

A Switching Regime Model for the EMG-Based Control of a Robot Arm

Panagiotis K. Artemiadis and Kostas J. Kyriakopoulos

Abstract—Human–robot control interfaces have received increased attention during the last decades. These interfaces increasingly use signals coming directly from humans since there is a strong necessity for simple and natural control interfaces. In this paper, electromyographic (EMG) signals from the muscles of the human upper limb are used as the control interface between the user and a robot arm. A switching regime model is used to decode the EMG activity of 11 muscles to a continuous representation of arm motion in the 3-D space. The switching regime model is used to overcome the main difficulties of the EMG-based control systems, i.e., the nonlinearity of the relationship between the EMG recordings and the arm motion, as well as the nonstationarity of EMG signals with respect to time. The proposed interface allows the user to control in real time an anthropomorphic robot arm in the 3-D space. The efficiency of the method is assessed through real-time experiments of four persons performing random arm motions.

Index Terms—Electromyographic (EMG) control, neuro-robotics, switching model.

I. INTRODUCTION

ROBOTS CAME TO light approximately 50 years ago. However, the way humans can interface with and finally control robots is still an important issue. The human–robot interface plays a role of the utmost significance, particularly since the use of robots is increasingly widening to everyday life tasks (e.g., service robots, robots for clinical applications). A large number of interfaces have been proposed in previous works [1]–[3]. Most of the previous developments propose complex mechanisms or systems of sensors, while, in most cases, the user should be trained to map his/her action (i.e., 3-D motion of a joystick or a haptic device) to the desired motion for the robot. In this paper, a new means of control interface is proposed, in which the user performs natural motions with his/her upper limb. Surface electrodes used for recording the

electromyographic (EMG) activity of the muscles of the upper limb were placed on the user’s skin. The recorded muscle activity was transformed to kinematic variables that were used to control the robot arm. Since in this study, an anthropomorphic robot arm was used, the user did not have to be acquainted with the interface mapping since natural arm motions sufficed to directly control the robot arm.

EMG signals have often been used as control interfaces for robotic devices. However, since the musculoskeletal system is very complex and the relationship of the EMG signals and arm motion is highly nonlinear [4], in most cases, only discrete control has been realized, focusing only, for example, at the directional control of robotic wrists [5] or at the control of multifingered robot hands to a limited number of discrete postures [6]–[10]. Arm exoskeletons [11] and arm power amplifiers [12] have used EMG signals as control interface in the past. However, most of the previous works decode only finite arm or hand postures from EMG signals [13]. Controlling a robot by using only finite postures can cause many problems regarding smoothness of motion though, particularly in cases where the robot performs everyday life tasks. Moreover, from a teleoperation point of view, a small number of finite commands or postures can critically limit the areas of application. Therefore, effectively interfacing a robot arm with a human entails the necessity of continuous and smooth control. An artificial neural network was used in [14] to estimate the continuous motion of fingers using EMG signals from the muscles of the forearm. However, in this paper, only one degree of freedom (DoF) per finger was decoded, and individual finger motion was assumed, while the method could not incorporate changes in EMG signals with respect to time, caused by muscle fatigue or electrode impedance changes. Continuous models have been built in the past to decode arm motion from EMG signals. The Hill-based muscle model [15], whose mathematical formulation can be found in [4], is more frequently used in the literature [16], [17]. However, only a few DoFs were analyzed (i.e., 1 or 2) since the nonlinearity of the model equations and the large number of the unknown parameters for each muscle make the analysis rather difficult.

Similarly, musculoskeletal models have been analyzed in the past [18]–[20], focusing on a small number of muscles and actuated DoFs. A neural network model was used for extracting continuous arm motion in the past using EMG signals [21]; however, the movements analyzed were restricted to single-joint isometric motions. Similarly in [22], one DoF robot arm was controlled using EMG signals. Therefore, random arm motions were never efficiently decoded through EMG signals for the scope of the EMG-based robot control.

Manuscript received May 28, 2009; revised October 19, 2009. This work was supported in part by the European Commission through contract Neurobotics (FP6-IST-001917) Project and in part by the European Union–European Social Fund (75%) and the Greek Ministry of Development–General Secretariat for Research and Technology (GSRT; 25%). This paper was recommended by Associate Editor D. Y. Lee.

P. K. Artemiadis is with the Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: partem@mit.edu).

K. J. Kyriakopoulos is with the Control Systems Laboratory, School of Mechanical Engineering, National Technical University of Athens, 157 80 Athens, Greece (e-mail: kkyria@mail.ntua.gr).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSMCB.2010.2045120

An important factor that is present in the EMG-based controlled systems, although never been investigated until now, is the fact that EMG signals change with respect to time.¹ These changes can be caused by muscle fatigue, changes in the level of muscle force production, sweat at the recording site, or small electrode movement with respect to its initial position [23]–[27]. However, all the algorithms that have been proposed use stationary models for translating EMG signals to motion. Therefore, time variation of EMG signals is not incorporated, making the aforementioned methods applicable only for short time periods.

In this paper, we propose a methodology for controlling an anthropomorphic robot arm using surface recordings from the muscles of the upper limb using a switching regime (SR) decoding model that is robust to time variation of EMG signals. Surface EMG electrodes were used to record from 11 muscles of the shoulder and the elbow. The system architecture was divided into two phases: the training and the real-time operation. During the training phase, the user was instructed to move his/her arm in random patterns with variable speed in the 3-D space. A position tracking system was used to record the arm motion during reaching, while EMG signals were recorded. Using the principal component analysis (PCA) method, the EMG recordings and the motion variables extracted from the position tracking system were represented in lower dimensional manifolds. Then, the low-dimensional embeddings of arm motion and EMG signals were used to train an SR decoding model. The switching variable of the proposed decoder was controlled through a Bayesian classifier of a set of time-varying EMG signal features, as well as the performed motion. As soon as the training phase had finished, the real-time operation phase commenced. A control law that utilized the decoder's motion estimates was applied to the robot arm actuators. In this phase, the user could teleoperate the robot arm in real time, while he could correct any possible robot deviations since he had visual contact with the robot. The efficacy of the proposed method was assessed through a large number of experiments, during which the users controlled the robot arm in performing random movements in the 3-D space.

The rest of this paper is organized as follows: The proposed system architecture is analyzed in Section II, the experiments are reported in Section III, while Section IV concludes this paper.

II. METHODS

A. Background and Problem Definition

1) *Data Acquisition and Processing*: The motion of the upper limb in the 3-D space was analyzed, although the wrist joint was not included for simplicity. Therefore, the shoulder and elbow joints were of interest. Three rotational DoFs were used to model the shoulder joint and one rotational DoF for the elbow joint. For the training of the proposed system, the motion of the upper limb should be recorded, and joint trajectories should be extracted. Therefore, to record the motion and then to

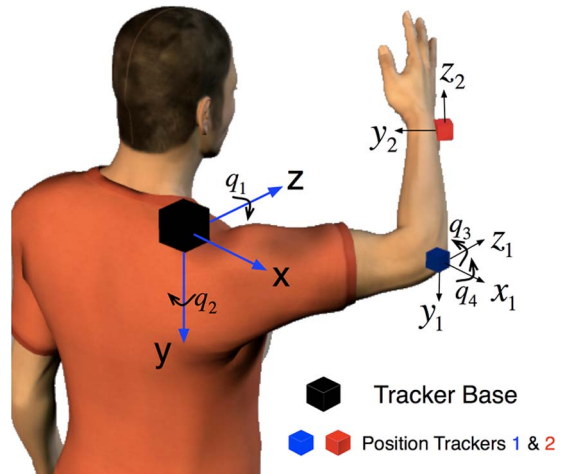


Fig. 1. User moves his arm in the 3-D space. Two position tracker measurements are used for computing the four joint angles. The tracker base reference system is placed on the shoulder. q_1 and q_2 jointly correspond to shoulder flexion–extension and adduction–abduction, q_3 corresponds to shoulder internal–external rotation, while q_4 corresponds to elbow flexion–extension.

