 + ... + h_{m+1}\|(A_{i+m},b_{i+m})-(\hat{A},\hat{b})\|^2 \geq h[\|(A_i,b_i)-(\hat{A},\hat{b})\|^2 + \|(A_{i+1},b_{i+1})-(\hat{A},\hat{b})\|^2 + ... + \|(A_{i+m},b_{i+m})-(\hat{A},\hat{b})\|^2]$, with h the smallest h_i. The term in square brackets has a minimum for $(\hat{A},\hat{b}) = (A^*,b^*)_{m+1} := \frac{(A_i,b_i)+...+(A_{i+m},b_{i+m})}{m+1}$ which is the barycenter of the distribution of $m+1$ points $\{(A_i,b_i),...,(A_{i+m},b_{i+m})\}$. Then we have

$$\begin{aligned}
&\|(A_i|b_i)-(A^*|b^*)_2\|^2 + \|(A_{i+1}|b_{i+1})-(A^*|b^*)_2\|^2 \\
&\leq \|(A_i|b_i)-(A^*|b^*)_{m+1}\|^2 + \|(A_{i+1}|b_{i+1})-(A^*|b^*)_{m+1}\|^2 \\
&\leq \|(A_i|b_i)-(A^*|b^*)_{m+1}\|^2 + ... + \|(A_{i+m}|b_{i+m})-(A^*|b^*)_{m+1}\|^2 \\
&\leq \|(A_i|b_i)-(\hat{A}|\hat{b})\|^2 + \|(A_{i+1}|b_{i+1})-(\hat{A}|\hat{b})\|^2 + ... + \|(A_{i+m}|b_{i+m})-(\hat{A}|\hat{b})\|^2
\end{aligned}$$

for any $(\hat{A}|\hat{b})$. Since also $\|(A_i|b_i)-(A^*|b^*)_2\|^2 + \|(A_{i+1}|b_{i+1})-(A^*|b^*)_2\|^2 = \frac{\|(A_i|b_i)-(A_{i+1}|b_{i+1})\|^2}{2}$, it follows that

$$e_a(\tau) \geq h\frac{\|(A_i|b_i)-(A_{i+1}|b_{i+1})\|^2}{2} \ .$$

Assuming that $\|(A_i|b_i)-(A_{i+1}|b_{i+1})\| \geq \Delta^c_{min}$ for all i, we have $e_a(\tau) \geq h\frac{(\Delta^c_{min})^2}{2}$. Then

$$W(\tau) \leq \frac{1}{a+h\frac{(\Delta^c_{min})^2}{2}-\varepsilon} \ ,$$

where in place of $\mathrm{Tr}(\tau)$ and of $\exp(-e_p(\tau)^2/\sigma^2)$ we have substituted one that is the maximum possible value they can take. Then since

$$W(\tau_i) \geq \frac{\exp(\frac{-\Delta^2}{\sigma^2})(1-\rho_0-\varphi)/2}{a+\varepsilon},$$

it is sufficient that

$$\frac{1}{a+h\frac{(\Delta^c_{min})^2}{2}-\varepsilon} < \frac{\exp(\frac{-\Delta^2}{\sigma^2})(1-\rho_0-\varphi)/2}{a+\varepsilon},$$

which implies

$$(\Delta^c_{min})^2 > \frac{4e^{\Delta^2/\sigma^2}(\varepsilon+a)}{(1-\rho_0-\varphi)h} + \frac{2(\varepsilon-a)}{h} \tag{34}$$

This requirement asks that the minimum distance between parameters generating adjacent segments in the sequence of dynamical systems (11) has to increase if the perturbation due to parameter uncertainty and noise increases. In fact when τ increases after τ_i and multiple segments are included in interval

(t_0, τ) the approximation error increases with respect to the one we have at $\tau = \tau_i$ where the only contribution is due to noise. If such increase is comparable with the contribution of noise that we have at $\tau = \tau_i$, then it becomes harder to say if we are including new segments in (t_0, τ) when τ increases. A way to prevent this is therefore to ask that the contribution to $e_a(\tau)$ when τ increases and new segments are included in (t_0, τ) is bigger than the one due to noise. This is guaranteed by a sufficiently big distance between $(A_i|b_i)$ and $(A_{i+1}|b_{i+1})$ for each i as expression (34) states. Also it is possible to show that constant k in expression (33) is proportional to (Δ^c_{min}) which means that for higher values of separations between points $(A_i|b_i)$ and $(A_{i+1}|b_{i+1})$ we are able to include a smaller portion of the segment starting at τ_i before realizing that a switch has occurred.

What we have shown up to this point is that $W(\tau)$ for $\tau \in [t_0 + \Delta\tau, \tau_l]$ has the global maximizer falling into $I = [\tau_i - \Delta\tau, \tau_i + \Delta\tau^+]$ if the noise level and parameter uncertainty are, for a given value of ρ_0 and (Δ^c_{min}), such that conditions (33) and (34) are verified and if we take for $\Delta\tau$ the value specified in (29). To have the same result hold for $\tau \in (t_0, \tau_l]$ we need to assume that there is no switching point in $(t_0, t_0 + \Delta\tau)$, which is certainly true if we ask that $\Delta\tau < \frac{1}{2}\min_i(\tau_i - \tau_{i-1})$. This condition has been implicitly assumed in order to be able to compute $\mathrm{Tr}(\tau)$ in the interior of segments between switching points τ_{i-1} and τ_i, therefore we can formalize this by letting $T = \min_i(\tau_i - \tau_{i-1})$ denote the shortest duration of a segment, and asking $\Delta\tau = -c_1 \ln\left(\frac{1-2\beta}{1-\beta}\right) < \frac{T}{2}$, which leads to

$$\beta < \frac{1 - e^{-T/(2c_1)}}{2 - e^{T/(2c_1)}}. \tag{35}$$

To complete the proof we need to show that we can determine A_i exactly. This is a classification problem of the same kind as discussed in Section 2.2. In fact by Assumption 4.1, $A_i = A^j_c$ for some j and $\mathcal{C}^j = B_{r_j}(A^j_c)$ with $C^j \bigcap C^k = \{\emptyset\}$ for $j \neq k$. This means that it is sufficient to have an estimation $\hat{A}$ of A_i which lies in $\mathcal{C}^j$, so that $\|\hat{A} - A^k_c\| \leq r_k$ is satisfied, by the non intersection property of the sets $\mathcal{C}^k$, only for $k = j$ which means $A^k_c = A^j_c = A_i$. Then given $\hat{\tau}_i$ we can compute an estimate for A_i by (15) as:

$$(\hat{A}|\hat{b}) = \left[\int_{t_0}^{\hat{\tau}_i} (A_i + \delta U_i|b_i)\overline{x}\,\overline{x}^T dt + d(t)\overline{x}^T dt\right]\left[\int_{t_0}^{\hat{\tau}_i} \overline{x}\,\overline{x}^T dt\right]^{-1}$$

and subtracting from both sides $(A_i|b_i)$ we find

$$(\hat{A}|\hat{b}) - (A_i|b_i) = \left[\int_{t_0}^{\tau_i} (\delta U_i|0)\overline{x}\,\overline{x}^T dt + \int_{\tau_i}^{\hat{\tau}_i} ((A_{i+1}|b_{i+1}) - (A_i|b_i))\,\overline{x}\,\overline{x}^T dt + \int_{t_0}^{\hat{\tau}_i} d(t)\overline{x}^T dt + \int_{\tau_i}^{\hat{\tau}_i} (\delta U_{i+1}|0)\overline{x}^T dt\right]\left[\int_{t_0}^{\hat{\tau}_i} \overline{x}\,\overline{x}^T dt\right]^{-1}$$

therefore in the worst case scenario where $\hat{\tau}_i > \tau_i$ we have $\|\hat{A} - A_i\| \leq \|(\hat{A}|\hat{b}) - (A_i|b_i)\| \leq k_1\overline{d} + k_2\overline{\delta} + (\hat{\tau}_i - \tau_i)k_3 \leq k_1\overline{d} + k_2\overline{\delta} + (\Delta\tau^+)k_3$ for suitable positive constants k_1, k_2, k_3. Since $A_i = A^j_c$, to obtain $\|\hat{A} - A_i\| = \|\hat{A} - A^j_c\| \leq r_j$ it is sufficient that

$$k_1\overline{d} + k_2\overline{\delta} + (\Delta\tau^+)k_3 \leq r_j \tag{36}$$

where $\Delta\tau^+$ is defined in (33), so that the function $C(\hat{\tau}_i, k) := \|\hat{A} - A^k_c\| - r_k$ is less or equal than zero if and only if $k = j$, and therefore we have found $A^j_c = A_i$. Therefore by (32), (34), (35), (36) and Lemma 4.5, and recalling that $\varphi = \rho_0\alpha + \alpha + \rho_0\beta + \rho_0\alpha\beta + \alpha\beta + \alpha$ and that $\beta = \beta(\overline{d}, \overline{\delta})$ and that $\alpha = \alpha(\beta) = \alpha(\overline{d}, \overline{\delta})$ we can derive conditions on the maximum allowed values for $\overline{d}$ and $\overline{\delta}$. Let d^* and δ^* be such bounds. □

Theorem 4.1 has been proved assuming $W(\tau)$ to be defined on (t_0, t_M) where $t_0 = \tau_{i-1}$ and $t_M = \tau_l$ is the duration of the process (21). The assumption that $t_0 = \tau_{i-1}$ is valid only at the first iteration

in which $t_0 = \tau_0$. Then we find the maximizer $\hat{\tau}_1$ of $W(\tau)$ for $\tau \in (\tau_0, \tau_l)$ which lies in an interval $I = [\tau_1 - \Delta\tau, \tau_1 + \Delta\tau^+]$ around τ_1 and is an estimate of the first switching time τ_1. Then we have to set t_0 for the second iteration so that the first switching point encountered after t_0 is τ_2. In order to do this we set $t_0 = \hat{\tau}_1 + \Delta\tau$ so that we make sure that the first switching time encountered is τ_2 and not τ_1 again. In fact if the maximization process of W takes place with $t_0 > \tau_i$ and in the worst case scenario with $t_0 = \tau_i + \Delta\tau + \Delta\tau^+$ nothing changes as long as $T - (\Delta\tau + \Delta\tau^+) > 2\Delta\tau$ which by (29) and (33) implies an other condition on the noise level, which added to the ones found in Theorem 4.1 give new values for d^* and δ^*.

