

few iterations are sufficient to significantly decrease the normal-force tracking error. Further, the proposed algorithm can cope with situations in which it is necessary to contour objects that are not perfectly identical to each other or that are not positioned exactly in the same location.

REFERENCES

- [1] G. Ferretti, G. Magnani, and P. Rocco, "Triangular force/position control with application to robotic deburring," *Mach. Intell. Robot. Control*, vol. 2, pp. 83–91, 2000.
- [2] G. Ziliani, G. Legnani, and A. Visioli, "A mechatronic approach for robotic deburring," *Mechatronics*, vol. 17, pp. 431–441, 2007.
- [3] S. Ahmad and C. N. Lee, "Shape recovery from robot contour-tracking with force feedback," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1990, pp. 447–452.
- [4] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators," *ASME J. Dyn. Syst., Meas., Control*, vol. 102, pp. 126–133, 1981.
- [5] J. Roy and L. L. Whitcomb, "Adaptive force control of position/velocity controlled robots: Theory and experiments," *IEEE Trans. Robot. Autom.*, vol. 18, no. 2, pp. 121–137, Apr. 2002.
- [6] K. L. Moore, "Iterative learning control An expository overview," *Appl. Comput. Controls, Signal Process., Circuits*, vol. 1, pp. 151–214, 1999.
- [7] T. Naniwa and S. Arimoto, "Learning control for robot tasks under geometric endpoint constraints," *IEEE Trans. Robot. Autom.*, vol. 11, no. 3, pp. 432–441, Jun. 1995.
- [8] F. Lange and G. Hirzinger, "Iterative self-improvement of force feedback control in contour tracking," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1992, pp. 1399–1404.
- [9] F. Jatta, G. Legnani, A. Visioli, and G. Ziliani, "On the use of velocity feedback in hybrid force/velocity control of industrial manipulators," *Control Eng. Pract.*, vol. 14, no. 9, pp. 1045–1055, 2006.
- [10] F. Jatta, G. Legnani, and A. Visioli, "Friction compensation in hybrid force/velocity control of industrial manipulators," *IEEE Trans. Ind. Electron.*, vol. 53, no. 2, pp. 604–613, Apr. 2006.
- [11] G. Ferretti, G. Magnani, and P. Rocco, "Toward the implementation of hybrid position/force control in industrial robots," *IEEE Trans. Robot. Autom.*, vol. 13, no. 6, pp. 838–845, Dec. 1997.
- [12] F. Aghili and M. Namvar, "A self-tuning torque feedback for control of manipulators," presented at the IFAC Symp. Robot Control, Bologna, Italy, 2006.
- [13] R. W. Longman, "Iterative learning control and repetitive control for engineering practice," *Int. J. Control*, vol. 73, no. 10, pp. 930–954, 2000.
- [14] M. Norrlof, "On analysis and implementation of iterative learning control," Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 1998.
- [15] A. Visioli and G. Legnani, "On the trajectory tracking control of industrial SCARA robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 49, no. 1, pp. 224–232, Feb. 2002.
- [16] P. R. Belanger, P. Dobrovolny, A. Helmy, and X. Zhang, "Estimation of angular velocity and acceleration from shaft-encoder measurements," *Int. J. Robot. Res.*, vol. 17, no. 11, pp. 1225–1233, 1998.
- [17] S. J. Ovaska and S. Valiiviita, "Angular acceleration measurement: A review," *IEEE Trans. Instrum. Meas.*, vol. 47, no. 5, pp. 1211–1217, Oct. 1998.

EMG-Based Control of a Robot Arm Using Low-Dimensional Embeddings

Panagiotis K. Artemiadis and Kostas J. Kyriakopoulos

Abstract—As robots come closer to humans, an efficient human–robot-control interface is an utmost necessity. In this paper, electromyographic (EMG) signals from muscles of the human upper limb are used as the control interface between the user and a robot arm. A mathematical model is trained to decode upper limb motion from EMG recordings, using a dimensionality-reduction technique that represents muscle synergies and motion primitives. It is shown that a 2-D embedding of muscle activations can be decoded to a continuous profile of arm motion representation in the 3-D Cartesian space, embedded in a 2-D space. The system is used for the continuous control of a robot arm, using only EMG signals from the upper limb. The accuracy of the method is assessed through real-time experiments, including random arm motions.