extract the joint angles of the four modeled DoFs, a magnetic position tracking system was used. The system was equipped with two position trackers and a reference system, with respect to which the 3-D position of the trackers was provided. To compute the four joint angles, one position tracker was placed at the user's elbow joint and the other one at the wrist joint. The reference system was placed on the user's shoulder. The setup, as well as the four modeled DoFs, is shown in Fig. 1. Let $\mathbf{T}_1 = [x_1 \ y_1 \ z_1]^T$ and $\mathbf{T}_2 = [x_2 \ y_2 \ z_2]^T$ be the position of the trackers with respect to the tracker reference system. Let $q_1, q_2, q_3,$ and q_4 be the four joint angles modeled as shown in Fig. 1. Finally, by solving the inverse kinematic equations, the joint angles are given by

$$\begin{aligned} q_1 &= \arctan 2(\pm y_1, x_1) \\ q_2 &= \arctan 2\left(\pm \sqrt{x_1^2 + y_1^2}, z_1\right) \\ q_3 &= \arctan 2(\pm B_3, B_1) \\ q_4 &= \arctan 2\left(\pm \sqrt{B_1^2 + B_3^2}, -B_2 - L_1\right) \end{aligned} \quad (1)$$

where

$$\begin{aligned} B_1 &= x_2 \cos(q_1) \cos(q_2) + y_2 \sin(q_1) \cos(q_2) - z_2 \sin(q_2) \\ B_2 &= -x_2 \cos(q_1) \sin(q_2) - y_2 \sin(q_1) \sin(q_2) - z_2 \cos(q_2) \\ B_3 &= -x_2 \sin(q_1) + y_2 \cos(q_1) \end{aligned} \quad (2)$$

where L_1 is the length of the upper arm. The length of the upper arm can be computed from the distance of the first position tracker from the base reference system, i.e.,

$$L_1 = \|\mathbf{T}_1\| = \sqrt{x_1^2 + y_1^2 + z_1^2}. \quad (3)$$

Likewise, the length of the forearm L_2 can be computed from the distance between the two position trackers, i.e.,

$$L_2 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}. \quad (4)$$

¹Explicit time variation is meant here, i.e., during a course of arm movements, the same motions could result to different EMG signals.

One out of the multiple solutions given by (1) was selected for each joint angle at each time instance, based on the range of motion for each human joint; if that was not enough for solving the ambiguity, the solution selected was the one that was closer to the previous value computed.

The position tracking system provided the position vectors \mathbf{T}_1 and \mathbf{T}_2 at the frequency of 30 Hz. Using an antialiasing finite-impulse-response filter (low pass, order: 24, cutoff frequency: 100 Hz), these measurements were resampled at a frequency of 1 kHz to be consistent with the muscle activation sampling frequency.

EMG signals were recorded from 11 muscles that are mainly responsible for the analyzed motion: deltoid (anterior), deltoid (posterior), deltoid (middle), pectoralis major, teres major, pectoralis major (clavicular head), trapezius, biceps brachii, brachialis, brachioradialis, and triceps brachii. Finally, surface bipolar EMG electrodes used for recording were placed on the user's skin following the directions given in [28].

2) *System Requirements*: It is widely reported in the biomechanics literature that EMG signals are not stationary, in that some signal features vary with respect to time. In other words, EMG patterns for the same motion change with respect to time. The latter can be caused due to several reasons such as muscle fatigue, changes in the muscle contraction level, changes in electrode impedance, or changes in muscle recruitment. A robust method for decoding the EMG activity to motion should incorporate those changes in the signal space and appropriately adapt to those in real time. Moreover, the method should decode a continuous representation of motion to allow for its use in a robot control scheme. Finally, the decoding method should be easily trainable to each user since the musculoskeletal system, arm dynamics, and, consequently, EMG signals are different among subjects. Even within the same subject, repositioning the electrodes can require a new system training; therefore, the decoder architecture should be easy and fast enough in terms of training.

In the following, the proposed architecture will be presented, explicitly pointing out how the above requirements are met.

B. Time Variation of EMG

The time variation of EMG signals should be detected and analyzed in real time. After analyzing the data recorded during a training session, we computed a set of signal features that were observed to vary with respect to time, and after signal preprocessing, these signal features are listed below.

Integral of absolute value (IAV): The IAV of the EMG signal of one muscle was calculated by

$$\text{IAV} = \frac{1}{M} \sum_{i=1}^M |e_i| \quad (5)$$

where $|e_i|$ was the absolute value of the i th sample, and M was the number of samples in each segment. A raw signal was digitized at the frequency of 1 kHz and partitioned in overlapping segments (i.e., time bins) of 100 ms. Therefore, $M = 100$, while since the bins were overlapping, the

signal characteristic was computed at the same frequency with that of the acquisition (i.e., 1 kHz).

Zero crossing (ZC): ZC was the number of times that the signal passes the zero amplitude axis. It was calculated by

$$\text{ZC} = \sum_{i=1}^M \text{sgn}(-e_i e_{i+1}) \quad (6)$$

where

$$\text{sgn}(e) = \begin{cases} 1 & \text{if } e > 0 \\ 0 & \text{otherwise.} \end{cases} \quad (7)$$

Variance (VAR): The variance was a measure of the signal power and was calculated by

$$\text{VAR} = \frac{1}{M-1} \sum_{i=1}^M e_i^2. \quad (8)$$

Median frequency (MDF): The MDF was the frequency at which the cumulative power spectrum of the recorded signal was divided into two parts of equal power. It was mathematically described by the following:

$$\int_0^{\text{MDF}} P(\omega) d\omega = \int_{\text{MDF}}^{\infty} P(\omega) d\omega = \frac{1}{2} \int_0^{\infty} P(\omega) d\omega \quad (9)$$

where $P(\omega)$ was the power spectral density, and ω was the frequency of the signal. In our case, the MDF was computed for $0 < \omega < 500$ Hz.

The calculation of the previously defined signal features for the training period showed that there was significant variation in their values with respect to the experiment time. Similar behavior was reported in previous studies [29]–[31].

In the following, we will analyze the architecture of an SR decoding scheme that is able to estimate arm motion without being affected by the observed time-varying EMG-signal features.

C. SR Decoding Model

1) *Data Preprocessing*: During the training period, the EMG recordings from each muscle were preprocessed, i.e., full-wave-rectified, low-pass-filtered, and normalized to their maximum voluntary isometric contraction (MVC) value [4]. MVC values for all muscles were acquired using guidelines found in [28]. Adequate relax time (approximately 1 min) was provided to the users among the individual procedures for acquiring MVC values for each muscle. Then, linear dynamics were imposed to the resulted signals to model the neural-excitation dynamic behavior. The parameters for dynamic constants were taken from [4]. Afterward, the signals were represented into a low-dimensional space using the PCA algorithm. It was found that a 2-D space could represent most of the original high-dimensional data variance (more than 96%). The authors have used the dimensionality reduction for muscle activations in the past for planar movements of the arm [32]. Therefore, the details of the method application are omitted. Furthermore, the

dimensionality reduction technique will also be used for representing the arm motion in a low-dimensional space, revealing motion primitives that are extensively discussed in the literature [33], [34]. Therefore, by using the PCA algorithm, the analyzed four-DoF motion, described in joint angles (i.e., q_1, q_2, q_3 , and q_4), was represented into a 2-D space. Indeed, it was found that most of the original data variance (97%) was represented using a 2-D space. It must be noted that representing a 4-D space into two dimensions did not affect the available workspace or the analyzed DoFs. The number of DoFs analyzed remained the same (i.e., 4), and the available arm workspace was always the 3-D space. Using the dimensionality reduction technique, the same variability was represented into another space, taking advantage of any underlying covariance, without losing any dimension of the original data. Moreover, it must be noted that the PCA method was chosen as the method that is the easiest to implement for data compression and dimensionality reduction [35]. Other methods could have been used (i.e., independent component analysis, factor analysis, etc.); however, for data compression, the PCA method is more computationally effective. Furthermore, the PCA method is based on the notion of Gaussian distributions around the principal axes, which was a characteristic that helped the consecutive analysis.