Remark 4.1. The result of Theorem 4.1 can be generalized to the case in which matrices A_j in system (1) have block diagonal form, i.e., $A_i = \text{blockdiag } (A_i^1, ..., A_i^L)$, $b = (b_1^T, ..., b_L^T)^T$, $d(t) = (d^1(t)^T, ..., d^L(t)^T)^T$, with $A_i^l \in \mathbb{R}^{p\times p}$, $L\,p = n$ and matrices A_i taking values in a finite set of matrices $\{A_1, ..., A_m\}$ having the same block diagonal structure. In such a case partitioning the state space into L independent subspaces $(x_1, ..., x_L)$ and assuming that for each one of the subspaces the matrix $[\int_{t_0}^t \overline{x}_l\ \overline{x}_l^T dt]^{-1}$ is non singular for $t > t_0$ we can compute the parameter estimate for each one of the blocks as

$$(\hat{A}_i^l, \hat{b}_i^l) = \int_{t_0}^{\tau} \dot{x}_l \overline{x}_l^T dt [\int_{t_0}^t \overline{x}_l\ \overline{x}_l^T dt]^{-1} \tag{37}$$

which is equal to expressions (15) for each one of the subsystems. In expression (16) we have $\hat{A} = \text{diag}(\hat{A}^1, ..., \hat{A}^L)$ and $\hat{b} = (\hat{b}^1, ..., \hat{b}^L)$. The parametric error (17) has the same definition where $\hat{A}$ and A_j have block diagonal structure. Then everything else holds at the same way. In particular if each block has the structure

$$A_i^l = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & \cdots & \cdots & \cdots & 1 \\ a_{l1} & a_{l2} & \cdots & \cdots & a_{lp} \end{pmatrix},\ b_i^l = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ b_{lp} \end{pmatrix} \qquad d^l(t) = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ d_{lp}(t) \end{pmatrix}$$

we can check that the estimates obtained by (37) preserve the structure. In fact let $[\int_{t_0}^t \overline{x}_l \overline{x}_l^T dt]^{-1} = (v_1, ..., v_l, v_{p+1})$, where v_i are column vectors, and let $[\int_{t_0}^t \overline{x}_l \overline{x}_l^T dt] = (w_1, w_2, ..., w_{p+1})^T$, where w_i are column vectors, then $w_i v_i = 1$ and $w_i v_j = 0$ for $j \neq i$. By substituting in the estimate expression in place of $\dot{x}_l$ its expression $(x_{l2}, ...x_{lp}, \dot{x}_{lp})$ we find that $\int_{t_0}^t \dot{x}_l \overline{x}_l^T dt = (w_2, w_3, ...w_p, \star)^T$, so that

$$\int_{t_0}^t \dot{x}_l \overline{x}_l^T dt [\int_{t_0}^t \overline{x}_l \overline{x}_l^T dt]^{-1} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \cdots & \cdots & \cdots & 1 & 0 \\ \star & \star & \cdots & \cdots & \star & \star \end{pmatrix},$$

which clearly preserves the structure of A and b.

Remark 4.2. If in the expression of the interconnection condition defined by (13) instead of $\dot{x}$ we have just some of the components of the state vector, that is $\overline{C}\dot{x}$, and the same for the definition of the transition factor (19), provided that $\overline{C}\dot{x}_{av}^0(\tau^-)$ and $\overline{C}\dot{x}_{av}^0(\tau^+)$ are non zero, everything still applies. (In fact in such a case we would have a matrix $\overline{C}$ selecting the components of $\dot{x}$ multiplying $\dot{x}$ in expression (13) and we would find the same matrix in (19) which would lead to an expression for γ_0 in equation (57) of the Appendix of the form

$$\gamma_0 \geq 1 - \frac{2\|\overline{C}(\tilde{M}_1 e^{A\tau} x_0 + A^{-1}\tilde{M}_1 e^{A\tau} b)\|}{\|\overline{C}\dot{x}_{av}^0(\tau^-)\|}$$

which leads to

$$\gamma_0 \geq 1 - k'(1 - e^{\lambda\Delta\tau})$$

where k' is a positive constant since $\|\overline{C}\dot{x}_{av}^0(\tau^-)\|$ is bounded away from zero by assumption. Then everything is the same.)

Remark 4.3. Assume that in the expression of $W(\tau)$ given in (20) we add a factor $s(\tau)$ with the properties that $s(\tau) \in [\frac{1}{K}, 1)$ for all τ, $K \geq 1$ and $s(\tau) \geq 1 - \nu$ for $\tau_{i-1} < \tau \leq \tau_i$, with $\nu \ll 1$. Then the proof of Theorem 4.1 proceeds at the same way with the following modifications. Expression (32) becomes

$$\frac{2c_2\beta}{a(1-\nu)} e^{\Delta^2/\sigma^2}(a+\varepsilon) + \varphi \leq 1 - \rho_0,$$

expression (33) becomes

$$(\tau - \tau_i)^2 > \max\{\Delta\tau^2, \frac{\varepsilon - a}{k} + \frac{2c_2\beta(\varepsilon+a)e^{\Delta^2/2}}{k(1-\rho_0-\varphi)(1-\nu)}\} := (\Delta\tau^+)^2,$$

and condition (34) becomes

$$(\Delta^c_{min})^2 > \frac{4e^{\Delta^2/\sigma^2}(\varepsilon+a)}{(1-\rho_0-\varphi)h(1-\nu)} + \frac{2(\varepsilon-a)}{h}.$$

The introduction of such a factor can be helpful in those cases in which (34) is not satisfied because the distance between parameters generating adjacent segments is not big enough compared to noise. In fact if we can find a function $s(\tau)$ that, for the values of τ for which condition (34) has been computed, is smaller than $1/K'$, for $K' \leq K$ big enough, then condition (34) becomes

$$(\Delta^c_{min})^2 > \frac{4e^{\Delta^2/\sigma^2}(\varepsilon+a)}{(1-\rho_0-\varphi)h\ K'} + \frac{2(\varepsilon-a)}{h},$$

which is satisfied for smaller values of Δ^c_{min}. Note also that such a factor can depend on the classification of the current segment, and it can be introduced for some of the classes only. A good choice of such a function will be presented in Section 5.4.

4.2 Algorithm implementation

The segmentation algorithm was implemented in MATLAB 6.0 in the case of planar motion modeled by systems of the form

$$\begin{pmatrix} \dot{x} \\ \ddot{x} \\ \dot{y} \\ \ddot{y} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ a_{1x,i} & a_{2x,i} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{1y,i} & a_{2y,i} \end{pmatrix} \begin{pmatrix} x \\ \dot{x} \\ y \\ \dot{y} \end{pmatrix} + \begin{pmatrix} 0 \\ b_{x,i} \\ 0 \\ b_{y,i} \end{pmatrix} + d(t), \tag{38}$$

with asymptotically stable dynamics in each interval between two switching points τ_{i-1} and τ_i. The interconnection condition that holds in this case is by replacing $\dot{x}$ with $\overline{C}(x, \dot{x}, y, \dot{y})^T$ in equation (13), with

$$\overline{C} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

By virtue of Remark 4.1, the particular structure of system (38) does not require any change. By virtue of Remark 4.2, the modified interconnection condition also does not affect result of Lemma 4.5 since $\dot{x}^0_{av}(\tau^+)$ and $\dot{x}^0_{av}(\tau^-)$ and $\dot{y}^0_{av}(\tau^+)$ and $\dot{y}^0_{av}(\tau^-)$ are nonzero. (If they were zero it would imply periodicity of solutions of the system under study which is not true since the dynamics are asymptotically stable.) For computing the estimates of the parameters according to (14) we used the discrete time version of system (38), that is

$$\begin{pmatrix} x_k \\ x_{k+1} \\ y_k \\ y_{k+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ a_{1x,i} & a_{2x,i} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{1y,i} & a_{2y,i} \end{pmatrix} \begin{pmatrix} x_{k-1} \\ x_k \\ y_{k-1} \\ y_k \end{pmatrix} + \begin{pmatrix} 0 \\ b_{x,i} \\ 0 \\ b_{y,i} \end{pmatrix} + d(k),$$

where we have used the same coefficients notation as used in system (38) for simplicity. In the actual segmentation algorithm, $W(\tau)$ takes the following form

$$W(\tau) = \exp\left(-\frac{(e_a(\tau) - e_a^c)^2}{\sigma_a^2}\right) \frac{\mathrm{Tr}(\tau)\exp(-\bar{e}_p^2(\tau))s(\tau)p(\tau)}{a + e_a(\tau)} \tag{39}$$

where we have introduced some slight modifications that exploit some additional information on the characteristics of the signal. In particular, the first term, $\exp(-\frac{(e_a(\tau)-e_a^c)^2}{\sigma_a^2})$, represents a Gaussian distribution of the approximation error around a mean value: we can obtain a guess of e_a^c and σ_a^2 by processing part of the data. The parametric error $\overline{e}_p$ takes into account also possible non-spherical shapes of the distribution of the parameters around the centers. Using the same notation used for defining e_p it can be written as $\overline{e}_p^2(\tau) = \min_j(\hat{A} - A_j)^T\Sigma_j^{-1}(\hat{A} - A_j)/\sqrt{\det(\Sigma_j)}$.

The term $s(\tau)$ is a shaping term, satisfying the properties described in Remark 4.3, which can be used to include additional information other than that derived from the dynamical parameters. Introducing such a term does not affect results of Theorem 4.1 as explained in Remark 4.3. The particular form we choose is introduced in Section 5.4.

The term $p(\tau)$ is introduced in case we have pauses in our signals. Pauses occur for the drawing tasks described in the next section and must be taken into account by the algorithm. Since we assumed that there are no pauses within a segment, eventual pauses are likely to be between one segment and the following one. If at time τ the segment (t_0, τ) contains a pause it will be penalized in an amount proportional to the length of the pause, i.e. $p(\tau) = k/(\text{pause length})$. Pauses are detected by making the difference between adjacent signal's samples and checking if the result is zero for more than $15 - 20$ steps.

The time t_0 which is the starting point of each iteration is obtained as explained at the end of the proof of Theorem 4.1. The way we implement this is by taking into account that the end of each segment reaches a steady state in which $\dot{x}$ and $\dot{y}$ are very small and comparable to noise (since the systems are all stable asymptotically). Then we estimate the length of the signal after $\hat{\tau}_i$ which has a poor content of information with respect to a given threshold depending on the noise level. This gives an estimate of the time interval we have to add to $\hat{\tau}_i$ in order to find a point t_0 which lies in the following segment.

Finally, the segmentation algorithm can be described in the following basic steps:

1. initialization: $t_0 = \tau_0$, $t_M = \tau_l$, $i = 1$;
2. maximize $W(\tau)$ for $\tau \in (t_0, t_M]$: $\hat{\tau}_i = \max_{\tau\in(t_0,t_M]} W(\tau)$;
3. compute class j of the segment found: $j = \arg_{k\in\{1,\dots,m\}}(\|\hat{A}(\hat{\tau}_i) - A_c^k\|) \leq r_k$;
4. compute $\Delta\tau$;
5. $t_0 = \hat{\tau}_i + \Delta\tau$;
6. $i = i + 1$;
7. go to (1);

where we recall that τ_0 and τ_l are the starting and ending points of the data stream.

5 Experimental results

To test our approach, we studied a 2D drawing task in which a set of shapes were drawn by five different subjects using a computer mouse (see examples in Figure 4). We hoped that these experiments could verify that human subjects use different dynamics for accomplishing different elementary tasks so to allow automatic recognition of actions. We considered just the motion of the wrist in 2D as a

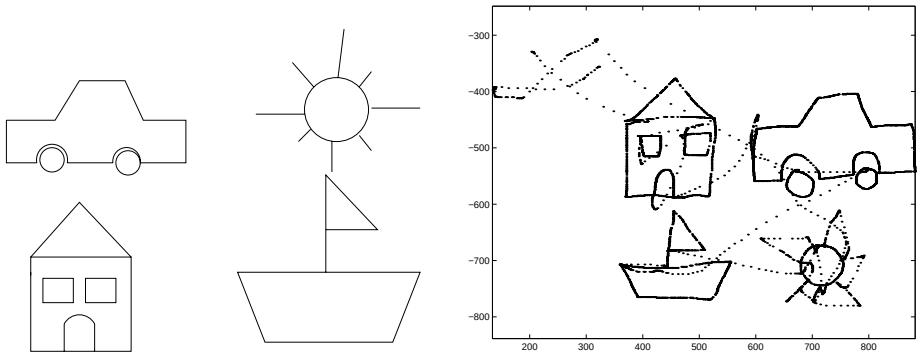

Figure 4: Prototypes of the four shapes shown to the subject and example of traces in (x, y) plane captured by the capturing system.

simple example to start with, and to develop a procedure for analyzing data easily generalizable and not subject to complexity constraints. In the following section we describe the device used for capturing data and details on the experiments.