Index Terms—Dimensionality reduction, electromyographic (EMG)-based control, EMG signals, neurorobotics.

I. INTRODUCTION

Despite the fact that robots came about approximately 50 years ago, the way humans control them is still an important issue. In particular, since robots are being used more frequently in everyday life tasks (e.g., service robots, robots for clinical applications), the human–robot interface plays a role of utmost significance. In this paper, a new mean of control interface is proposed, according to which, the user performs natural motions with his/her upper limb, while superficially recorded electromyographic (EMG) activity of the muscles of the upper limb is transformed to kinematic variables that are used to control a robot arm.

EMG signals have often been used as control interfaces for robotic devices. However, in most cases, only discrete control has been realized, focusing only, for example, at the directional control of robotic wrists [1]. This can cause many problems regarding smoothness of motion, while from a teleoperation point of view, a small number of finite commands or postures can critically limit the areas of application. Since most previous studies focused on EMG signal discrimination, a variety of algorithms have been proposed for this scope. A statistical log-linearized-Gaussian-mixture-neural network has been proposed in [1] to discriminate EMG patterns for wrist motions. A small number of researchers have tried to build continuous models to decode arm motion from EMG signals. The Hill-based muscle model [2], [3] is more frequently used in [4]. However, only a few degrees of freedom (DOFs) were analyzed (i.e., 1 or 2), since the nonlinearity of the model and the large numbers of unknown parameters for each muscle make the analysis rather difficult. A neural-network model was used to extract continuous arm motion in the past using EMG signals [5]; however, the movements analyzed were restricted to single-joint, isometric motions.

Manuscript received October 28, 2008; revised December 11, 2009. First published January 29, 2010; current version published April 7, 2010. This paper was recommended for publication by Associate Editor F. Thomas and K. Yamane and Editor W. K. Chung upon evaluation of the reviewers' comments. This work was supported by the European Commission under the contract NEUROBOTICS (FP6-IST-001917) project.

P. K. Artemiadis was with the Control Systems Lab, School of Mechanical Engineering, National Technical University of Athens, Athens, Greece. He is now 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, Athens 15780, 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/TRO.2009.2039378

Recent work in the field of biomechanics proposes that muscles are activated collectively, forming time-varying muscle synergies [6]. This finding suggests that muscle activations can be represented into a low-dimensional space, where these synergies can be represented instead of individual activations. Studies in human–motor control have also suggested that a low-dimensional representation is feasible at the arm kinematic level (i.e., joint angles) as well [7]. Identifying those underlying low-dimensional representations of muscle activations and movements performed, one could come up with a more robust way of decoding EMG signals to motion.

In this paper, a methodology for controlling an anthropomorphic robot arm using EMG signals from the muscles of the upper limb is proposed. Nine bipolar surface EMG electrodes record the muscular activity of equal number of muscles acting on the shoulder and the elbow joints. The system architecture is divided into two phases: the training phase and the real-time operation phase. During the training phase, the user is instructed to move his/her arm toward random positions in the three-dimensional (3-D) space. A position-tracking system is used to record the arm motion (i.e., joint angles) during reaching. The recorded muscle activations are represented into a low-dimensional space through a dimensionality reduction technique. Joint angle profiles are also embedded into a low-dimensional manifold. The mapping between those two low-dimensional spaces is realized through a linear model whose parameters are identified using the previously collected data. As soon as the linear model is trained, the real-time operation phase commences. During this phase, the trained model outputs the decoded motion using only the EMG recordings. A control law that utilizes these motion estimates is applied to the robot arm actuators. In this phase, the user can teleoperate the robot arm in real-time, while he/she can correct any possible deviations since he/she has visual contact with the robot. The efficacy of the proposed method is assessed through a large number of experiments, during which, the user controls the robot arm in performing random movements in the 3-D space.