2) *Decoding Model*: Having represented the muscle activations and the performed joint kinematics into two low-dimensional spaces, one could then build a model that would use the EMG low-dimensional embeddings to estimate performed motion. It is quite obvious that from a physiological point of view, a model that would describe the function of the skeletal muscles in actuating the human joints would be generally complex. Therefore, using such a model for real-time decoding would be problematic. For this reason, we adopted a more flexible decoding model in which we introduced “hidden” or “latent” variables we called \mathbf{x} . These hidden variables could model the unobserved intrinsic system states and, thus, facilitate the correlation between the low-dimensional embeddings of the muscles activations \mathbf{U} and the joint angles \mathbf{y} . Let $\mathbf{U}_t \in \mathbb{R}^2$ be the 2-D vector of the low-dimensional representation of the 11 muscle recordings, at time $t = kT$, $k = 1, \dots$, and $\mathbf{y}_t \in \mathbb{R}^2$ be the low-dimensional embedding of the arm joint angles at the same time instance. The model used for decoding the EMG activity to performed motion was defined as

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{U}_t + \mathbf{v}_t \\ \mathbf{y}_t &= \mathbf{C}\mathbf{x}_t + v_t \end{aligned} \quad (10)$$

where $\mathbf{x}_t \in \mathbb{R}^d$ is a hidden state vector, d is the dimension of this vector, and \mathbf{v}_t and v_t are the zero-mean Gaussian noise in process and observation equations, respectively, i.e., $\mathbf{v}_t \sim N(\mathbf{0}, \mathbf{W})$ and $v_t \sim N(0, \mathbf{Q})$, where $\mathbf{W} \in \mathbb{R}^d$ and $\mathbf{Q} \in \mathbb{R}^2$ are the covariance matrices of \mathbf{v}_t and v_t , respectively. Matrices $\mathbf{A}_{d \times d}$, $\mathbf{B}_{d \times 2}$, and $\mathbf{C}_{2 \times d}$ represented the dynamics of the hidden states, the relation between the low-D embeddings of muscle activation and the hidden state dynamics, and the relation of the hidden states to the output variables of the model, respectively. Details on the model structure can be found in [32]. The authors have used this kind of model in decoding EMG activity to motion in the past. However, the motion was restricted to a plane, using a small set of muscles, while the testing period

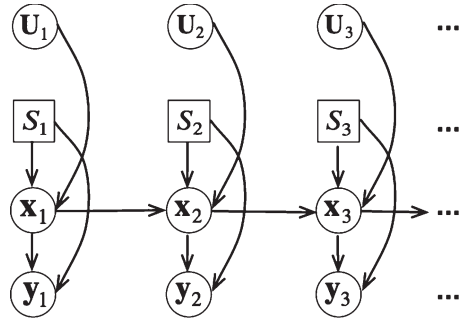


Fig. 2. Representation of the SR model using a directed graph. (Squares) Discrete variables. (Circles) Continuous variables. Arrows represent dependence of the variable pointed by the arrow, by the variable from which the arrow is originated. 1, 2, 3 are time indexes. \mathbf{U}_i , \mathbf{x}_i , and \mathbf{y}_i are vectors as defined in (10), $i = 1, 2, 3, \dots$. The bended arrow starting from \mathbf{U}_i represents the dependence of the hidden state vector \mathbf{x}_i to the input \mathbf{U}_i . This dependence is represented through the matrix \mathbf{B} in (10). Similarly, matrix \mathbf{A} is represented by the straight arrow connecting \mathbf{x}_{i-1} and \mathbf{x}_i , stating the dynamic behavior of the hidden state vector. Since the discrete switching variable S is controlling which model of the form (10) is going to be used at each time instance, it affects both \mathbf{x}_i and \mathbf{y}_i . The latter is represented by the arrows starting from S_i and pointing to \mathbf{x}_i and \mathbf{y}_i , respectively. Switching variable S has no Markovian dynamics by definition (see Section II-C); therefore, no arrows connect S_{i-1} with S_i , $i = 1, 2, 3, \dots$

was limited; thus, changes in the time-varying characteristics of EMG signals did not affect the decoder’s accuracy. In this paper, the motion of the arm was extended to the 3-D space, while the time variation of EMG signals was incorporated in the decoder. The latter is discussed below.

Our goal was to model the probabilistic relationship between the low-dimensional embeddings of muscle activations and the motion of the upper limb. To that end, we exploited an SR model that is illustrated in Fig. 2. In contrast to a simple linear decoding model of the form (10), the SR model introduces a discrete “switching” variable S , which corresponds to a set of models of the form (10). This switching variable took values from 1 to N , where N is the number of the models. Therefore, if $S = k$, $k = 1, \dots, N$, it means that the k th model was used for decoding. The main reasoning was that the observations (i.e., the EMG signals) and their relationship with arm motion may be described by different models represented by this switching variable. Those different models can represent, in a stochastic way, the relationship between the EMG signals and the arm motion, in various different situations, i.e., variation of EMG signals with respect to time due to muscle fatigue, sweat, electrode movement with respect to its initial position, etc.

In previous works on SR models [36], the switching variable S had Markovian dynamics, and the probability of each variable $P(S_m = k)$, $k = 1, \dots, N$ at each time instance m , affected the result. In our case, we introduced a winner-take-all rule, which actually selects one out of the N models to use at each time instance m , based on the probability of the switching variable S , $P(S_m = k)$, $k = 1, \dots, N$. Therefore, the j th model was selected if

$$j = \arg \max_{k=1, \dots, N} (P(S_m = k)). \quad (11)$$

Moreover, we defined that the switching variable S has no dynamics, i.e., $P(S_m = a_1 | S_{m-1} = a_2) = P(S_m = a_1)$,

$a_1, a_2 = 1, \dots, N$. This means that one model was used at each time, instead of a weighted sum of all the models [36], [37]. Moreover, the model used at each time instance was not affected by the selection of the model used in the previous time instance. Introducing dynamics in the switching variable has been argued to cause delayed switching among the models [36], which, in our case, can cause erroneous arm motion estimates.

D. EMG Time Variation and Switching Decoding

To incorporate in the decoding scheme the EMG signal variation, we related the switching variable S with the time-varying EMG signal features analyzed in Section II-B. However, EMG signal features are not only time-dependent but also related with the performed motion. Therefore, a feature vector \mathbf{F} was defined, which included the four aforementioned signal characteristics that were computed at each time bin for each muscle and the motion variables of the arm. As motion variables, the angular velocities \dot{q}_w , $w = 1, \dots, 4$, were used since it was shown from the data recordings that their dependence on the EMG signal features was stronger than the dependence of the joint angles.² Joint angular velocities were computed by the time differentiation of the joint angles computed from (1). The feature vector \mathbf{F} of each muscle i , at each time instance m , was given by

$$\mathbf{F}_m^{(i)} = \begin{bmatrix} \text{IAV}_m^{(i)} & \text{ZC}_m^{(i)} & \text{VAR}_m^{(i)} & \text{MDF}_m^{(i)} & \dots \\ \dot{q}_{1m} & \dot{q}_{2m} & \dot{q}_{3m} & \dot{q}_{4m} \end{bmatrix}^T$$

$$i = 1, \dots, 11, \quad m = 1, \dots \quad (12)$$

where time instances m and $m - 1$ are 1 ms away and correspond to the time instances that the signal features are computed, as defined in Section II-B.³

Since motion profiles were incorporated in our analysis, the training session included a large range of motion variability. Thus, the user was instructed to move his arm to all the possible ranges of motion for the shoulder and the elbow at various speeds.

As noted previously, we wanted to incorporate the time variation of the EMG signals in the decoding scheme. For this reason, we related the regime switching characteristic with the feature vector describing the time variation of the EMG signal features and the corresponding arm motion. In other words, we related the switching variable S with a Bayesian classifier applied on the feature vector $\mathbf{F}_m^{(i)}$ for each muscle i . The Bayesian classifier defined separate classes for each muscle, and then based on these classes, the decision about the switching variable S was made.

²Joint accelerations were also computed to investigate their correlation with EMG signal features. However, acceleration measurements were highly noisy due to double differentiation, while strong correlation with EMG signal features was not observed.

³Since the computation of the MDF requires a fast Fourier transform, which has high computational cost, a new MDF value is essentially computed every 100 ms. This value is used at the feature vector $\mathbf{F}_m^{(i)}$ for 100 times (i.e., until a new one is computed) without loss of generality since MDF is not expected to vary significantly in a period of 100 ms.