5.1 Experimental setup

Our subjects drew using the XPaint program on a PC running Red Hat Linux 7.2 with a screen measuring 1600×1200 pixels and a working window of 700×500 pixels. The user left the trace of the trajectory in the working window only when the left mouse button was pressed. For acquiring x and y time traces we implemented a C routine which was activated in the background at the beginning of each experimental session and sampled the (x, y) position of the pointer everywhere on the screen at the rate of 100 Hz and a spatial resolution of one pixel. The routine makes use of XWindow libraries and captures the pointer position through the function "XQueryPointer" which is called by a timer every 10 ms and gives the coordinates in pixels with respect to the upper left corner of the screen. Every 30 minutes the data was saved into files by means of a parallel process. The data so obtained consists of an array with three columns containing time, x position at that time, y position at the same time. The time interval between one sample and the following one turned out to be mostly constant except for slight variations every once in a while due to higher priority of other processes. In order to have constant sampling time the data was processed through an algorithm that linearly interpolates data in the regions in which the time interval is not exactly 10 ms. Pixelization of the coordinates does not heavily affect the data since the trajectories under study are usually more than 50 pixels long.

We defined 4 different drawings by means of prototypes shown in Figure 4: car, sun, ship, and house. Each of the 5 subjects was shown the prototypes and was asked to reproduce them on a 700×500 pixel canvas; the dimensions of each drawing could be chosen arbitrarily according to the ones with which the user was more comfortable, the only specification was to reproduce the prototypes with as high fidelity as possible in a reasonable amount of time. Each subject drew 10-20 examples for each shape.

In order to accomplish each drawing task the user had to perform a sequence of actions such as "reach a point A" and "draw a line up to point B". These actions are the ones that we will consider as candidates for being elementary motions and then defining a pair of movemes. The idea is then to use the result of Theorem 4.1 so as to find the sequence of reach and draw movements that the user did in order to accomplish the task and the switching times between one and the other. In Figure 4 (right) we show also an example of the traces followed by a user while drawing in (x, y) plane. In order to apply Theorem 4.1 we check first that reach and draw actions define a well posed pair of movemes according to Definition 2.3.

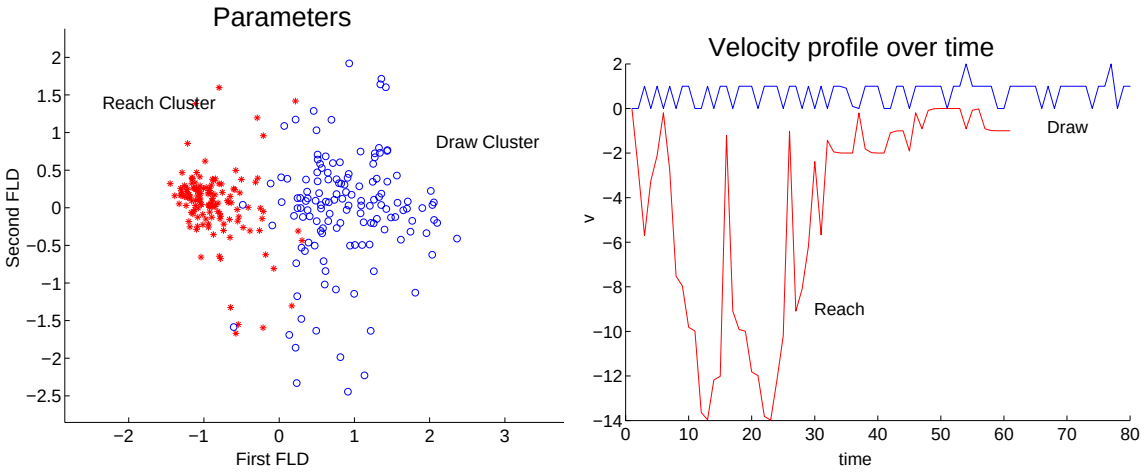

Figure 5: Parameter estimates for reach and draw examples projected on the first two Fisher linear discriminants (left). Typical velocity profile for reach and draw (right).

5.2 Classification

We start from the hypothesis that "draws", which are straight lines traced with a specific intention (like drawing a side of the house), and "reaches", which happen with the intention of shifting fast the equilibrium position, define a well-posed pair of movemes. We segmented out by hand a set of straight draws from houses and cars drawn by 2 of the subjects. Reach examples were obtained from a special experiment session in which the users had to point and click at random buttons appearing on a 700×500 pixels window during a simple video game implemented in MATLAB 6.0.

Before settling for model (38), we considered several other dynamical models for representing the reach and draw signals in time, starting from a first order, decoupled model for x and y motion,

$$\dot{x} = a_{1x}x + b_x$$
$$\dot{y} = a_{1y}y + b_y,$$

and proceeding to a second order coupled model,

$$\begin{pmatrix} \dot{x} \\ \ddot{x} \\ \dot{y} \\ \ddot{y} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ a_{1x} & a_{2x} & a_{3x} & a_{4x} \\ 0 & 0 & 0 & 1 \\ a_{3y} & a_{4y} & a_{1y} & a_{2y} \end{pmatrix} \begin{pmatrix} x \\ \dot{x} \\ y \\ \dot{y} \end{pmatrix} + \begin{pmatrix} 0 \\ b_x \\ 0 \\ b_y \end{pmatrix}. \tag{40}$$

The reach dynamical parameters were estimated from 140 examples of reach trajectories obtained from the video game implemented in MATLAB, and the draw dynamical parameters were estimated from 140 examples of draw trajectories segmented out from cars and houses of 2 of the subjects. The dynamical parameters were estimated for each one of the dynamical models proposed (first order for x and y, decoupled; first order for x and y, coupled; second order for x and y, decoupled; second order for x and y, coupled).

By proceeding with standard pattern recognition techniques (see (Bishop, 1995) for example), we trained a Gaussian classifier for the parameters derived from the 140 examples per class (training set) for each one of the model classes proposed, and obtained the best results for the second order for x and

y, decoupled, dynamical model (obtained by letting $a_{3x} = 0$, $a_{4x} = 0$, $a_{3y} = 0$, $a_{4y} = 0$ in system (40)):

$$\begin{pmatrix} \dot{x} \\ \ddot{x} \\ \dot{y} \\ \ddot{y} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ a_{1x} & a_{2x} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{1y} & a_{2y} \end{pmatrix} \begin{pmatrix} x \\ \dot{x} \\ y \\ \dot{y} \end{pmatrix} + \begin{pmatrix} 0 \\ b_x \\ 0 \\ b_y \end{pmatrix}. \tag{41}$$

For such a model we obtained 3.2% training error, and we tested the generalization properties of the resulting classifier on a test set of 323 additional reach examples (obtained from the MATLAB videogame) and 118 additional draw examples obtained from the drawings of other two subjects (different from the ones used for the training set) and obtained 3.63% test error.

Figure 5 represents the projection of the parameters belonging to the training set (living in $\mathbb{R}^4$) on the first two Fisher linear discriminants (Bishop, 1995) and typical velocity profiles for the draw and reach trajectories. We let $\hat{\mathcal{C}}^R$ and $\hat{\mathcal{C}}^D$ denote the reach and draw clusters, respectively, according to the notation used in Section 2.3. From the right figure of Figure 5 we notice that a reach trajectory is usually characterized by a bell shaped velocity profile with high velocity variation in a small time, while a draw trajectory is characterized by an almost constant or slowly varying velocity.

Since our data set contains also circular shapes like the wheels of the cars, we also introduced a circle class beyond the reach and draw classes. The dynamical model by which we represent such a class is system (40), so that we have 8 parameters for classification. We considered an additional parameter that is the value of ω/T were ω is the principal frequency estimated and T is the duration of the trajectory: we expect for a circle that to be about 2π. We then trained a Gaussian classifier in $\mathbb{R}^9$ on a training set composed of 101 examples derived from the wheels of the cars and the suns of two of the subjects and obtained 4% training error on the circle class. We then tested the classifier on a test set of 124 elements derived from the wheels of the cars and suns of two other subjects (different from the one used for training) and obtained 8% error on the circle class. The higher test error on the circle class is due to higher variance across subjects than the variance we have for reach and draw tasks. Introducing the circle class does not alter the training and test errors obtained for the reach and draw classes (the percentage of reach and draws classified as circles is close to be zero), therefore the cumulative training and test errors are 3.4% and 4.6% respectively.

5.3 Well-posedness

Before going on and basing our classification algorithm on the found sets $\hat{\mathcal{C}}^R$ and $\hat{\mathcal{C}}^D$, we have first to check that reach and draw classes of signals are well-posed so that situation depicted in Figure 1 does not happen. To check this, we find candidate constraints which can describe reach and draw trajectories, so that we may apply Lemma 2.3. Reach trajectories are asymptotically stable with bell-shaped velocity profiles. Draw trajectories are characterized by asymptotic stability properties and by straight lines in (x, y) plane. These requirements for the model (41) imply $a_{1x} = a_{1y}$ and $a_{2x} = a_{2y}$. Some of these parameters are reported in Figure 6 where we can see that their classification is ambiguous since they lie in the boundary region between the first and the second cluster. Then we have a situation analogous to the one reported in Figure 1, where the light sets are $\hat{\mathcal{C}}^D$ and $\hat{\mathcal{C}}^R$ and the dark set is made up by elements like the diamonds in Figure 6. Thus there exist parameters that generate trajectories satisfying both draw (asymptotic stability and straight lines in (x, y) plane) and reach constraints (asymptotic stability and bell shaped velocity profile with high acceleration peak) whose class is undecidable.

As an extreme example of this, we show in Figure 7 the shape of a house that has been artificially generated by parameters lying in the region in between the clusters of Figure 6 (dark set of Figure 1), which the classifier classifies as reaches. This happens because the dynamical parameters associated to draw trajectories can significantly differ from each other according to the particular task, and also the velocity profile can consistently vary with respect to the one shown in Figure 5. We show these differences in Figure 8, where we report the draw parameters when a user draws straight lines between two points (as it happens in the draws of the house, ship, car), or a line trying to trace an already existing line, or just a line with no constraints (as it happens in the rays of the suns). We decide

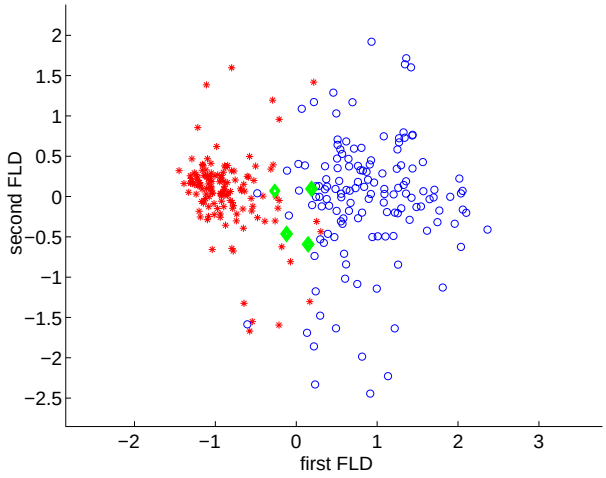

Figure 6: Diamonds represent some of the parameters corresponding to straight (x, y) trajectories.