II. METHODOLOGY

A. Background and Problem Definition

There is no doubt that the musculoskeletal system of humans is quite efficient, yet very complex. Narrowing our interest down to the upper limb and not considering finger motion, approximately 30 muscles actuate 7 DOFs. In this paper, we are focusing on the principal joints of the upper limb, i.e., the shoulder and the elbow. The wrist motion is not included in the analysis for simplicity. Hence, 4 DOFs will be analyzed from a kinematic point of view, as shown in Fig. 1.

In order to record the motion of the upper limb and then to extract the joint angles of the four modeled DOFs, a magnetic position-tracking system was used. The system is equipped with two position trackers and a reference system, with respect to which the 3-D position of the trackers is provided. In order to compute the four joint angles, one position tracker is placed at the user's elbow joint and the other one at the wrist joint. The reference system is placed on the user's shoulder. The setup, as well as the four modeled DOFs, are shown in Fig. 1. Let $\mathbf{T}_1 = [x_1 \ y_1 \ z_1]^T$, $\mathbf{T}_2 = [x_2 \ y_2 \ z_2]^T$ denote 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

$$q_1 = \arctan 2(\pm y_1, x_1)$$

$$q_2 = \arctan 2(\pm \sqrt{x_1^2 + y_1^2}, z_1)$$

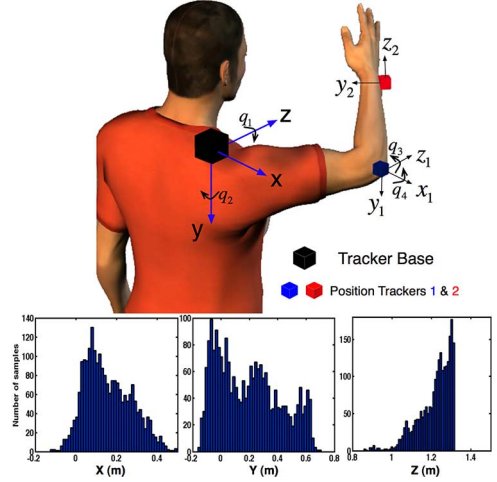


Fig. 1. (Top) User performs random motion in the 3-D space. q_1 and q_2 axes of rotation are perpendicularly intersecting and, therefore, can model the positioning of the elbow at the 3-D space due to the shoulder motion. q_3 corresponds to shoulder internal–external rotation, while q_4 corresponds to elbow flexion–extension. (Bottom) The variability of the 3-D position of the hand is shown through the histograms of each of the x , y , and z coordinate of the hand with respect to the base frame located on the shoulder.

$$q_3 = \arctan 2(\pm B_3, B_1)$$

$$q_4 = \arctan 2(\pm \sqrt{B_1^2 + B_3^2}, B_2 - L_1) \quad (1)$$

and

$$B_1 = x_2 \cos(q_1) \cos(q_2) + y_2 \sin(q_1) \cos(q_2) - z_2 \sin(q_2)$$

$$B_2 = -x_2 \cos(q_1) \sin(q_2) - y_2 \sin(q_1) \sin(q_2) - z_2 \cos(q_2)$$

$$B_3 = -x_2 \sin(q_1) + y_2 \cos(q_1) \quad (2)$$

where L_1 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., $L_1 = \|\mathbf{T}_1\| = \sqrt{x_1^2 + y_1^2 + z_1^2}$. Likewise, the length of the forearm L_2 can be computed from the distance between the two position trackers, i.e., $L_2 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$. One out of the multiple solutions given by (1) is selected for each joint angle, based on the range of motion for each human joint; if that is not enough to solve the ambiguity, the solution selected is the one that is closer to the previous value computed.

The position-tracking system provides the position vectors $\mathbf{T}_1, \mathbf{T}_2$ at the frequency of 60 Hz. Using an antialiasing finite-impulse response (FIR) filter, these measurements are resampled at the frequency of 1 kHz to be consistent with the muscle activations sampling frequency. An upsampling for the position tracker is selected instead of downsampling EMG signals in order to achieve the highest frequency possible for the proposed robot-control interface.