Let $\mathbf{f}^{(i)}$ be the set of the possible classes for muscle i as defined above, i.e.,

$$\mathbf{f}^{(i)} = \{f_1^{(i)}, f_2^{(i)}, \dots, f_{g_i}^{(i)}\} \quad (13)$$

where g_i is the number of the possible classes for muscle i . To decide the class for the muscle at each time instance m , according to the measured feature vector $\mathbf{F}_m^{(i)}$, we computed the conditional probability of the muscle being at the class $f_j^{(i)}$, $j = 1, \dots, g_i$, given the feature vector $\mathbf{F}_m^{(i)}$, i.e.,

$$P\left(f_j^{(i)} | \mathbf{F}_m^{(i)}\right). \quad (14)$$

This was done by using the Bayes theorem [38], which, in our case, was described by the following:

$$P\left(f_j^{(i)} | \mathbf{F}_m^{(i)}\right) = \frac{p\left(\mathbf{F}_m^{(i)} | f_j^{(i)}\right) P\left(f_j^{(i)}\right)}{p\left(\mathbf{F}_m^{(i)}\right)}, \quad j = 1, \dots, g_i \quad (15)$$

where $p\left(\mathbf{F}_m^{(i)} | f_j^{(i)}\right)$ is the probability density function (pdf) of the feature vector $\mathbf{F}_m^{(i)}$ given the class $f_j^{(i)}$, $P\left(f_j^{(i)}\right)$ is the prior probability of the class being $f_j^{(i)}$, and

$$p\left(\mathbf{F}_m^{(i)}\right) = \sum_{j=1}^{g_i} p\left(\mathbf{F}_m^{(i)} | f_j^{(i)}\right) P\left(f_j^{(i)}\right) \quad (16)$$

is the evidence factor that can be considered as a scale factor that guarantees that the posterior probabilities sum to one. The g_i classes for each muscle i are considered equally likely to happen, i.e.,

$$P\left(f_{(1)}^{(i)}\right) = P\left(f_{(2)}^{(i)}\right) = \dots = P\left(f_{(g_i)}^{(i)}\right) \quad (17)$$

and since

$$\sum_{r=1}^{g_i} P\left(f_{(r)}^{(i)}\right) = 1 \quad (18)$$

it is

$$P\left(f_{(1)}^{(i)}\right) = P\left(f_{(2)}^{(i)}\right) = \dots = P\left(f_{(g_i)}^{(i)}\right) = \frac{1}{g_i}. \quad (19)$$

To decide the class of the muscle i at each time instance m , related to the recorded time-varying signal features, we used (15) to compute the probability of being at the class $f_j^{(i)}$ for each j , $j = 1, \dots, g_i$, given the feature vector $\mathbf{F}_m^{(i)}$. Then, the class with the largest probability was assigned for muscle i . This is done at every time step m using the new feature vector $\mathbf{F}_m^{(i)}$. However, the pdf of the feature vector $\mathbf{F}_m^{(i)}$ given the class $f_j^{(i)}$, i.e., $p\left(\mathbf{F}_m^{(i)} | f_j^{(i)}\right)$, the so-called *likelihood* term, had to be computed. This was achieved by using the data collected through the training period. Since there was no specific relation between the coefficients of the feature vector, a flexible method of modeling, called finite mixture models, was used.

Finite mixtures of distributions provide a mathematical-based approach to the statistical modeling of a wide variety of random phenomena [39]. In our case, where more than

one component (i.e., features) is to be modeled, which is not independent, a multivariate mixture model was used. Moreover, a common assumption, in practice, is to take the component densities to be Gaussian. Therefore, a multivariate Gaussian mixture model (GMM) was used for modeling the multivariate density of the feature vector $\mathbf{F}_m^{(i)}$. Let $\mathbf{F}_m^{(i)}$ be the observed feature vector of muscle i at time instance m during the training procedure. The pdf of this can be modeled using a GMM that is defined by

$$p\left(\mathbf{F}_m^{(i)}\right) = \sum_{h=1}^{g_i} \pi_h \phi_h\left(\mathbf{F}_m^{(i)}, \mu_h, \Sigma_h\right) \quad (20)$$

where $\phi_h(\mathbf{F}_m^{(i)}, \mu_h, \Sigma_h)$ represents a multivariate Gaussian density function with μ_h as the mean vector, Σ_h is the respective covariance matrix, and $\boldsymbol{\pi} = [\pi_1 \dots \pi_{g_i}]^T$ is the vector of mixing proportions of the mixture, which sum to one i.e.,

$$\sum_{h=1}^{g_i} \pi_h = 1. \quad (21)$$

Using the training data collected, the parameters of the GMM, i.e., $\boldsymbol{\pi}$, $\boldsymbol{\mu}$, and $\boldsymbol{\Sigma}$, were fitted using the expectation-minimization algorithm [39]. The number of the Gaussian components g_i was determined by using the Akaike criterion, which is a widely used measure of goodness of fit of an estimated statistical model. For further information, the reader should refer to [40].

In our case, the mixture components were used for clustering the signal characteristics into the aforementioned classes. This was done once the mixture models had been fitted using probabilistic clustering of the data into g_i clusters that were obtained in terms of the fitted posterior probabilities of component membership for the data. An outright assignment of the data into g_i clusters was achieved by assigning each data point to the component that had the highest posterior probability of belonging to. For this purpose, we let $r(\mathbf{F}_m^{(i)})$ denote an allocation rule for assigning the feature vector $\mathbf{F}_m^{(i)}$ to one of the components of the mixture model, where $r(\mathbf{F}_m^{(i)}) = l$ implies that the observation was assigned to the l th component ($l = 1, \dots, g_i$). The optimal or Bayes rule $r_B(\mathbf{F}_m^{(i)})$ for the allocation of $\mathbf{F}_m^{(i)}$ was defined by

$$r_B\left(\mathbf{F}_m^{(i)}\right) = l \quad \text{if} \quad \psi_l\left(\mathbf{F}_m^{(i)}\right) \geq \psi_h\left(\mathbf{F}_m^{(i)}\right) \quad h = 1, \dots, g_i \quad (22)$$

where $\psi_l(\mathbf{F}_m^{(i)})$ is the posterior probability that the entity belongs to the l th component with $\mathbf{F}_m^{(i)}$ having been observed on it, and it was given by

$$\begin{aligned} \psi_l\left(\mathbf{F}_m^{(i)}\right) &= pr\left\{\text{entity} \in l\text{th component} \mid \mathbf{F}_m^{(i)}\right\} \\ &= \frac{\pi_l \phi_l\left(\mathbf{F}_m^{(i)}\right)}{\sum_{h=1}^{g_i} \pi_h \phi_h\left(\mathbf{F}_m^{(i)}\right)}. \end{aligned} \quad (23)$$

Relating the classes corresponding to the time variations of EMG signal characteristics to the g clusters is straightforward.

That is why the same notation g_i was used from the start, regarding the number of classes and the number of mixture components of the GMM fitted to the pdf of the muscle i .

Therefore, from the aforementioned analysis and after the training period, the class related to the recorded time-varying signal features was assigned to each muscle i at each time instance m using (15). For each muscle i , the feature vector $\mathbf{F}_m^{(i)}$ is computed. Then, for each of the classes j , $j = 1, \dots, g_i$, the conditional probability of the muscle belonging to the class $f_{(j)}^{(i)}$, given in (14), can be computed using (15), where

$$\begin{aligned} p\left(\mathbf{F}_m^{(i)} \mid f_{(j)}^{(i)}\right) &= \sum_{h=1}^{g_i} \pi_h \phi_h\left(\mathbf{F}_m^{(i)}, \mu_h, \Sigma_h\right) \\ P\left(f_{(1)}^{(i)}\right) &= P\left(f_{(2)}^{(i)}\right) = \dots = P\left(f_{(g_i)}^{(i)}\right) = \frac{1}{g_i} \\ p\left(\mathbf{F}_m^{(i)}\right) &= \sum_{j=1}^{g_i} p\left(\mathbf{F}_m^{(i)} \mid f_{(j)}^{(i)}\right) P\left(f_{(j)}^{(i)}\right). \end{aligned} \quad (24)$$

To this point, the probability of each muscle i belonging to each of the possible g_i classes at each time instance m , i.e., $P(f_{(j)}^{(i)} \mid \mathbf{F}_m^{(i)})$, was computed. Therefore, since muscle classification is considered independent among the muscles,⁴ the resulting probability of each class describing all the muscles, at a given time instance m , is given by the product of the probabilities of this class for all the muscles. Moreover, since the classification of the muscles will essentially control the switching of the SR model introduced above, we defined the following:

$$P(S_m = j) = \prod_{i=1}^{11} P\left(f_{(j)}^{(i)} \mid \mathbf{F}_m^{(i)}\right), \quad j = 1, \dots, g_i \quad (25)$$

which equalized the probability of the switching variable S of the SR model previously defined, with the probability of each class describing all the muscles. In other words, if a class j had a large probability of describing most of the muscles, then the probability of the switching variable S taking the value j , i.e., $P(S_m = j)$, would be large too. Consequently, by using rule (11), the final model j was selected to be used for decoding. It must be noted, that, this way, the number N of the possible values for the switching variable S is now defined equal to the number of the classes g_i , of each muscle i , which are also defined to be the same across the 11 muscles. This assumption does not affect anything presented so far since the final number of models used in the SR model is now directly controlled by the classification of the pdf of the EMG-signal feature distribution, which perfectly coincides with the rationale of using the SR model, i.e., the description of EMG-to-motion relation, in different situations, caused by the time variation of the EMG signal features. This resulted to a robust decoding

⁴Each muscle classification regarding the time-varying features of their activation is considered independent since not all muscles contribute equally to all motions. Therefore, there could be muscles which develop fatigue (or any other phenomenon that causes signal feature change) in different time instances than other muscles. Consequently, we assume independence across muscles, regarding their time-varying signal feature classification.

method, the accuracy of which was not affected by the changes of EMG time-varying characteristics. The analyzed architecture was then used to control the 3-D motion of a robot arm in real time using only EMG recordings.