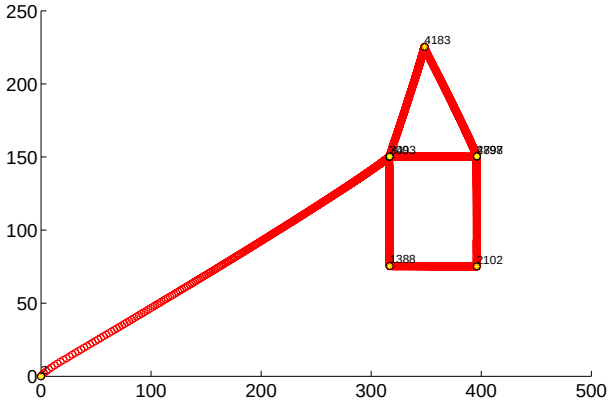

Figure 7: House generated by points in the overlapping region of clusters of Figure 6.

therefore to use three classes instead of one for the draw: we call them targeting, tracing, and free motion respectively.

Using these definitions, we see from Figure 8 that there is an evident overlapping of the parameter sets of the reach class and free motion class. Therefore we exclude from the panorama the free motion class, and show that the draw class, seen as union of the tracing and targeting motions, can be described in terms of constraints g_D, h_D, g_R, h_R as introduced in Lemma 2.3, such that the system of equations (8) is infeasible. Driven by the characteristics of the velocity profiles of the targeting and tracing draw and reach reported in the bottom right plot of Figure 8, we define the following constraints. The reach trajectories achieve the desired value in a time smaller than a fixed one with respect to a unit step input (which implies a certain acceleration peak), and in the draw trajectories the velocity variation has to be smaller than a given value. We then rewrite these constraints in the form of Lemma 2.3 as $\dot{x} - a\dot{y} = g_D(\dot{x}, \dot{y}) = 0$ and $b - \ddot{x} - \ddot{y} = h_D(\ddot{x}, \ddot{y}) \geq 0$ for the draw motions, and $\ddot{x} + \ddot{y} - c = h_R(\ddot{x}, \ddot{y}) \geq 0$

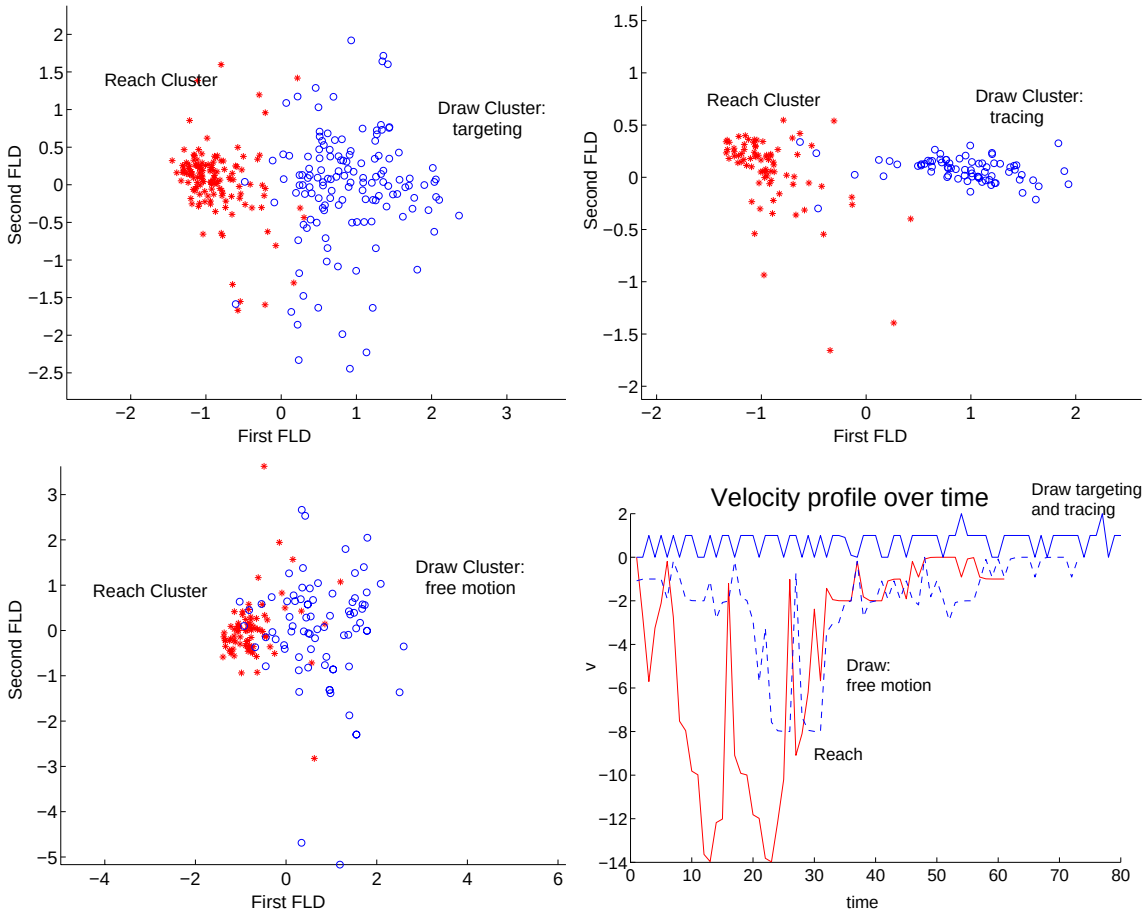

Figure 8: Parameter estimates for reach and draw targeting, draw tracing and free motion draw with velocity profiles.

for the reach motions, so that the system

$$\begin{aligned} g_D(\dot{x}, \dot{y}) &= 0 \\ h_D(\ddot{x}, \ddot{y}) &\geq 0 \\ h_R(\ddot{x}, \ddot{y}) &\geq 0 \end{aligned} \tag{42}$$

becomes infeasible for suitable b and c. Then if we assume that the constraints above define fair specifications for reach and draw trajectories for the values of b and c that make system (42) infeasible, then the reach and draw classes of signals are well-posed according to Lemma 2.3 . Moreover the $\hat{\mathcal{C}}^R$ and $\hat{\mathcal{C}}^D$ clusters of Figure 5 represent reach and draw actions well, which thus define a pair of movemes M^R and M^D. Therefore in what follows we will not deal with free motion draw so to have well-posedness and fairly apply results of Section 2 and Section 4 to our problem. (Specifically we eliminate the sun diagram from our set of shapes.)

5.4 Segmentation algorithm performance

We implemented the proposed segmentation algorithm in MATLAB on the data acquired as described in the previous sections considering a number of three movemes: the reach, the draw, and the

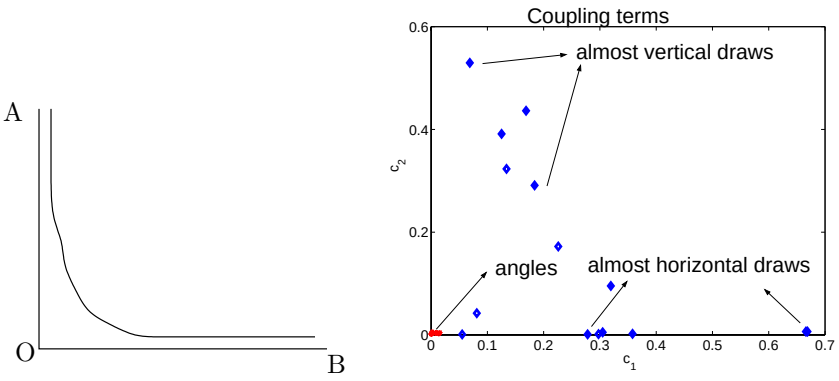

Figure 9: The right plot reports the coupling parameters (stars) obtained for angles AOB shown in the left plot, and the coupling parameters (diamonds) obtained for the vertical and horizontal draws separately.

circle movemes. Before reporting the algorithm performance we describe the choice of the function $s(\tau)$ introduced in Section 4.2 in expression (39). The need for introducing a term with additional informations comes from the fact that system (41), chosen for representing the movemes, can approximate with acceptable approximation errors angles in x, y plane, as shown in the left plot of Figure 9, while having parameters that still lie in the $\hat{\mathcal{C}}^R$ or $\hat{\mathcal{C}}^D$ sets. Then the estimated parameters of system (41) for a given trajectory do not contain information to discriminate between one draw and an angle. This is due to the absence of any xy coupling information. In fact coupling information would discriminate quite clearly between the single draw case and the angle of Figure 9: we can check that for approximating an xy trajectory of the kind of the angle of Figure 9 with the simplest system containing xy coupling, such as

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} d_1 & c_1 \\ c_2 & d_2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + b. \tag{43}$$

We obtain estimated coupling terms $(\hat{c}_1, \hat{c}_2)^T = \hat{c}$ that are close to zero for the angle and bounded away from zero for the single draw, as shown in the right plot of Figure 9 (the same we obtain for the single reaches that are approximatively straight lines). Thus we choose a shaping term in expression (39), for the reach and the draw classes, of the form

$$s(\tau) = \frac{1}{1 + L\ \exp(-(\hat{c} - \overline{c})^T \Sigma_c^{-1} (\hat{c} - \overline{c}))/\sqrt{\det(\Sigma_c)}},$$

with $L \geq 1$, where $\overline{c}$ and Σ_c are obtained by means of a learning phase in which we train the Gaussian classifier, $\exp(-(z - \overline{c})^T \Sigma_c^{-1} (z - \overline{c}))/\sqrt{\det(\Sigma_c)}$, on a set of about 25 examples of angles. The value of Σ_c turns out to be very small resulting in a very narrow Gaussian around the mean as we can deduce form the concentrated cluster of angle's parameters of Figure 9. By simple computation we can show that $s(\tau)$ satisfies the conditions of Remark 4.3 since $s(\tau) \in [1/(1+L), 1)$ for all τ, and $s(\tau) \approx 1$ for $\tau \leq \tau_i$, i.e. when we have just one draw or one reach. Moreover when the segment under observation is an angle $s(\tau) \ll 1$.

Since in our data set some squares (windows of the houses) have rounded angles and look very similar to circles, we introduced a higher level step in the algorithm, in which we decide if a segment detected as a circle is more likely to be a square. At each iteration in which a circle is detected, to decide if the data segmented as a circle is more likely to be a square, we run the segmentation algorithm again on that data without the circle classifier (that is by assuming that the data is a sequence of reaches or draws or both). Then if the algorithm segments it into a sequence of draws, we compute the likelihood of each draw that has been detected as the product $\exp\left(-\frac{(e_a(\tau) - e_a^c)^2}{\sigma_a^2}\right) \exp(-\bar{e}_p^2(\tau))$, which is the part

of (39) that quantifies how good the detected segment is as representative of its class. We then average the likelihood of all the detected draws and compare it to a threshold obtained by preprocessing some of the squares and some of the circles (about 10 examples each). This higher level process does not affect performance drastically, but turns out to be helpful in 3–4 cases in which the windows of the houses have not evident corners.

Finally, for minimizing the algorithmic time, we set t_M *a priori* to be t_0 plus the maximum duration of a segment that in our case turned out to be 500 time steps.

The algorithm takes as input the signal $(x(t), y(t))$ and gives as outputs a sequence of segmentation points and the classification of the trajectory between two detected segmentation points. The algorithm performance was computed by assuming a ground truth: we expected to detect a segmentation point at the beginning and at the end of each reach, draw and circle, and also we expected each one of them to be properly classified. Then the algorithm error was computed as the sum of classification error (i.e., a trajectory which is correctly segmented but wrongly classified) and segmentation error (i.e., a trajectory which is over segmented: every detected point that is not in the data is counted as an error; or a missed segmentation point). An estimate of such an error was computed on segmentation results on cars, ships, houses sequences deriving from two subjects each. The error estimate is reported in Table 1 which was obtained by counting the total number of segmentation points detected (denominator) and the number of segments which were clearly mis-classified or mis-segmented (numerator).