Based on the biomechanics [8], a group of nine muscles that are mainly responsible for the studied motion is recorded: deltoid (anterior), deltoid (posterior), deltoid (middle), pectoralis major, pectoralis major (clavicular head), trapezius, biceps brachii, brachioradialis, and triceps brachii. A smaller number of muscles could have been recorded (e.g., focusing on one pair of agonist–antagonist muscles for each joint). However, in order to investigate a wider arm motion variability, where less-significant muscles could play an important role in specific arm configurations, a group of nine muscles were selected. Surface

bipolar EMG electrodes used for recording are placed on the user's skin, following the directions given in [8]. Raw EMG signals after amplification are digitized at the sampling frequency of 1 kHz. Then, a full wave rectification takes place, and the signals are then low-pass filtered using a fourth-order Butterworth filter, with a cutoff frequency of 4 Hz. Finally, the signals from each muscle are normalized to their maximum voluntary isometric contraction value [3]. Three able-bodied subjects were used (three males of 27 ± 3 years old), while during the experiment, the subjects were standing close to the robot arm, with their neck positioned looking at front. All experimental procedures were conducted under a protocol approved by the National Technical University of Athens Institutional Review Board.

As mentioned earlier, the system initiates with a training phase, where EMG recordings and motion data are collected for model training. Concerning EMG recordings, after data collection, we have the muscle activations $u_k^{(i)}$ of each muscle i at time kT , where T the sampling period, and $k = 1, 2, \dots$. Regarding motion data, using the position-tracker readings and (1), the corresponding joint angles q_{1k} , q_{2k} , q_{3k} , and q_{4k} are collected. The dimension of these sets is quite large (i.e., nine variables for muscle activations and four for joint angles) making the mapping between them excessively hard. In order to deal with this dimensionality issue, a dimension-reduction technique is applied.

B. Dimension Reduction

The problem of dimension reduction is introduced as an efficient way to overcome the curse of the dimensionality when dealing with vector data in high-dimensional spaces and as a modeling tool for such data. In our case, muscle activations and joint angles are the high-dimensional data that will be embedded into two manifolds of lower dimension. The most widely used dimension-reduction technique is principal component analysis (PCA) [9]. In this paper, the PCA algorithm will be implemented twice: once for finding the new representation of the muscle activation data and then once more for the representation of joint angles. For details about the method, see [9]. In order to collect data during the training period, the user is instructed to move the arm in the 3-D Cartesian space toward random locations, as shown in Fig. 1. The variability of these locations during the training phase is also depicted through histograms for each axis. Muscle activation and joint angles are available after the preprocessing described earlier.

The PCA algorithm results in a representation of the original data to a new coordinate system of lower dimension, using the *first* eigenvectors of the original data as axes. In our case, for muscle recordings, the first two principal components were capable of describing 95% of the total variance. Regarding joint angles, the first two principal components described 93% of the total variance as well. Therefore, the low-dimensional representation of the nine muscles activation during 3-D motion of the arm is defined by

$$\Xi = \mathbf{V}^T \mathbf{K} \quad (3)$$

where \mathbf{V} is a 9×2 matrix, whose columns are the first two eigenvectors resulting from the PCA method and \mathbf{K} is the $9 \times m$ matrix computed from the matrix of EMG measurements after subtracting the mean value of each muscle across the m measurements. Likewise, the low-dimensional representation of joint angles during 3-D motion of the arm is defined by

$$\Phi = \mathbf{W}^T \mathbf{\Lambda} \quad (4)$$

where \mathbf{W} is a 4×2 matrix, whose columns are the two first eigenvectors resulting from the PCA method, and $\mathbf{\Lambda}$ is the $4 \times m$ matrix computed from the matrix of joint angle measurements after subtracting the mean value of each joint angle across the m measurements.