E. Robot Control

A seven-DoF anthropomorphic robot arm (PA-10, Mitsubishi Heavy Industries) is used. Only four DoFs of the robot are actuated (joints of the shoulder and the elbow), while the others are kept fixed at zero position via electromechanical brakes. The arm is horizontally mounted to mimic the human arm. The robot motors were torque controlled. To control the robot arm using the desired joint angle vector \mathbf{q}_d ,⁵ an inverse dynamic controller was used, defined by

$$\tau = \mathbf{I}(\mathbf{q}_r)(\ddot{\mathbf{q}}_d + \mathbf{K}_v\dot{\mathbf{e}} + \mathbf{K}_p\mathbf{e}) + \mathbf{G}(\mathbf{q}_r) + \mathbf{C}(\mathbf{q}_r, \dot{\mathbf{q}}_r)\dot{\mathbf{q}}_d + \mathbf{F}_{fr}(\dot{\mathbf{q}}_r) \quad (26)$$

where $\tau = [\tau_1 \ \tau_2 \ \tau_3 \ \tau_4]^T$ is the vector of robot joint torques, $\mathbf{q}_r = [q_{1r} \ q_{2r} \ q_{3r} \ q_{4r}]^T$ are the robot joint angles, \mathbf{K}_v and \mathbf{K}_p are the gain matrices, and \mathbf{e} is the error vector between the desired and the robot joint angles, i.e.,

$$\mathbf{e} = [q_{1d} - q_{1r} \ q_{2d} - q_{2r} \ q_{3d} - q_{3r} \ q_{4d} - q_{4r}]^T. \quad (27)$$

\mathbf{I} , \mathbf{G} , \mathbf{C} , and \mathbf{F}_{fr} are the inertia tensor, the gravity vector, the Coriolis-centrifugal matrix, and the joint friction vector of the four actuated robot links and joints, respectively, identified in [41]. Vector $\ddot{\mathbf{q}}_d$ corresponds to the desired angular acceleration vector that was computed through simple differentiation of the desired joint angle vector $\mathbf{q}_d = [q_{1d} \ q_{2d} \ q_{3d} \ q_{4d}]^T$ using a necessary low-pass filter to cut off high frequencies.

By using the above controller, the robot arm was teleoperated by the user in joint space, i.e., robot joint angles mimic human-decoded joint angles. However, since the robot's and user's links have different length, the direct control in joint space would lead the robot end-effector to a different position in space than that desired by the user. Consequently, the user's hand position was computed by using the estimated joint angles and then commands the robot to position its end-effector at this point in the 3-D space. This was realized by using the forward kinematics of the human arm to compute the user's hand position and then solving the inverse kinematics for the robot arm to drive its end-effector to the same pose in the 3-D space. If \mathbf{p}_d , the pose vector, was computed from the human arm forward kinematics, then the desired joint angle vector \mathbf{q}_d was computed via the robot inverse kinematics [42], as depicted in Fig. 3.

F. Discussion and Rationale on the Decoding Architecture

The human musculoskeletal system and, consequently, the motor control system are highly nonlinear [4]. Moreover, since an EMG signal corresponds to muscle force, the relation between EMG recordings and arm motion is also nonlinear.

⁵The vector of joint angles decoded from EMG signals.

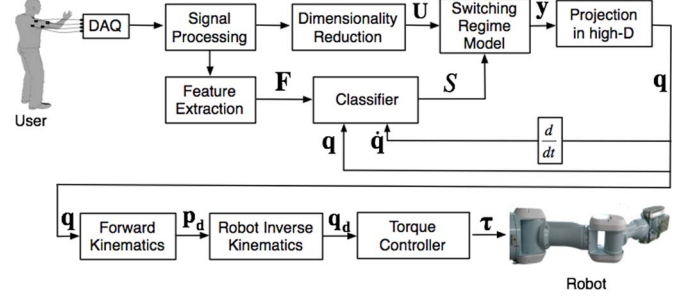


Fig. 3. Block diagram of the proposed methodology. \mathbf{q} is the vector of the four joint angles decoded from the EMG signals, while \mathbf{p}_d is the pose vector computed through the human arm kinematics given the four joint angles.

However, an analytic model of the musculoskeletal system, including the analyzed four DoFs of the arm, would be very complex, with a large number of unknown parameters to be identified. This would result to a model that would be very sensitive to the training paradigms, while it would entail a highly computational effort, making it impractical for real-time applications. By choosing linear techniques (i.e., the PCA method) and linear models with hidden states as in (10), we modeled the relationship between EMG and arm motion from a stochastic point of view. This enabled us to use well-known and computational effective techniques, resulting in a practical, efficient, and easily used method for controlling robotic devices using EMG signals. Furthermore, if EMG-controlled systems are to be widely used for rehabilitation or medical reasons, there is a need for a simple, easily trainable, and robust system.

As it was observed, there were significant EMG signal features that were changing during the experiment. These changes prohibited the use of a stationary model for decoding since this could result to erroneous arm motion estimates. A switching decoding model was developed that could compensate for the changes in the EMG signal level and, finally, accurately estimate the arm motion. The SR architecture introduced above allowed the switching between different decoding models that could map the relationship between EMG signals and motion, incorporating EMG signal features that changed along the experiment. In Section II-G, the proposed model performance was assessed through an adequate number of experiments, with different subjects. Moreover, a comparison with other models from the literature that can decode a continuous representation of motion using EMG signals will be presented.

G. Experimental and Verification Procedures

The proposed architecture was assessed through remote teleoperation of the robot arm using only EMG signals from the 11 muscles as analyzed above. The robot arm used was a seven-DoF anthropomorphic manipulator (PA-10, Mitsubishi Heavy Industries). Two PCs were used, running a Linux operating system. One of the PCs communicated with the robot controller through the ARCNET protocol in the frequency of 500 Hz, while the other acquired the EMG signals and the position tracker measurements (during the training phase). The two PCs were connected through serial communication (RS-232) interface for synchronization purposes. EMG signals were acquired

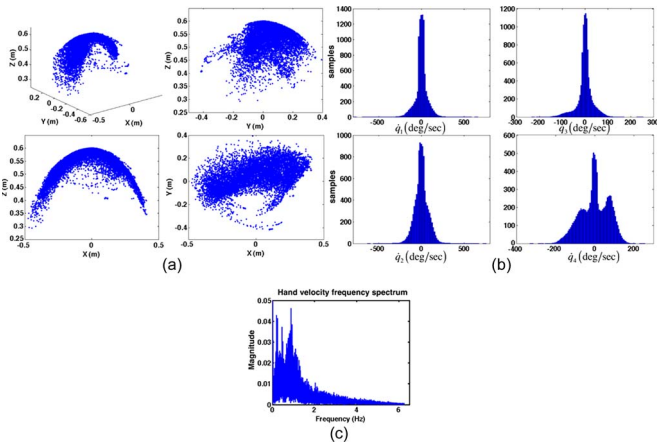


Fig. 4. User's upper limb motion during training. (a) Three-dimensional position of the user's hand in the Cartesian system defined in Fig. 1, and the three 2-D views of the same figure. (b) Distributions of the rotational velocities per joint during the training phase. (c) Frequency spectrum of the user's hand velocity during training.

using a signal acquisition board (NI-DAQ 6036E, National Instruments) connected to an EMG system (Bagnoli-16, Delsys Inc.). Single differential surface EMG electrodes (DE-2.1, Delsys Inc.) were used. The position tracking system (Isotrak II, Polhemus Inc.) used during the training phase was connected with the PC through serial communication interface (RS-232). The size of the position sensors is 2.83 (width) \times 2.29 (length) \times 1.51 (height) cm.