Table 1: Algorithm error

	classification error	segmentation error	cumulative error
CAR	$112/1333 = 8.4\%$	$20/1333 = 1.5\%$	9.9%
HOUSE	$108/1050 = 10.29\%$	$23/1050 = 2.19\%$	12.48%
SHIP	$99/1093 = 9.06\%$	$3/1093 = 0.27\%$	9.3%

Table 2: Confusion matrix

	Predicted			
Actual	Reach	Draw	Circle	unsegmented/unclassified
Reach	167	32	0	10
Draw	35	395	0	12
Circle	0	0	44	2

The average error is about 10.5%. We report several pictures which show the segments classified as reach, the segments classified as draws, the ones classified as circles and the unclassified ones. The little circles represent the segmentation points that the algorithm found. The units on the x and y axis are in pixels. From the table it is evident that the major contribution to the algorithm error is due to classification error, in particular to confusion between reach and draw since it can happen that a subject draws quickly without paying enough attention so that some of the draws that should be targeting draws turn out to be free motion draws (see for example Figure 13). Also some of the reaches may be confused with draws because they are accomplished too slowly and carefully (especially when they are short). These errors are more clearly reported in Table 2 where we show the confusion matrix obtained counting the number of reaches, draws and circles in a set of cars and ships coming from one of the subjects. Some confusion is also due to the fact that some of the segments are too short, as

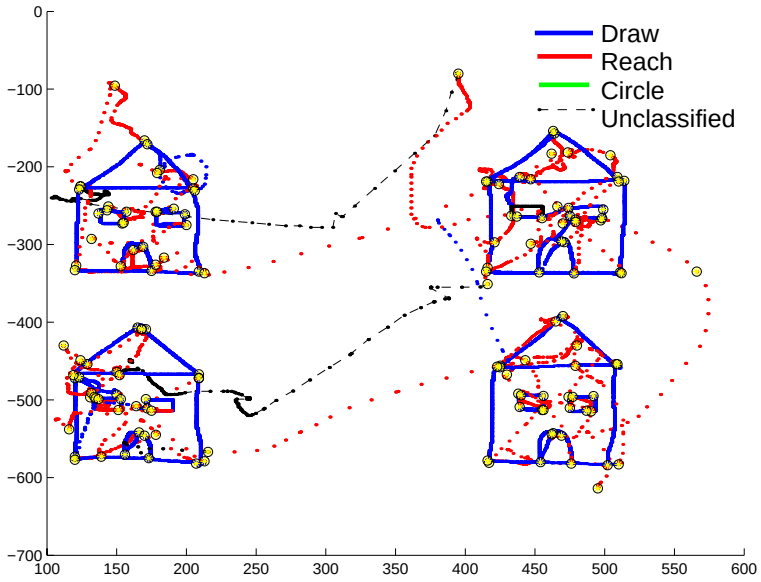

Figure 10: Segmentation results on 4 houses of subject 3.

for example in some of the windows of the houses, so that there is not enough information to classify them properly (see Figure 10). This explains also why among the three different categories (ship, car, house) the house category is the one which shows the highest error. The segmentation error comes almost entirely form missed segmentation points and not from over-segmentation: the biggest portion comes from the windows of the houses in which some corners were not detected because of too small dimensions: at such small dimensions the hand dynamics is likely to vary with respect to the one used for larger motions and the pixelization error and the mouse dynamics may be not negligible anymore (see Figure 10 again). In Figure 14 we report for completeness the results of the segmentation algorithm on some of the suns: as we can notice by the figure, the classification error of the rays is about 50%, due to the confusion between reaches and free motion draws, as explained in the previous section.

5.5 Categorization

To illustrate a possible usage of the output of the segmentation algorithm, we want to recognize what is the category (car, house, ship) of a certain drawing based on the sequence of movemes that according to the segmentation algorithm composes it. To this end, we develop an elementary categorization algorithm that takes as input for each shape the sequence of movemes that it is composed of according to the segmentation algorithm, and recognizes the category of a shape based just on the number of reaches, draws and circles that the representing movemes sequence contains. The way this is done is by associating to each sequence corresponding to one of the three shapes the vector of natural numbers $(R, D, C)^T$, which are the number of reaches, draws and circles present in the sequence respectively. We then train a Gaussian classifier with the $(R, D, C)^T$ vectors of 15 cars (as drawn by one subject), 9 houses (from two subjects), and 4 ships (from two subjects) so to learn the Gaussian distributions in $\mathbb{R}^3$ that represent the three different shapes, and obtain 0% training error. Based on the learned distributions we can classify the remaining data according to a Gaussian metric. We thus perform the test on the remaining 53 ships (from three subjects), 36 houses (from three subjects), 31 cars (from two different subjects) and obtain 5.4% test error which is quite small since we based our discrimination just on the basis of the number of movemes and not on their order in time. If we add the suns to this dataset the error increases drastically up to 11% on the test set. This happens because when in a sequence a

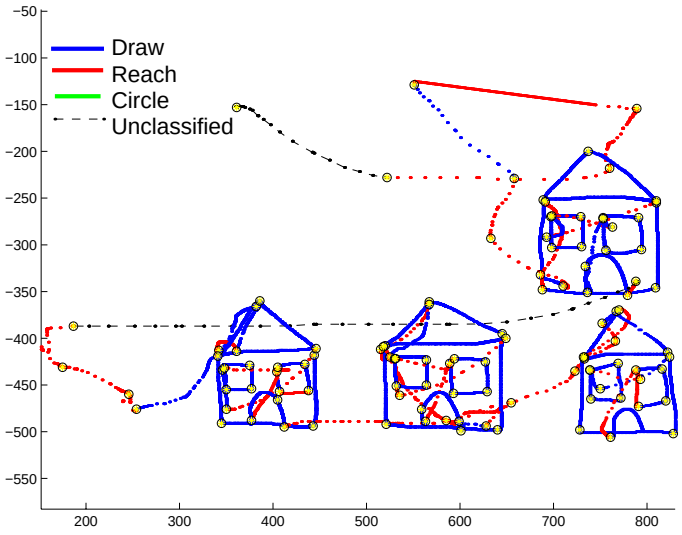

Figure 11: Segmentation results on 4 houses of subject 2.

circle is detected, depending on the numbers of reaches and draws, the vector $(R, D, C)^T$ can fall either in the sun Gaussian distribution or in the car Gaussian distribution because in the training set there are cars with just one detected circle. Therefore the discrimination is based just on the number of reaches and draws, which is not reliable in the case of the sun where the rays can be classified either way. One could use more sophisticated tools like introducing some information about the order of the performed movemes so to use the segmentation algorithm output for higher level kinds of discrimination.

6 Conclusions

We have introduced a dynamical formulation for the notion of movemes. We have addressed the classification and segmentation problems and proposed an algorithm with error and performance analysis. We have also provided a procedure for finding movemes from data and shown with an example how it is applicable in practice. The experimental results show that the moveme segmentation and classification performance of the proposed algorithm is about 90% on our data set when training and testing are performed on data coming from distinct subjects. We finally show, with an example, that the output of the segmentation algorithm can be used so to solve higher level tasks like discriminating between activities composed by movemes and found an error on our data set of 5% when using a simple-minded recognition strategy.

Future directions include the exploration of 3D motion of the human body with the same approach. We hope that the tools here developed will render the search for movemes faster, and that the segmentation and classification algorithm will be applicable to the higher dimensional case with minor modifications. The 3D motion case includes also more joints in the considered dynamics, so that it will be possible to look at the whole body motion in 3D. Furthermore, it will be interesting to generalize the current segmentation and classification algorithms to the on-line case. At this point it would be useful also to set up a possible solution to the prediction problem, which is one of predicting the next action (or actions) on the basis of what has already happened. Moreover exploring different classes of dynamical systems may help modeling human motion with greater accuracy. Also issues regarding to what extent models are user independent and to what extent we need to train on different individuals should be addressed.

The ideas proposed in this paper suggest some theoretical directions which could be explored. These

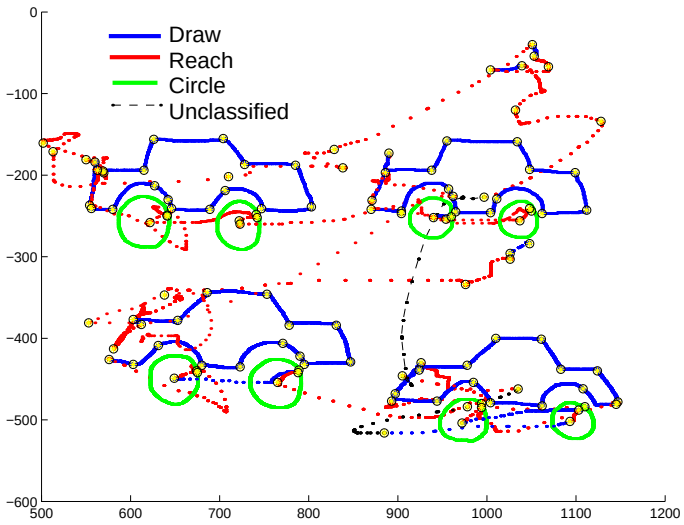

Figure 12: Segmentation results on 4 cars of subject 1.

directions include the theory of observability and estimation of hybrid systems that are systems in which we have both continuous and discrete dynamic evolution. The problem is to determine the parameter values (discrete states), continuous state, and switching times between discrete states, based on the time evolution of an output variable.

Acknowledgment

The authors would like to thank all of the people that participated to the experiments presented here.

7 Appendix

Proof of Lemma 4.1

Proof. Let $\hat{\Theta} := (\hat{A}, \hat{b})$, $\Theta_i = (A_i, b_i)$ and $\Theta_{i+1} = (A_{i+1}, b_{i+1})$, then for $\tau > \tau_i$ equation (15) becomes

$$\hat{\Theta}(\tau) = \left[\int_{t_0}^{\tau_i} \Theta_i \overline{x}\,\overline{x}^T dt + \int_{\tau_i}^{\tau} \Theta_{i+1} \overline{x}\,\overline{x}^T dt\right] \left[\int_{t_0}^{\tau} \overline{x}\,\overline{x}^T dt\right]^{-1}$$

and subtracting Θ_i we find

$$\tilde{\Theta}(\tau) := \hat{\Theta} - \Theta_i = \left[\int_{\tau_i}^{\tau} (\Theta_{i+1} - \Theta_i) \overline{x}\,\overline{x}^T dt\right] \left[\int_{t_0}^{\tau} \overline{x}\,\overline{x}^T dt\right]^{-1} \tag{44}$$

Define $g(\tau) = \|\tilde{\Theta}(\tau)\|^2 = \mathrm{tr}(\tilde{\Theta}(\tau)^T \tilde{\Theta}(\tau))$ and identify the matrix $\tilde{\Theta} \in \mathbb{R}^{n \times (n+1)}$ with the vector $\tilde{\theta} \in \mathbb{R}^{n(n+1)}$ by stretching columns so that $g(\tau) = \tilde{\theta}(\tau)^T \tilde{\theta}(\tau)$. Since $g(\tau)$ is composition of C^∞ functions, by Taylor's theorem we have

$$g(\tau) = g(\tau_i) + g'(\tau_i)(\tau - \tau_i) + \frac{1}{2} g''(c)(\tau - \tau_i)^2 \,, \qquad \tau \in [\tau_i, \tau_{i+1}]$$

where $c \in [\tau_i, \tau]$. Clearly $g(\tau_i) = 0$ and we can check that $g'(\tau_i) = 0$ also. In fact

$$g'(\tau) = (\tilde{\theta}')^T \tilde{\theta} + \tilde{\theta}^T \tilde{\theta}'$$