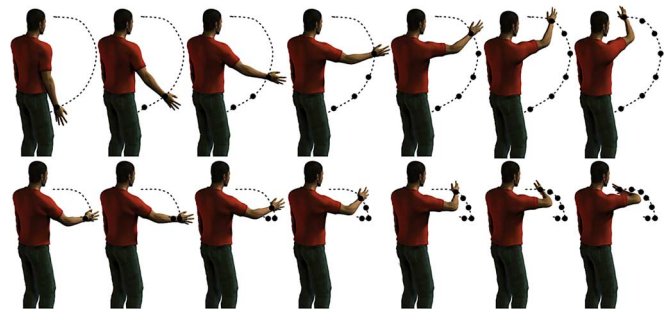


Fig. 2. Motion primitives computed from the first two principal components. (Top) The motion described along the first eigenvector. (Bottom) The motion described along the second eigenvector.

Using the aforementioned dimension-reduction technique, the high-dimensional data of muscle activations and corresponding joint angles were represented into two manifolds of fewer dimensions.

In particular for joint angles, using two instead of four variables to describe arm movement suggests motor primitives, which is a general conception that has been extensively analyzed in [6]. For analysis reasons, it is interesting to see what these two variables describe in the high-dimensional space. In other words, how the variation in the two axes of the low-dimensional manifold can be represented back into the high-dimensional space of the four joint angles and, consequently, the arm movement. This is shown in Fig. 2. In the top of Fig. 2, the arm motion depicted corresponds to the variation in the first axis of the low-dimensional manifold, i.e., along the first eigenvector extracted from the PCA on the arm kinematics. It is evident that the first-principal component describes the motion of the arm on a plane parallel to the coronal plane. Considering the second-principal component, it can be regarded as describing motion in the transverse plane. The authors do not claim that human-motor-control system uses these two motor primitives to perform any 3-D motion in general. As noted before, the proof of the presence of internal coordination mechanisms of the human-motor control is way beyond the scope of this study. On the contrary, this paper focuses on extracting task-specific motor primitives and, by using the proper mathematical formulation, employ them to control robots. It must be noted, however, that being able to represent the motion of the human arm in Cartesian space by using only two independent variables (i.e., the two low-dimensional representation of arm motion extracted through the PCA) restricts the variability of performed motion. Therefore, if the strict notion of DOFs is adopted, 2 DOFs are actually decoded using EMG. However, based on the suggestion of motor synergies, i.e., the human joint angles dependencies present in arm movements, these 2 DOFs decoded can be represented back in the high-dimensional space, where four human joint angles are actuated, and the human arm is finally moving in the 3-D Cartesian space, with limited workspace, however.

C. Decoding Arm Motion from EMG Signals

Having the low-dimensional embeddings, we can define the following linear-state-space model used in our previous works to map muscle activations to arm motion in real-time

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \mathbf{B}\boldsymbol{\xi}_k + \mathbf{w}_k \\ \mathbf{z}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{v}_k \end{aligned} \quad (5)$$

where $\mathbf{x}_k \in \mathbb{R}^d$ is a hidden state vector at time instance kT , $k = 1, 2, \dots, T$ is the sampling period, d is the dimension of this vector, $\boldsymbol{\xi} \in \mathbb{R}^2$ is the vector of the low-dimensional muscle activations,

and $\mathbf{z}_k \in \mathbb{R}^2$ is the vector of the low-dimensional joint kinematics. The matrix \mathbf{A} determines the dynamic behavior of the hidden state vector \mathbf{x} , \mathbf{B} is the matrix that relates muscle activations ξ to the state vector \mathbf{x} , while \mathbf{C} is the matrix that represents the relationship between the joint kinematics \mathbf{z}_k and the state vector \mathbf{x} . \mathbf{w}_k and \mathbf{v}_k represent zero-mean-Gaussian noise in the process and observation equations, respectively, i.e., $\mathbf{w}_k \sim N(\mathbf{0}, \Psi)$ and $\mathbf{v}_k \sim N(\mathbf{0}, \Gamma)$, where $\Psi \in \mathbb{R}^{d \times d}$ and $\Gamma \in \mathbb{R}^{2 \times 2}$ are the covariance matrices of \mathbf{w}_k and \mathbf{v}_k , respectively. The hidden variables can model the unobserved, intrinsic system states and thus facilitate the correlation between the observed muscle activations and arm kinematics. Model training entails the estimation of the matrices \mathbf{A} , \mathbf{B} , \mathbf{C} , Ψ , and Γ . This is implemented using an iterative prediction-error minimization (i.e., maximum likelihood) algorithm [10]. The dimension d of the state vector should also be selected for model fitting. This is done in parallel with the fitting procedure by deciding the greater number of states, where any additional states do not contribute more to the model input–output behavior.