During the training phase, the user was instructed to move his arm to random positions in the 3-D space, with variable speed, to cover a wide range of the arm workspace. The 3-D position of the user's hand during this session, along with the distribution of the arm velocity, is shown in Fig. 4. As it can be seen, the user was moving his hand in a large portion of the available workspace, with speed varying inside the range of arm speed profiles for everyday-life tasks (i.e., 150°/s–200°/s for most joints) [43]. The frequency spectrum of the hand velocity is also shown to prove that the user was performing arm motions that spanned most of the usual frequency range of arm motions in everyday-life tasks (i.e., 1–1.5 Hz) [43]. The training phase has no resting periods. The duration of the training phase was decided at 4 min since, after that period, the user wanted to rest his arm. Moreover, during the real-time operation, a possible user would not be able to teleoperate the robot arm in the 3-D space for more than 4 min without a resting period. Regarding the computational cost of the model training, it took approximately 3 min in a PC with a 2.1-MHz Core Duo processor and 2-GB RAM. The training code was implemented using MATLAB (Mathworks Inc.).

As soon as the training period ended and the proposed switching model was trained, the real-time operation phase commenced. During this phase, the user had visual contact with the robot arm while teleoperating it in the 3-D space. Only EMG signals were used for estimating arm motion and, finally, controlling in real time the robot arm. The position tracking system was kept into place for a few experiments, only for offline validation purposes. Five experimental sessions were conducted using each subject, with each session lasting for

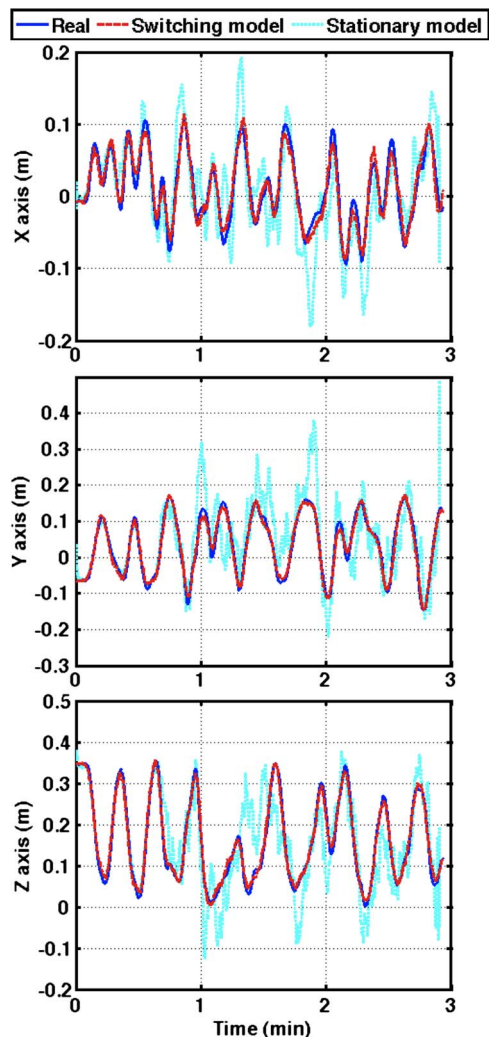


Fig. 5. Real and estimated hand trajectory along the x -, y -, and z -axes for a 3-min period. Estimates from the proposed switching method are quite close to the ground truth during the whole 3-min test, while the stationary model accuracy decreases after a period of approximately 30 s.

3.5 min. Real and estimated motion data were recorded during these sessions for evaluating the model performance. The real joint angle profiles were computed from the position tracker sensors, which were kept in place (i.e., on the user's arm) for offline validation reasons. Using the kinematic equations, we computed the estimated hand trajectory in the 3-D space using the estimated joint angles. This phase could have as many resting periods as desired by the user; however, it was noticed that, usually, the user was resting his arm after approximately 3 min of operation. The system was tested by four able-bodied persons, who found it convenient and accurate, while they were easily acquainted with its operation. All experimental procedures were conducted under a protocol approved by the National Technical University of Athens Institutional Review Board.

The proposed methodology was compared with other models used in the literature for decoding a continuous representation of motion using EMG signals. The linear filter method, widely used in the literature for decoding motion using neural signals [44], was used. The support vector machine (SVM) method

TABLE I
EFFECTIVENESS COMPARISON AMONG THE SWITCHING ARCHITECTURE, A STATIONARY MODEL, A LINEAR FILTER, AND AN SVM IN DECODING MOTION IN THE CARTESIAN SPACE. MEAN AND STANDARD DEVIATION VALUES ARE REPORTED FOR FIVE EXPERIMENTAL SESSIONS, LASTING 3.5 MIN EACH, ACROSS ALL THE FOUR SUBJECTS PARTICIPATED

Decoding model	CC_x	CC_y	CC_z	$RMSE_x$ (cm)	$RMSE_y$ (cm)	$RMSE_z$ (cm)
SR model	0.94 ± 0.02	0.95 ± 0.02	0.92 ± 0.02	2.34 ± 0.5	1.98 ± 0.3	2.12 ± 0.3
Stationary model	0.84 ± 0.02	0.80 ± 0.04	0.79 ± 0.05	4.12 ± 0.9	4.30 ± 0.8	5.94 ± 1.1
Linear Filter	0.82 ± 0.04	0.83 ± 0.05	0.84 ± 0.04	6.39 ± 3.2	7.30 ± 4.4	5.94 ± 2.1
SVM	0.80 ± 0.07	0.82 ± 0.05	0.81 ± 0.06	3.82 ± 0.6	3.30 ± 0.7	4.14 ± 0.6

was also used [7], [45], [46]. The experimental data used for model comparison were the same within the models and were collected through experimental sessions in which four male subjects (age group 24–30 years old) participated. Two criteria were used for assessing the accuracy of the reconstruction of human motion using the decoding models. These were the root-mean-square error (RMSE) and the correlation coefficient (CC). The latter describes essentially the similarity between the reconstructed and the true motion profiles and constitutes the most common means of reconstruction assessment for decoding purposes. Perfect matching between the estimated and the true angles corresponds to $CC = 1$. Standard deviations of the criterion values across all the experiments were also computed.

III. RESULTS

The estimated user's hand 3-D trajectory along with the ground truth, for an experimental session with one of the subjects, is depicted in Fig. 5. As it can be seen, the method could estimate the hand trajectory with high accuracy, compensating for EMG changes with respect to time. The latter is shown in Fig. 5, where the estimates based on a stationary decoding model of the same form of (10), which did not compensate for EMG time variation, are shown.⁶ As it can be seen, using a stationary model, the accuracy of the estimates decreases with time due to time variation of EMG signals. The motion presented in Fig. 5 was continuous without any resting periods. This was chosen to show that the method proposed was able to estimate the upper limb motion, even after EMG changes caused mainly by, but not limited to, muscle fatigue. Most users were not able to move their arms for longer periods without resting. However, the method was able to provide accurate motion estimates for long periods (including resting intervals).

Table I includes the values of the CC and the RMSE for the 3-D hand trajectory estimates coming from the switching model, the stationary model, and the other two methods used, i.e., the linear filter and the SVM algorithm. As it can be seen, the SR model was highly accurate in predicting arm motion, and it outperformed the other methods. The accuracy of the four compared methods in estimating arm motion with respect to time is shown in Fig. 6. The CC and RMSE criterion

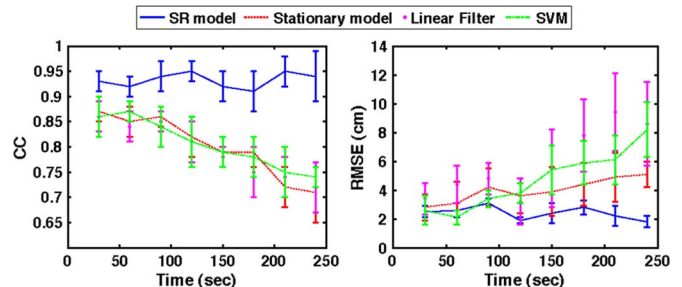


Fig. 6. CC and RMSE values (mean and standard deviations) for the estimated user's hand position in the Cartesian space with respect to time using the four models. Real and estimated distance of the user's hand with respect to the origin of the Cartesian reference system was used for computing the criterion values.

values for the Cartesian position of the user's hand in the 3-D space were calculated at each time interval of 30 s.⁷ Data within the corresponding 30-s interval were only used. Mean and standard deviation values are calculated for the five experimental sessions across the four subjects participated. As it can be seen, the proposed SR model was able to robustly estimate the user's motion, while the other three models showed deteriorating accuracy with respect to time.