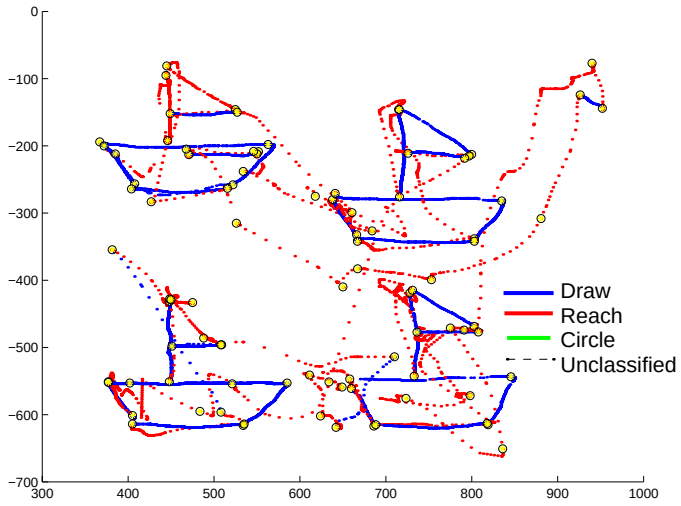

Figure 13: Segmentation results on 4 ships of subject 1.

and since $\tilde{\theta}(\tau_i) = 0$, $g'(\tau_i) = 0$. On the contrary $g''(\tau_i) \neq 0$ in fact

$$g''(\tau) = (\tilde{\theta}'')^T\tilde{\theta} + 2(\tilde{\theta}')^T\tilde{\theta}' + \tilde{\theta}^T\tilde{\theta}''$$

so that $g''(\tau_i) = 2(\tilde{\theta}')^T(\tau_i)\tilde{\theta}'(\tau_i)$. Also $\tilde{\theta}'(\tau_i) \neq 0$ because $\tilde{\Theta}'(\tau_i) \neq 0$ as we can see from the following expression

$$\tilde{\Theta}'(\tau) = \left[(\Theta_{i+1} - \Theta_i)\overline{x}(\tau)\,\overline{x}(\tau)^T - \tilde{\Theta}\right]\left[\int_{t_0}^{\tau} \overline{x}\,\overline{x}^T dt\right]^{-1}$$

which is non zero for $\tau = \tau_i$ since $(\Theta_{i+1} - \Theta_i)\overline{x}(\tau_i)\overline{x}(\tau_i)^T \neq 0$ by (13). Finally we have

$$g(\tau) = \frac{1}{2}g''(c)(\tau - \tau_i)^2 \qquad \forall t \in [\tau_i, \tau_{i+1}]$$

with $g''(c) > 0$. Since for each τ there exists a c such that the above relation is satisfied we can consider the minimum of $g''(c)$ on $[\tau_i, \tau_{i+1}]$ which is bounded away from zero since $g''(\tau_i) \neq 0$ and $g(\tau) > 0$ for $\tau > \tau_i$. Then

$$g(\tau) \geq k_1(\tau - \tau_i)^2 \qquad \forall t \in [\tau_i, \tau_{i+1}]$$

for $k_1 = \min_{\tau\in[\tau_i,\tau_{i+1}]} \frac{1}{2}g''(c)$. □

Proof of Lemma 4.2

Proof. Let $\tilde{x} = x - \hat{x}$ so that

$$e_a(\tau) = \frac{1}{\tau - t_0}\int_{t_0}^{\tau} \tilde{x}^T(t)\tilde{x}(t)dt$$

where $x(t)$ is generated by $\dot{x} = A_i x + b_i$ for $t \in [t_0, \tau_i]$ and it is generated by $\dot{x} = A_{i+1}x + b_{i+1}$ for $t \in [\tau_i, \tau_{i+1}]$. Using the short-hand notation $\hat{\Theta} := (\hat{A}|\hat{b})$, $\Theta_i = (A_i|b_i)$, $\Theta_{i+1} = (A_{i+1}|b_{i+1})$, $\tilde{\Theta} = (A|b) - (\hat{A}|\hat{b}) = \Theta - \hat{\Theta}$, where $(A|b)$ can be either $(A_i|b_i)$ either $(A_{i+1}|b_{i+1})$. Thus we write the dynamics for $\tilde{x}$ as

$$\dot{\tilde{x}} = A\tilde{x} + \tilde{\Theta}\hat{\overline{x}} \tag{45}$$

where $\hat{x}$ is the state of (16) with the parameter estimates obtained from (15) with $d(t) = 0$ and $\delta = 0$, $\hat{\overline{x}} = (\hat{x}^T, 1)^T$. Then we can review $\tilde{x}$ as function of $\tilde{\Theta}$ so that also $e_a = e_a(\tilde{\Theta}) := g(\tilde{\Theta})$ which is a C^∞

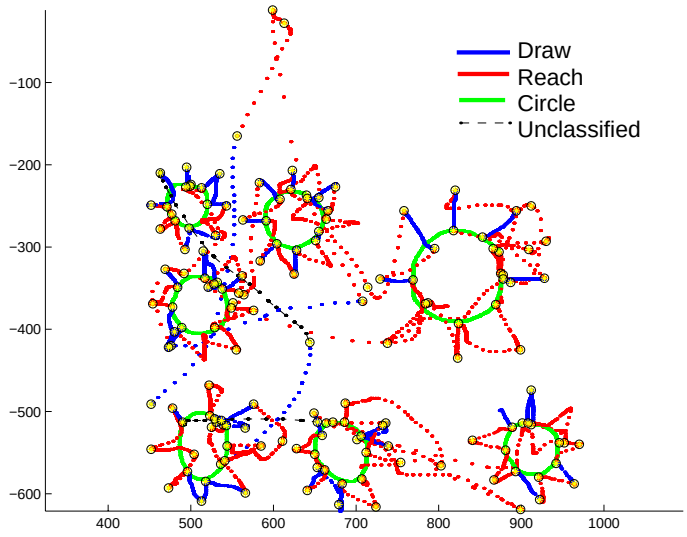

Figure 14: Segmentation results on suns of subject 1.

function. We study the dependence of $g(\tilde{\Theta})$ from the free variable $\tilde{\Theta}$ by computing its derivatives and applying Taylor's theorem. We then have $g(0) = 0$ since if $\tilde{\Theta} = 0$ and $\tilde{x}(0) = 0$ and A is a.s. system (45) stays at the equilibrium $\tilde{x} = 0$. Let us show $D_g(0) = 0$.

$$D_g(c) \cdot \tilde{\Theta} = \frac{d}{ds} g(c + s\tilde{\Theta})|_{s=0} \tag{46}$$

and substituting in the expression of g in place of $\tilde{x}$ its expression

$$\tilde{x}(t) = \int_0^t e^{A(t-\alpha)} \tilde{\Theta} \hat{\overline{x}}(\alpha) d\alpha$$

and computing (46) we obtain

$$D_g(c) \cdot \tilde{\Theta} = \frac{1}{\tau - t_0} \int_{t_0}^{\tau} \int_0^t \int_0^t [\hat{\overline{x}}^T(\sigma)^T c^T e^{(A^T(t-\sigma)+A(t-\alpha))} \tilde{\Theta} \hat{\overline{x}}(\alpha) + \hat{\overline{x}}^T(\sigma)^T \tilde{\Theta}^T e^{(A^T(t-\sigma)+A(t-\alpha))} c \hat{\overline{x}}(\alpha)] d\alpha d\sigma dt$$

which is clearly zero if $c = 0$. As far as $D_g^2(0)$ is concerned it is not zero since

$$D_g^2(c) \cdot (\tilde{\Theta}, \tilde{\Theta}) = \frac{d}{dt_2} \frac{d}{dt_1} [g(c + t_1\tilde{\Theta} + t_2\tilde{\Theta})]|_{t_1=t_2=0}$$

which is computed as done before and it turns out to be

$$D_g^2(c) \cdot (\tilde{\Theta}, \tilde{\Theta}) = \frac{1}{\tau - t_0} \int_{t_0}^{\tau} \int_0^t \int_0^t [\hat{\overline{x}}^T(\sigma)^T \tilde{\Theta}^T e^{(A^T(t-\sigma)+A(t-\alpha))} \tilde{\Theta} \hat{\overline{x}}(\alpha) + \hat{\overline{x}}^T(\sigma)^T \tilde{\Theta}^T e^{(A^T(t-\sigma)+A(t-\alpha))} \tilde{\Theta} \hat{\overline{x}}(\alpha)] d\alpha d\sigma dt$$

which is not zero for $c = 0$. We then identify the space $\mathbb{R}^{n \times n+1}$ with $\mathbb{R}^{n(n+1)}$ and associate to the matrix $\tilde{\Theta} \in \mathbb{R}^{n \times n+1}$ the vector $\tilde{\theta} \in \mathbb{R}^{n(n+1)}$. In particular let E be a compact set in $\mathbb{R}^{n(n+1)}$ such that for $\tau \in [\tau_i, \tau_{i+1}]$ $\tilde{\theta}(\tau) \in E$. For $\tilde{\theta}(\tau) \in E$ by Taylor's theorem we have

$$g(\tilde{\Theta}) = g(\tilde{\theta}) = g(0) + d_g(0) \cdot \tilde{\theta} + \frac{1}{2} d_g^2(c) \cdot (\tilde{\theta}, \tilde{\theta})$$

for $c = \alpha\tilde{\theta}$, $\alpha \in [0,1]$, where the first two terms are zero and the last one is zero only if $\tilde{\theta}$ is zero as we showed earlier. Since for each $\tilde{\theta} \in E$ there is a c such that the above expression is verified we can consider the smallest eigenvalue of $\frac{1}{2}d_g^2(c)$ as $\tilde{\theta}$ varies in E, then

$$g(\tilde{\Theta}) = g(\tilde{\theta}) \geq k_2\|\tilde{\theta}\|^2 = k_2 \mathrm{tr}(\tilde{\Theta}^T\tilde{\Theta}) = k_2\|\tilde{\Theta}\|^2 \tag{47}$$

where $k_2 = \min_{\tilde{\theta}\in E} \lambda_{min}(d_g^2(c))$. Since for $\tau > \tau_i$ $e_a(\tau) = \frac{1}{\tau - t_0}\int_{t_0}^{\tau} \tilde{x}^T(t)\tilde{x}(t)dt \geq \frac{1}{\tau - t_0}\int_{t_0}^{\tau_i} \tilde{x}^T(t)\tilde{x}(t)dt$, we have by (47) for $\tau > \tau_i$ $e_a(\tau) = g(\tilde{\Theta}) \geq k_2\|\Theta_i - \hat{\Theta}\| = \|(A_i|b_i) - (\hat{A}|\hat{b})\|$, which completes the proof. □