One could argue that from the physiological point of view, the musculoskeletal system and, consequently, the human–motor-control system is highly nonlinear. However, as noted before, an analytic model of the musculoskeletal system, including the analyzed 4 DOFs of the arm, would be very complex, with a large numbers of unknown parameters to be identified. By choosing linear techniques (i.e., the PCA method) and linear models with hidden states as in (5), we try to model the relationship between EMG and arm motion from a stochastic point of view. The latter enable 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.

Having the model trained, the real-time operation phase commences. Raw EMG signals are collected, preprocessed, and then represented by the low-dimensional manifolds using (3). Then, the fitted model (5) is used. The model outputs the low-dimensional arm kinematics vector \mathbf{z}_k at each time instance kT . This vector is transformed back to the 4-D dimensional space, representing the estimates for the four joint angles of the upper limb. This is done by using (4) and solving it for the high-dimensional vector of joint angles, i.e., $\mathbf{q}_{H_k} = \mathbf{W}\mathbf{z}_k$, where \mathbf{q}_{H_k} is the 4×1 vector with the four estimates for the arm joint angles at time instance kT , i.e., the high-dimensional representation of arm kinematics. Having computed the estimated joints angles $\mathbf{q}_{H_k} = [\hat{q}_{1_k} \ \hat{q}_{2_k} \ \hat{q}_{3_k} \ \hat{q}_{4_k}]^T$, we can then command the robot arm. However, since the robot and the user's links have different lengths, the direct control in joint space would lead the robot end-effector in a different position in space than that desired by the user. Consequently, the user's hand position should be computed by using the estimated joint angles and then command the robot to position its end-effector at this point in space. This is 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 position in the 3-D space. Hence, the final command to the robot arm is in joint space. Therefore, the robot controller analysis assumes that a final vector $\mathbf{q}_d = [q_{1d} \ q_{2d} \ q_{3d} \ q_{4d}]^T$ containing the four desired robot joint angles is provided, where these joint angles are computed through the robot inverse kinematics, as described earlier. An inverse-dynamic-control law is utilized to drive the robot arm and is defined by

$$\begin{aligned} \tau = & \mathbf{I}(\mathbf{q}_r) (\ddot{\mathbf{q}}_d + \mathbf{K}_v \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e}) + \mathbf{G}(\mathbf{q}_r) \\ & + \mathbf{C}(\mathbf{q}_r, \dot{\mathbf{q}}_r) \dot{\mathbf{q}}_r + \mathbf{F}_{fr}(\dot{\mathbf{q}}_r) \end{aligned} \quad (6)$$

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

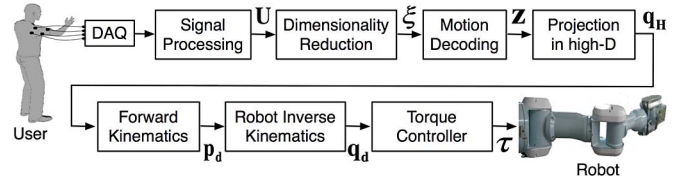


Fig. 3. Proposed system architecture. \mathbf{p}_d denotes the human-hand-position vector computed from the estimated joint angles through the human-forward kinematics.