From the above analysis of the results, it turns out that the method could be used for the real-time control of robots using EMG signals, for long periods, compensating for the EMG changes with respect to time. Finally, it must be noted that during real-time operation, the evaluation time required by the system to process the data was negligible since it was only a small portion of 1 ms that each loop of acquiring and processing EMG signals lasts.

The robot control was achieved through the torque controller defined in (26) using as desired robot trajectories the ones computed using the motion estimates from the SR model. The proposed robot controller was analyzed in [41], where it was verified that the robot could follow the desired trajectory with 1-ms delay (due to communication bandwidth) achieving an error in joint positioning less than 0.05° . Therefore, presenting the robot arm trajectories would be redundant since they would coincide with the estimates from the SR model.

IV. CONCLUSION AND DISCUSSION

In this paper, a methodology for controlling an anthropomorphic robot arm using EMG signals from the muscles of

⁶The motion estimates shown using a stationary model were computed using only the first model of the switching architecture, i.e., not allowing the model change with respect to time. Using a single model fitted to the whole training period and using this model instead could be another solution. However, the latter was proved to be less accurate in estimating motion than the first model of the switching architecture depicted.

⁷The distance of the user's hand with respect to the origin of the Cartesian reference system was used, instead of each coordinate separately, for simplicity.

the upper limb has been proposed. EMG signals recorded from muscles of the upper limb have been used for extracting kinematic variables (i.e., joint angles) to control an anthropomorphic robot arm in real time. Activations from 11 muscles have been used to estimate the motion of four DoFs of the user's arm during motion tasks in the 3-D space. EMG signal characteristic changes are usually noticed after a period of approximately 30 s of operation [23]. For this reason, a probabilistic framework has been designed to assign to each of the muscles recorded a *class* related to the recorded time-varying signal features. Then, an SR model has been built in such a way to compensate for the EMG changes. The proposed method has been tested in several real-time teleoperation tasks of a robot arm in the 3-D space by four different subjects. It has been shown from the experimental results that the proposed method could estimate the human arm motion using only EMG signals with high accuracy. The method proved to be robust to EMG signal characteristic variation, while it outperformed other widely used algorithms.

The novelty of the method proposed here can be centered around two main issues. First, the proposed method is not affected by EMG changes with respect to time. Since EMG is widely known as a nonstationary signal, the fact that the proposed method can compensate for EMG changes through time (caused by muscle fatigue or changes in the level of muscle force production) is quite important for the field. The second important issue presented here is that, to the best of our knowledge, this is the first time a continuous profile of 3-D arm motion (including four DoFs) is extracted using only EMG signals. Most previous works extract only discrete information about motion, while there are some works that estimate continuous arm motion, constrained, however, to isometric movements, single DoF, or very smooth motions [14], [21], [22]. In this paper, the method was tested for random motions in the 3-D space, with variable speed profiles.

Moreover, this study proposes a methodology that can be easily trained to each user and takes a little time to build the decoding model, while the computational load during real-time operation is negligible. In other words, the purpose of this study was to introduce a method for mapping EMG activity to motion that can be easily trained and used in a variety of applications. However, the authors do not propose a method that can be trained in a specific situation and generalizable to other ones that significantly differ. For example, significant changes in human arm dynamics, e.g., by having the user holding a heavy object, would require a new training session of the model to be able to decode motion in an efficient way. However, the method proposed is quite systematic and easy to apply in such situations, with low-computational cost.

It must be noted that the analysis of muscle fatigue or other muscle functions that cause changes into the electromyogram, from a physiological point of view, is out of the scope of this paper. However, the effects that these functions have on the EMG signals have been analyzed and mathematically incorporated into the proposed architecture to result to a robust system for EMG-based motion decoding.

With the use of EMG signals and robotic devices in the area of rehabilitation receiving increased attention during the last

years, our method could be proved beneficial. Moreover, the proposed method can be used in a variety of applications, where an efficient human-robot interface is required. For example, prosthetic or orthotic robotic devices mainly driven by user-generated signals can be benefitted by the proposed method, concluding to a user-friendly and effective control interface.

REFERENCES

- [1] Y. Woo-Keun, T. Goshozono, H. Kawabe, M. Kinami, Y. Tsumaki, M. Uchiyama, M. Oda, and T. Doi, "Model-based space robot teleoperation of ETS-VII manipulator," *IEEE Trans. Robot. Autom.*, vol. 20, no. 3, pp. 602–612, Jun. 2004.
- [2] T. Tayh-Jong, A. Bejczy, G. Chuanfan, and X. Ning, "Intelligent planning and control for telerobotic operations," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 1994, pp. 389–396.
- [3] J. Park and O. Khatib, "A haptic teleoperation approach based on contact force control," *Int. J. Robot. Res.*, vol. 25, no. 5/6, pp. 575–591, May/Jun. 2006.
- [4] F. E. Zajac, "Muscle and tendon: Properties, models, scaling, and application to biomechanics and motor control," in *CRC Critical Reviews in Biomedical Engineering*, J. R. Bourne, Ed. Boca Raton, FL: CRC Press, 1986, pp. 359–411.
- [5] O. Fukuda, T. Tsuji, M. Kaneko, and A. Otsuka, "A human-assisting manipulator teleoperated by EMG signals and arm motions," *IEEE Trans. Robot. Autom.*, vol. 19, no. 2, pp. 210–222, Apr. 2003.
- [6] J. Zhao, Z. Xie, L. Jiang, H. Cai, H. Liu, and G. Hirzinger, "Levenberg-marquardt based neural network control for a five-fingered prosthetic hand," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 4482–4487.
- [7] S. Bitzer and P. van der Smagt, "Learning EMG control of a robotic hand: Towards active prostheses," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 2819–2823.
- [8] M. Zecca, S. Micera, M. C. Carrozza, and P. Dario, "Control of multifunctional prosthetic hands by processing the electromyographic signal," *Crit. Rev. Biomed. Eng.*, vol. 30, no. 4–6, pp. 459–485, 2002.
- [9] D. Nishikawa, W. Yu, H. Yokoi, and Y. Kakazu, "EMG prosthetic hand controller using real-time learning method," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 1999, pp. 153–158.
- [10] K. Takahashi, T. Nakauke, and M. Hashimoto, "Remarks on hands-free manipulation using bio-potential signals," in *Proc. ISIC/IEEE Int. Conf. Syst., Man Cybern.*, 2007, pp. 2965–2970.
- [11] K. Kiguchi, S. Kariya, K. Watanabe, K. Izumi, and T. Fukuda, "An exoskeletal robot for human elbow motion support-sensor fusion, adaptation, and control," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 31, no. 3, pp. 353–361, Jun. 2001.
- [12] Y. Suzuki, T. Tanaka, S. Kaneko, S. Moromugi, and M. Feng, "Soft sensor suits as man-machine interface for wearable power amplifier," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, 2005, pp. 1680–1685.
- [13] K. Kita, R. Kato, H. Yokoi, and T. Arai, "Development of autonomous assistive devices—Analysis of change of human motion patterns," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, 2006, pp. 316–321.
- [14] R. J. Smith, F. Tenore, D. Huberdeau, R. Etienne-Cummings, and N. V. Thakor, "Continuous decoding of finger position from surface EMG signals for the control of powered prostheses," in *Proc. Annu. Int. IEEE EMBS Conf.*, 2008, pp. 197–200.
- [15] A. V. Hill, "The heat of shortening and the dynamic constants of muscle," *Proc. R. Soc. Lond. B, Biol.*, vol. 126, no. 843, pp. 136–195, Oct. 1938.
- [16] E. Cavallaro, J. Rosen, J. C. Perry, S. Burns, and B. Hannaford, "Hill-based model as a myoprocessor for a neural controlled powered exoskeleton arm—Parameters optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 4514–4519.
- [17] P. K. Artemiadis and K. J. Kyriakopoulos, "Teleoperation of a robot manipulator using EMG signals and a position tracker," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2005, pp. 1003–1008.
- [18] J. Potvin, R. Norman, and S. McGill, "Mechanically corrected EMG for the continuous estimation of erector spine muscle loading during repetitive lifting," *Eur. J. Appl. Physiol.*, vol. 74, no. 1/2, pp. 119–132, Aug. 1996.
- [19] D. G. Lloyd and T. F. Besier, "An EMG-driven musculoskeletal model to estimate muscle forces and knee joint moments in vivo," *J. Biomechanics*, vol. 36, no. 6, pp. 765–776, Jun. 2003.
- [20] K. Manal, T. S. Buchanan, X. Shen, D. G. Lloyd, and R. V. Gonzalez, "Design of a real-time EMG driven virtual arm," *Comput. Biol. Med.*, vol. 32, no. 1, pp. 25–36, Jan. 2002.