Proof of Lemma 4.3

Proof. Denote $(\tilde{A}|\tilde{b}) := \tilde{\Theta} = (A|b) - (A_1|b_1)$ and construct the error system

$$\dot{e} = \dot{x} - \dot{z} = Ax + b - A_1 z - b_1 - d(t)$$

and letting $A = A_1 + \tilde{A}$, $b = b_1 + \tilde{b}$ we have

$$\dot{e} = A_1 e + \tilde{\Theta}\overline{x} - d(t)$$

where $\overline{x} = (x^T, 1)^T$. Since A is Hurwitz consider Lyapunov function for the above error system $V = e^T P e$ with $P > 0$ such that $PA_1 + A_1 P + Q = 0$ for some $Q > 0$. then we have

$$\dot{V} = -e^T Q e - d^T P e - e^T P d + \overline{x}\tilde{\Theta}^T P e + e^T P \tilde{\Theta}\overline{x}$$

and substituting the proper upper bounds we find

$$\dot{V} \leq -\lambda\|e\|^2 + \|e\|[\lambda_p \overline{d} + 2\lambda_p \overline{\delta}\|\overline{x}\|]$$

where Λ_p is the largest eigenvalue of P and λ is the smallest absolute value of the eigenvalues of Q and completing the squares we finally obtain

$$\dot{V} \leq -0.5\lambda\|e\|^2 + \lambda_p^2\overline{d}^2 4/\lambda + 16\lambda_p^2\overline{\delta}^2/\lambda$$

therefore since the second and third terms are bounded $\|e\|^2$ will be bounded in a ball centered in the origin of radius $\frac{8\lambda_p^2\overline{d}^2 + 3\lambda_p^2\overline{\delta}^2\|\overline{x}\|^2}{\lambda^2}$ since $e(0) = 0$ by assumption. □

Proof of Lemma 4.4

Proof. Let us first prove relation (27). Then substitute in expression (15) for system (11) $x = x^0 + \tilde{x}$ where we denote with x^0 the state of the nominal system (12). Then from Lemma (4.3) we have $\|\tilde{x}\| \leq k_1\overline{\delta} + k_2\overline{d}$ for suitable positive constants k_1 and k_2. Then we can write also

$$\Phi := \int_{t_0}^{\tau} \overline{x}(t)^T\overline{x}(t)dt = \Phi^0 + \tilde{\Phi} \tag{48}$$

where $\Phi^0 = \int_{t_0}^{\tau} \overline{x}^0(t)^T\overline{x}^0(t)dt$ and $\tilde{\Phi}$ denotes the reminder which is directly proportional to δ and $d(t)$. Let $\hat{\Theta} = (\hat{A}, \hat{b})$ that is the estimate for (11) derived from (15), and $\hat{\Theta}^0 = (\hat{A}^0, \hat{b}^0)$ that is the estimate for the nominal system computed by letting in (15) $d = 0$ and $\delta = 0$. If $\tau \leq \tau_i$ the result is straightforward, let's then show it for $\tau > \tau_i$. By letting

$$v = [\int_{t_0}^{\tau_i} (A_i + \delta U_i|b_i)\overline{x}\overline{x}^T dt + \int_{\tau_i}^{\tau} (A_{i+1} + \delta U_{i+1}|b_{i+1})\overline{x}\overline{x}^T dt + \int_{t_0}^{\tau} d(t)\overline{x}^T dt]$$

and

$$w = [\int_{t_0}^{\tau_i} (A_i|b_i)\overline{x}^0(\overline{x}^0)^T dt + \int_{\tau_i}^{\tau} (A_{i+1}|b_{i+1})\overline{x}^0(\overline{x}^0)^T dt]$$

we have by (48) and (15)

$$\hat{\Theta} - \hat{\Theta}^0 = v[\Phi^0 + \tilde{\Phi}]^{-1} - w[\Phi^0]^{-1} = (v - w - (\Phi^0)^{-1}\tilde{\Phi}w)[\Phi]^{-1}$$

If we compute the term $v-w$ by rewriting again $x = x^0 + \tilde{x}$ we realize that it is proportional directly to noise and parameter uncertainty, so that all the terms in the right hand side are proportional to noise and parameter uncertainty. Further computation leads to an upper estimate for the norm of the right hand side:

$$\|\hat{A} - \hat{A}^0\| \leq \|\hat{\Theta} - \hat{\Theta}^0\| \leq k_p(\overline{d} + \overline{d}^2 + \overline{d}^3 + \overline{\delta} + \overline{\delta}^2 + \overline{\delta}^3) \ . \tag{49}$$

Letting $e_p^0(\tau) = \|\hat{A}^0 - A_i\|$ we have by the definition of parametric error (17) $\|\hat{A}^0 - A_i\| < \|\hat{A}^0 - A_j\|$ for any $j \neq i$ and letting $e_p(\tau) = \|\hat{A} - A_k\|$ for some k we have again by definition (17) that $\|\hat{A} - A_k\| < \|\hat{A} - A_j\|$ for any $j \neq k$. Therefore with $\hat{A} = \hat{A}^0 + \tilde{A}$ we have

$$e_p(\tau) = \|\hat{A} - A_k\| < \|\hat{A} - A_i\| = \|\hat{A}^0 - A_i + \tilde{A}\| \leq \|\hat{A}^0 - A_i\| + \|\tilde{A}\| = e_p^0(\tau) + \|\tilde{A}\|$$

which gives the right side of (27). For the left side we have

$$\|\hat{A} - A_k\| = \|\hat{A}^0 - A_k + \tilde{A}\| \geq \|\hat{A}^0 - A_k\| - \|\tilde{A}\| > \|\hat{A}^0 - A_i\| - \|\tilde{A}\| = e_p^0(\tau) - \|\tilde{A}\|$$

which with (49) concludes the proof of (27).

For proving (28), let $\tilde{x} = x - \hat{x} = x - x^0 + x^0 - \hat{x}^0 + \hat{x}^0 - \hat{x}$ in the expression of $e_a(\tau)$ given by (18) for system (11), where $\hat{x}^0$ is being generated by system (16) with $\hat{A}^0$ and $\hat{b}^0$ obtained from (15) with $d = 0$ and $\delta = 0$. Therefore letting $\tilde{x}^0 = x^0 - \hat{x}^0$, we have

$$e_a(\tau) = \frac{1}{\tau - t_0}\int_{t_0}^{\tau} (\tilde{x}^0(t))^T\tilde{x}^0(t)dt + \frac{1}{\tau - t_0}\int_{t_0}^{\tau} [2(\tilde{x}^0(t))^T(x(t) - x^0(t))$$

$$+2(\tilde{x}^0(t))^T(\hat{x}(t) - \hat{x}^0(t)) + 2(x(t) - x^0(t))^T(\hat{x}(t) - \hat{x}^0(t))$$

$$+(x(t) - x^0(t))^T(x(t) - x^0(t)) + (\hat{x}(t) - \hat{x}^0(t))^T(\hat{x}(t) - \hat{x}^0(t))]dt$$

where the first integral is $e_a^0(\tau)$. By Lemma 4.3 we have $\|x - x^0\| \leq k_1\overline{\delta} + k_2\overline{d}$ and $\|\hat{x}^0 - \hat{x}\| \leq k_3\|\hat{\Theta} - \hat{\Theta}^0\|$, where $\|\hat{\Theta} - \hat{\Theta}^0\| \leq k_p(\overline{d} + \overline{d}^2 + \overline{d}^3 + \overline{\delta} + \overline{\delta}^2 + \overline{\delta}^3)$ by (49, so that the second integral is directly proportional to noise and parameter uncertainty. Further computation leads to an upper bound for such integral of the form $\varepsilon = k_a(\overline{d} + \overline{d}^2 + \overline{d}^3 + \overline{d}^4 + \overline{\delta} + \overline{\delta}^2 + \overline{\delta}^3 + \overline{\delta}^4 + \overline{\delta}^6)$ which leads to expression (28). □

Proof of Lemma 4.5

Proof. Given a system of the form of one of (11) we can always rewrite it as

$$\dot{x} = Ax + b + w(t)$$

where $w(t)$ is including also the part deriving from δ. We can rewrite the solution of such a system as

$$x(t) = e^{At}x(0) + A^{-1}e^{At}b - A^{-1}b + w_1(t)$$

where w_1 is the contribution due to noise $w(t)$. Therefore

$$\dot{x} = Ae^{At}x(0) + e^{At}b + \overline{w}(t) = \dot{x}^0(t) + \overline{w}(t) \tag{50}$$

Then we compute $\dot{x}_{av}(\tau^-)$ and $\dot{x}_{av}(\tau^+)$. We rewrite $\dot{x}$ as in (50) with A_1, b_1 and x_{01} in place of A, b and $x(0)$ for computing $\dot{x}_{av}(\tau^-)$ for $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i$, and with A_2, b_2 and $x_{02}(0)$ in place of A, b and $x(0)$ for computing $\dot{x}_{av}(\tau^+)$ for $\tau \leq \tau_i - \Delta\tau$ or $\tau \geq \tau_i$. The situation is described in Figure 15. Then by integration we have for $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i$

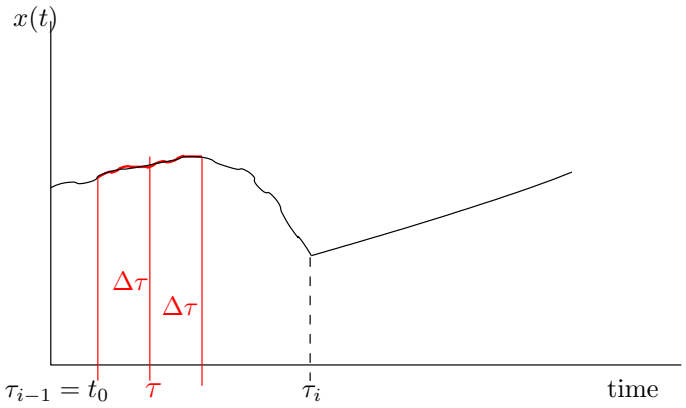

Figure 15: Intuitive picture in 2D space of sets $\mathcal{C}^j$ and $\mathcal{C}^k$.

$$\dot{x}_{av}(\tau^-) = (\frac{I - e^{-A_1\Delta\tau}}{\Delta\tau})e^{A_1\tau}x_{10} + A_1^{-1}(\frac{I - e^{-A_1\Delta\tau}}{\Delta\tau})e^{A_1\tau}b_1 + \overline{w}_1 = \dot{x}^0_{av}(\tau^-) + \overline{w}_1 \ . \tag{51}$$

For $\tau \leq \tau_i - \Delta\tau$ or $\tau \geq \tau_i$ we have

$$\dot{x}_{av}(\tau^+) = (\frac{e^{A_2\Delta\tau} - I}{\Delta\tau})e^{A_2\tau}x_{20} + A_2^{-1}(\frac{e^{A_2\Delta\tau} - I}{\Delta\tau})e^{A_2\tau}b_2 + \overline{w}_2 = \dot{x}^0_{av}(\tau^+) + \overline{w}_2 \tag{52}$$

where $\dot{x}^0_{av}(\tau^-)$ and $\dot{x}^0_{av}(\tau^+)$ are the perturbation free terms. Let

$$\gamma = \frac{\dot{x}_{av}(\tau^-)^T\dot{x}_{av}(\tau^+)}{\|\dot{x}_{av}(\tau^-)\|\|\dot{x}_{av}(\tau^+)\|}$$

which by (52), (51) and by letting $M_1 = \frac{I-e^{-A_1\Delta\tau}}{\Delta\tau}$ and $M_2 = \frac{e^{A_2\Delta\tau}-I}{\Delta\tau}$ becomes

$$\gamma = \frac{(M_1e^{A_1\tau}x_{01} + A_1^{-1}M_1e^{A_1\tau}b_1 + \overline{w}_1)^T(M_2e^{A_2\tau}x_{02} + A_2^{-1}M_2e^{A_2\tau}b_2 + \overline{w}_2)}{\|M_1e^{A_1\tau}x_{01} + A_1^{-1}M_1e^{A_1\tau}b_1 + \overline{w}_1\|\|M_2e^{A_2\tau}x_{02} + A_2^{-1}M_2e^{A_2\tau}b_2 + \overline{w}_2\|} \ .$$