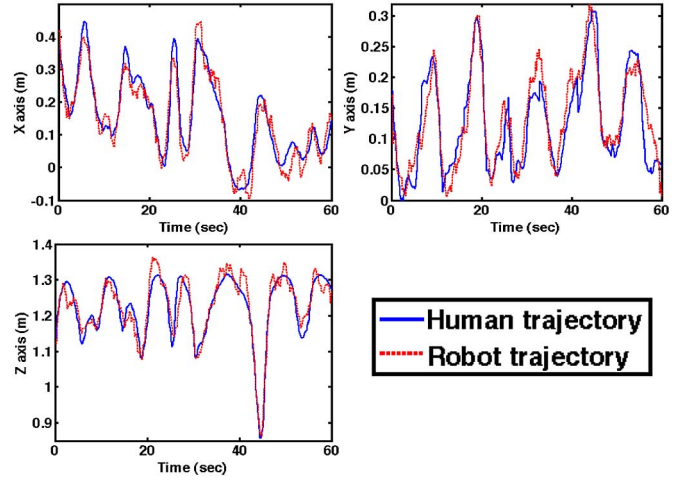


Fig. 4. Human hand and robot end-effector trajectory along the x , y , and z axes, for a 1-min period.

the robot joint angles, while \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, which are identified in [11]. A block diagram depicting the total architecture proposed for decoding EMG signals into motion and the control of the robot arm is depicted in Fig. 3.

III. RESULTS

A. Hardware and Experiment Design

The proposed architecture is assessed through a remote robot arm teleoperation scenario. The robot arm used is a 7-DOF anthropomorphic manipulator (PA-10, Mitsubishi Heavy Industries). Details on the experimental setup can be found in [12]. The user is initially instructed to move his/her arm toward randomly selected points in the 3-D space, as shown in Fig. 1, to collect training data. As soon as the model is trained, the real-time operation phase takes place. The user is instructed to move randomly the arm in the 3-D space, having visual contact with the robot arm. The user's hand trajectory in the 3-D space is depicted in Fig. 4, along with the robot trajectory based on the EMG-based-decoding method, during the real-time-operation phase. The proposed system was tested by three subjects in total with similar results.

B. Efficiency Assessment

The root-mean-squared error (RMSE) and the correlation coefficient (CC), whose definitions can be found in [12], will be used to assess method performance. These criteria were used since they can well

TABLE I
METHOD EFFICIENCY COMPARISON BETWEEN HIGH AND LOW-DIMENSIONAL REPRESENTATION OF DATA IN CARTESIAN SPACE

Representation dimensionality	CC_x	CC_y	CC_z	$RMSE_x$ (cm)	$RMSE_y$ (cm)	$RMSE_z$ (cm)
Low	0.96±0.03	0.97±0.02	0.95±0.03	2.25±0.45	2.53±0.39	1.95±0.45
High	0.83±0.04	0.79±0.05	0.82±0.05	8.82±1.97	9.01±2.12	14.12±2.53

Mean and standard deviation values for ten experiments are reported.

TABLE II
EFFICIENCY COMPARISON BETWEEN THE PROPOSED STATE-SPACE MODEL AND THE LINEAR-FILTER METHOD IN CARTESIAN SPACE

Decoding model	CC_x	CC_y	CC_z	$RMSE_x$ (cm)	$RMSE_y$ (cm)	$RMSE_z$ (cm)
State-space	0.96	0.97	0.95	2.25	2.53	1.95
Linear-filter	0.87	0.78	0.79	9.45	14.14	12.34

Mean values for ten experiments are reported.

describe the correlation of the estimated trajectory with respect to the real-arm motion. Real and estimated motion data were recorded for ten experiments, of 30 s each. Using the hand kinematics to transform joint angle estimates to hand trajectory in the Cartesian space, the criteria values were computed and are listed in Table I. A chi-square significance test to assess the system accuracy across different subjects and in different cases was also conducted. The test hypothesized that the error between the estimated and real-hand trajectories along the three axes of the 3-D space was characterized by a zero-meaned distribution of 2 cm variance. The probabilities concluded by the test were 0.78, 0.81, and 0.89, respectively, which means that the hypothesis that the error is always small was quite probably true.