- [21] Y. Koike and M. Kawato, "Estimation of dynamic joint torques and trajectory formation from surface electromyography signals using a neural network model," *Biol. Cybern.*, vol. 73, no. 4, pp. 291–300, Sep. 1995.
- [22] W. Ryu, B. Han, and J. Kim, "Continuous position control of 1 DOF manipulator using EMG signals," in *Proc. Int. Conf. Convergence Hybrid Inf. Technol.*, 2008, pp. 870–874.
- [23] N. A. Dimitrova and G. V. Dimitrov, "Interpretation of EMG changes with fatigue: Facts, pitfalls, and fallacies," *J. Electromyography Kinesiology*, vol. 13, no. 1, pp. 13–36, Feb. 2003.
- [24] T. Moritani, M. Muro, and A. Nagata, "Intramuscular and surface electromyogram changes during muscle fatigue," *J. Appl. Physiol.*, vol. 60, no. 4, pp. 1179–1185, Apr. 1986.
- [25] J. Fernández, R. Acevedo, and C. Tabernig, "Influence of muscular fatigue in evoked electromyogram," *J. Phys.: Conf. Ser.*, vol. 90, p. 012 079, 2007.
- [26] R. Merletti, M. Knafitz, and C. D. Luca, "Myoelectric manifestations of fatigue in voluntary and electrically elicited contractions," *J. Appl. Physiol.*, vol. 69, no. 5, pp. 1810–1820, Nov. 1990.
- [27] M. Schwartz, *Biofeedback: A Practitioner's Guide*. New York: Guilford Press, 1987.
- [28] J. R. Cram and G. S. Kasman, *Introduction to Surface Electromyography*. Gaithersburg, MD: Aspen, 1998.
- [29] J. Jeong, W. Cho, Y. Kim, and H. Choi, "Recognition of lower limb muscle EMG patterns by using neural networks during the postural balance control," in *Proc. Int. Conf. Biomed. Eng.*, 2007, pp. 82–85.
- [30] M. M. Morlock, V. Bonin, G. Muller, and E. Schneider, "Trunk muscle fatigue and associated EMG changes during a dynamic iso-inertial test," *Eur. J. Appl. Physiol.*, vol. 76, no. 1, pp. 75–80, Jun. 1997.
- [31] H. Oka, "Estimation of muscle fatigue by using EMG and muscle stiffness," in *Proc. IEEE Int. Conf. Eng. Med. Biol. Soc.*, 1996, vol. 4, pp. 1449–1450.
- [32] P. K. Artemiadis and K. J. Kyriakopoulos, "EMG-based teleoperation of a robot arm using low-dimensional representation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2007, pp. 489–495.
- [33] B. Lim, S. Ra, and F. Park, "Movement primitives, principal component analysis, and the efficient generation of natural motions," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 4630–4635.
- [34] F. A. Mussa-Ivaldi and E. Bizzi, "Motor learning: The combination of primitives," *Philos. Trans. Roy. Soc. Lond. B, Biol. Sci.*, vol. 355, no. 1404, pp. 1755–1769, Dec. 2000.
- [35] A. Daffertshofer, C. J. C. Lamoth, O. G. Meijer, and P. J. Beek, "PCA in studying coordination and variability: A tutorial," *Clin. Biomech.*, vol. 19, no. 4, pp. 415–428, May 2004.
- [36] C. Kim, J. Piger, and R. Startz, "Estimation of Markov regime-switching regression models with endogenous switching," *J. Econometrics*, vol. 143, no. 2, pp. 263–273, Apr. 2008.
- [37] F. Felici, J.-W. van Wingerden, and M. Verhaegen, "Subspace identification of MIMO LPV systems using a periodic scheduling sequence," *Automatica*, vol. 43, no. 10, pp. 1684–1697, Oct. 2007.
- [38] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Hoboken, NJ: Wiley, 2001.
- [39] G. McLachlan and D. Peel, *Finite Mixture Models*. Hoboken, NJ: Wiley, 2000.
- [40] K. P. Burnham and D. R. Anderson, *Model Selection and Multimodel Inference: A Practical-Theoretic Approach*. New York: Springer-Verlag, 2002.
- [41] N. A. Mpompos, P. K. Artemiadis, A. S. Oikonomopoulos, and K. J. Kyriakopoulos, "Modeling, full identification and control of the Mitsubishi PA-10 robot arm," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, Zurich, Switzerland, 2007, pp. 1–6.
- [42] L. Sciacivco and B. Siciliano, *Modeling and Control of Robot Manipulators*. New York: McGraw-Hill, 1996.
- [43] D. M. Mirkov, S. Milanovic, D. B. Ilic, and S. Jaric, "Symmetry of discrete and oscillatory elbow movements: Does it depend on torque that the agonist and antagonist muscle can exert?" *Motor Control*, vol. 6, no. 3, pp. 271–281, Jul. 2002.
- [44] J. M. Carmena, M. A. Lebedev, R. E. Crist, J. E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. C. S. Henriquez, and M. A. L. Nicolelis, "Learning to control a brain-machine interface for reaching and grasping by primates," *PLoS Biol.*, vol. 1, no. 2, pp. 193–208, Nov. 2003.
- [45] C. Castellini, F. Orabona, G. Metta, and G. Sandini, "Internal models of reaching and grasping," *Adv. Robot.*, vol. 21, no. 12, pp. 1545–1564, 2007.
- [46] L. Chen, P. Yang, X. Xu, X. Guo, and X. Zhang, "Fuzzy support vector machine for EMG pattern recognition and myoelectrical prosthesis control," in *Proc. 4th Int. Symp. Neural Netw.: Part II—Adv. Neural Netw.*, 2007, pp. 1291–1298.



Panagiotis K. Artemiadis received the Diploma and Ph.D. degrees in mechanical engineering from the National Technical University of Athens, Athens, Greece, in 2003 and 2009, respectively.

Since 2009, he has been a Postdoctoral Associate with the Newman Laboratory for Biomechanics and Human Rehabilitation, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge. He has authored or coauthored more than 17 papers published in journals and presented at refereed conferences, and three of them have been included in the top papers in international conferences. He has been a Reviewer and an Associate Editor of a number of journals and conferences, and has worked on projects funded by the European Commission. His current research interests include the areas of rehabilitation robotics, neurorobotics, control of robot manipulators, and mechatronics.

Dr. Artemiadis is a member of the Hellenic Bioscientific Association and the Technical Chamber of Greece.



Kostas J. Kyriakopoulos received the Diploma in mechanical engineering (Honors) from the National Technical University of Athens (NTUA), Athens, Greece, in 1985 and the M.S. and Ph.D. degrees in electrical, computer, and systems engineering (ECSE) from Rensselaer Polytechnic Institute (RPI), Troy, NY, in 1987 and 1991, respectively.

From 1988 to 1991, he did research with the National Aeronautics and Space Administration Center for Intelligent Robotic Systems for Space Exploration. Between 1991 and 1993, he was an Assistant

Professor with ECSE, RPI, and the New York State Center for Advanced Technology in Automation and Robotics. Since 1994, he has been with the Control Systems Laboratory of the Mechanical Engineering Department, NTUA, where he is currently a Professor and the Director of the Departmental Computation Laboratory. He has published close to 180 papers to journals and refereed conferences. His current interests include the areas of nonlinear control system applications in sensor-based motion planning and control of multirobotic systems: manipulators and vehicles (mobile, underwater, and aerial) and micromechatronics and biomechatronics.

Dr. Kyriakopoulos is a member of the European Robotics Research Network and the Technical Chamber of Greece. He was a recipient of the G. Samaras Award of Academic Excellence from NTUA, the Bodosakis Foundation Fellowship (1986–1989), the Alexander Onassis Foundation Fellowship (1989–1990), and the Alexander Von Humboldt Foundation Fellowship (1993). He serves in the editorial committee of a number of journals and has served as an administrative member of a number of international conferences.