We first isolate in this expression the contribution due to noise. Proceeding with standard arguments we obtain

$$\gamma \leq \gamma_0(1 + \beta) + \beta \tag{53}$$
$$\gamma \geq \gamma_0(1 - \beta) - \beta \tag{54}$$
$$\tag{55}$$

where

$$\gamma_0 = \frac{(M_1e^{A_1\tau}x_{01} + A_1^{-1}M_1e^{A_1\tau}b_1)^T(M_2e^{A_2\tau}x_{02} + A_2^{-1}M_2e^{A_2\tau}b_2)}{\|M_1e^{A_1\tau}x_{01} + A_1^{-1}M_1e^{A_1\tau}b_1\|\|M_2e^{A_2\tau}x_{02} + A_2^{-1}M_2e^{A_2\tau}b_2\|} \tag{56}$$

and

$$\beta = \overline{w}\frac{\|(M_1e^{A_1\tau}x_{01} + A_1^{-1}M_1e^{A_1\tau}b_1)\| + \|M_2e^{A_2\tau}x_{02} + A_2^{-1}M_2e^{A_2\tau}b_2\| + \overline{w}}{\|M_1e^{A_1\tau}x_{01} + A_1^{-1}M_1e^{A_1\tau}b_1 + \overline{w}_1\|\|M_2e^{A_2\tau}x_{02} + A_2^{-1}M_2e^{A_2\tau}b_2 + \overline{w}_2\|}$$

where $\overline{w}$ is an upper bound on $\overline{w}_1$ and $\overline{w}_2$ and it is proportional to $\overline{\delta}$ and $\overline{d}$ so that $\beta \leq k(\overline{\delta}+\overline{\delta}^2+\overline{d}+\overline{d}^2)$ for a suitable constant k. We proceed by finding a lower bound for γ_0 for $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau$ and an upper bound for $\tau = \tau_i$. Consider the case $\tau_{i-1} + \Delta\tau \leq \tau \leq \tau_i - \Delta\tau$ first. Then we have

$A_1 = A_2 = A$, $b_1 = b_2 = b$ and $x_{01} = x_{02} = x_0$ and $M_2 = e^{A\Delta\tau}M_1$, so that letting $M_2 = M_1 + \tilde{M}_1$, with $\tilde{M}_1 = e^{A\Delta\tau} - I$ which is going to zero as $\Delta\tau$ goes to zero, we find

$$\gamma_0 = \frac{(M_1 e^{A\tau}x_0 + A^{-1}M_1 e^{A\tau}b)^T[(M_1 e^{A\tau}x_0 + A^{-1}M_1 e^{A\tau}b) + (\tilde{M}_1 e^{A\tau}x_0 + A^{-1}\tilde{M}_1 e^{A\tau}b)]}{\|M_1 e^{A\tau}x_0 + A^{-1}M_1 e^{A\tau}b\|\|(M_1 e^{A\tau}x_0 + A^{-1}M_1 e^{A\tau}b) + (\tilde{M}_1 e^{A\tau}x_0 + A^{-1}\tilde{M}_1 e^{A\tau}b)\|} .$$

By minorating the numerator and majorating the denominator and adding and subtracting one, we obtain

$$\gamma_0 \geq 1 - \frac{2\|\tilde{M}_1 e^{A\tau}x_0 + A^{-1}\tilde{M}_1 e^{A\tau}b\|}{\|M_1 e^{A\tau}x_0 + A^{-1}M_1 e^{A\tau}b\|} = 1 - \frac{2\|\tilde{M}_1 e^{A\tau}x_0 + A^{-1}\tilde{M}_1 e^{A\tau}b\|}{\dot{x}^0_{av}(\tau^-)} \tag{57}$$

which leads to

$$\gamma_0 \geq 1 - k(1 - e^{\lambda\Delta\tau})$$

where λ is the eigenvalue with smallest real part and k is a suitable constant. Then from (54) we derive

$$\gamma \geq [1 - k(1 - e^{\lambda\Delta\tau})](1 - \beta) - \beta$$

and asking for example to have $\gamma \geq 1 - (k+2)\beta$, by equating $[1 - k(1 - e^{\lambda\Delta\tau})](1 - \beta) - \beta = 1 - (k+2)\beta$ we finally get a possible value for $\Delta\tau$:

$$\Delta\tau = -\frac{1}{\lambda}\ln\left(\frac{1 - 2\beta}{1 - \beta}\right)$$

which is a choice of $\Delta\tau$ which goes to zero when $\beta \to 0$ and it is such that $\mathrm{Tr}(\tau) \leq (k+2)\beta$. Then we have proved (30).

The proof of (31) proceeds analogously: let first consider the expression for γ_0 and find an upper bound for it when $\tau = \tau_i$, and then combine it with (53) to obtain the result. In this case $A_1 \neq A_2$, $b_1 \neq b_2$ and $x_{10} \neq x_{20}$, so let in expression (56) $M_1 = A_1 + \tilde{M}_1$ with $\tilde{M}_1 = \frac{I - e^{-A_1\Delta\tau} - A_1\Delta\tau}{\Delta\tau}$ and $M_2 = A_2 + \tilde{M}_2$ with $\tilde{M}_2 = \frac{e^{A_2\Delta\tau} - A_1\Delta\tau - I}{\Delta\tau}$, then proceeding with the same arguments as before and isolating the part due to tilde terms we obtain

$$\gamma_0 \leq \frac{(A_1 e^{A_1\tau}x_{01} + e^{A_1\tau}b_1)^T(A_2 e^{A_2\tau}x_{02} + e^{A_2\tau}b_2)}{\|A_1 e^{A_1\tau}x_{01} + e^{A_1\tau}b_1\|\|A_2 e^{A_2\tau}x_{02} + e^{A_2\tau}b_2\|}(1 + \alpha) + \alpha$$

where

$$\alpha = k\frac{1 - e^{-\lambda\Delta\tau} - \lambda\Delta\tau}{\Delta\tau}$$

where $\frac{1 - e^{-\lambda\Delta\tau} - \lambda\Delta\tau}{\Delta\tau}$ is the largest among the eigenvalue of $\tilde{M}_1$ and $\tilde{M}_2$ and it is going to zero as $\Delta\tau$ goes to zero and k is a suitable bound independent on $\Delta\tau$. Then from expression (53) we have $\gamma \leq \gamma_0(1 + \beta) + \beta$ and also

$$\frac{(A_1 e^{A_1\tau}x_{01} + e^{A_1\tau}b_1)^T(A_2 e^{A_2\tau}x_{02} + e^{A_2\tau}b_2)}{\|A_1 e^{A_1\tau}x_{01} + e^{A_1\tau}b_1\|\|A_2 e^{A_2\tau}x_{02} + e^{A_2\tau}b_2\|} = \frac{\dot{x}^0(\tau_i^-)^T\dot{x}^0(\tau_i^+)}{\|\dot{x}^0(\tau_i^-)\|\|\dot{x}^0(\tau_i^+)\|} \leq \rho_0$$

therefore

$$\gamma \leq [\rho_0(1 + \alpha) + \alpha](1 + \beta) + \beta$$

so that

$$\mathrm{Tr}(\tau_i) = \frac{1 - \gamma}{2} \geq \frac{1 - \rho_0 - \varphi}{2}$$

with $\varphi = \rho_0\alpha + \alpha + \rho_0\beta + \rho_0\alpha\beta + \alpha\beta + \alpha$ which is going to zero as perturbation (and therefore $\Delta\tau$) is going to zero. □

References

Bishop, C.M. (1995). *Neural Networks for Pattern Recognition*. Clarendon. Oxford.

Bizzi, E. and F.A. Mussa-Ivaldi (1999). Toward a neurobiology of coordinate transformations. *New Cog. Neuroscience* MIT Press, Cambridge, MA, 489–500.

Bregler, C. and J. Malik (1997). Learning and recognizing human dynamics in video sequences. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition*. Puerto Rico. pp. 568–674.

Collins, R.T., A. J. Lipton and T. Kanade (2000). Introduction to the special section on video surveillance. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **22**, 745–746.

Flanagan, J.R. and A.M. Wing (1997). The role of internal models in motion planning and control: evidence from grip force adjustments during movements of hand-held loads. *The Journal of Neuroscience* **17**, 1519–1528.

Goncalves, L., E. Di Bernardo and P. Perona (1998). Reach out and touch space (motion learning). In: *Proc. of the Third International Conference on Automatic Face and Gesture Recognition*. Nara, Japan. pp. 234–239.

Goncalves, L., E. Di Bernardo, E. Ursella and P. Perona (1995). Monocular tracking for human arm in 3d. In: *Proc. of the 7th Int.conference on Computer Vision, ICCV*. pp. 764–770.

Gustafsson, F. (2000). *Adaptive Filtering and Change Detection*. John Wiley & Sons.

Kawato, M. (1999). Internal models for motor control and trajectory planning. *Current Opinion in Neurobiology* **9**, 718–727.

Laptev, I. and T. Lindeberg (2001). Tracking of multi-state hand models using particle filtering and a hierarchy of multi-scale image features. In: *IEEE Workshop on Scale-Space and Morphology*. Vancouver, Canada. pp. 63–74.

Lavielle, M. (1998). Optimal segmentation of random processes. *IEEE Trans. on Signal Processing* **46**, 1365–1373.

Ljung, L. (1999). *System Identification*. Prentice Hall. New Jersey.

Munich, M.E. and P. Perona (1996). Visual input for pen-based computers. In: *Proc. of the 13th Int.conference on Pattern Recognition, ICPR*.

Mussa-Ivaldi, F.A., S.F. Giszter and E. Bizzi (1994). Linear combinations of primitives in vertebrate motor control. *Proc. of the National Academy of Science* **91**, 7534–7538.

Pedotti, A., P. Crenna, A. Deat, C. Frigo and J. Massion (1989). Postural synergies in axial movements: short and long term adaptation. *Exp. Brain Res. 74* pp. 3–10.

Rabiner, L.R. and B.H. Juang (1993). *Fundamentals of Speech Recognition*. Prentice Hall. Englewood Cliffs, New Jersey.

Söderström, T. and P. Stoica (1989). *System Identification*. Prentice Hall. Hemel Hempstead.

Silva, F., L. Velho, P. Cavalcanti and J. Gomes (1997). A new interface paradigm for motion capture based animation. In: *Proc. of 10th Brazilian Symposium of Computer Graphics and Image Processing*. pp. 49–56.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer Verlag.

Waldherr, S., S. Thurn, R. Romero and D. Margaritis (1998). Template-based recognition of pose and motion gestures on a mobile robot. In: *Proc. of the AAAI 15th National Conference on Artificial Intelligence*. pp. 977–982.

Wellner, P. (1991). The digital desk calculator: Tactile manipulator on a desk top display. In: *Proc. of the ACM Symposium on User Interface and Technology*. Hilton Head. pp. 27–33.

Willsky, A.S. and H.L. Jones (1976). A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *IEEE Trans. on Automatic Control* **21**, 108–112.

Wilson, A. and A. Bobick (1995). Learning visual behavior for gestures analysis. In: *Proc. of IEEE Symposium on Computer Vision*. Coral Gables, FL. pp. 229–234.

Yacoob, Y. and L. Davis (1996). Recognizing human facial expressions from long image sequences using optical flow. *IEEE Trans. on Pattern Analysis and Machine Intelligence 18(6)* pp. 636–642.

Zordan, V.B. and J.K. Hodgins (1999). Tracking and modifying upper-body human motion data with dynamic simulation. In: *Computer Animation and Simulation*. pp. 13–22.