A characteristic of the method that is worth assessing is the use of the low-dimensional representation of the muscle activation and human kinematic variables. In order to conclude if this approach finally facilitated the decoding method, we tried to estimate a model given by (5) using the high-dimensional data for muscle activations and human joint angles. The same training and testing data were used, as described earlier, for the comparison to be meaningful. The results are shown in Table I. One can see that the decoding method using the high-dimensional data concluded to a model whose results were worse than those of the proposed decoding in the low-dimensional space.

Another parameter worth assessing is the type of the model used (i.e., linear model with hidden states). The authors feel that a comparison with an algorithm that is well-known and of similar complexity is rational. Thus, a comparison with the linear-filter method was done. The linear-filter method is a widely used method for decoding arm motions (especially when using neural signals), which has achieved exciting results so far [13]. Briefly, if Q_k are the kinematics variables decoded (i.e., joint angles) at time $t_k = kT$ and $\mathcal{E}_{i,k-j}$ are the muscle activation of muscle i at time t_{k-j} , the computation of the linear-filter entails finding a set of coefficients $\pi = [a \ f_{1,1} \ \dots \ f_{i,j}]^T$ so that

$$Q_k = a + \sum_{i=1}^v \sum_{j=0}^N f_{i,j} \mathcal{E}_{i,k-j} \quad (7)$$

where a is a constant offset, $f_{i,j}$ are the filter coefficients, v is the number of muscles recorded, and the parameter N specifies the number of time bins used. A typical value of the latter is 100 ms; thus, $N = 100$, for a sampling period of 1 ms [13]. The coefficients can be learned from training data using simple least-squares regression. In our case, for the

sake of comparison, the same training data were used for both the state-space model and the linear-filter, and after training, both models were tested using the same testing data as before. Values for RMSE and CC for these two methods are reported in Table II. It must be noted that the low-dimensional representation for muscle activations and kinematic was used since linear-filter behaved better using those kinds of data rather than using the high-dimensional data.

IV. CONCLUSION AND DISCUSSION

In this paper, a novel human-robot interface for robot teleoperation was introduced. EMG signals recorded from muscles of the upper limb were used for extracting kinematic variables (i.e., joint angles) in order to control an anthropomorphic robot arm in real time. The novelty of the method proposed here can be centered around two main issues that are discussed in the following.

First, the dimensionality reduction is quite significant, since it not only revealed some interesting aspects regarding the 3-D movements studied, but it also aided the matching between the EMG signals and motion since signal correlations were extracted, and the number of variables was drastically reduced. The latter led to the fact that a simple linear model with hidden states proved quite successful in mapping EMG signals to arm motion. The fact that the 3-D arm motion is somehow constrained by the use of only two independent variables that describe arm motion does not hinder the applicability of the method. This is based on the suggestion of motor synergies, which allows those 2 DOFs decoded to be represented back in the high-dimensional space, where four human joint angles are actuated, concluding to motion of the human arm in the 3-D Cartesian space with limited though workspace.

The second important issue presented here is that, to the best of our knowledge, this is the first time a continuous profile of the 3-D arm motion 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; however, they are constrained to isometric movements, single DOF, or very smooth motions [5]. In this paper, the method was tested for motions in the 3-D space, with variable speed profiles. Moreover, this paper proposes a methodology that can be easily trained to each user and takes little time to build the decoding model, while the computational load during real-time operation is negligible.

REFERENCES

- [1] 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.
- [2] A. V. Hill, "The heat of shortening and the dynamic constants of muscle," in *Proc. R. Soc. Lond. Biol.*, 1938, pp. 136–195.
- [3] F. E. Zajac, "Muscle and tendon: Properties, models, scaling, and application to biomechanics and motor control," in *Proc. Crit. Rev. Biomed. Eng.*, vol. 17, Boca Raton, FL: CRC, 1989, pp. 359–411.
- [4] 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.
- [5] 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, pp. 291–300, 1995.
- [6] A. d'Avella, A. Portone, L. Fernandez, and F. Lacquaniti, "Control of fast-reaching movements by muscle synergy combinations," *J. Neurosci.*, vol. 25, no. 30, pp. 7791–7810, 2006.
- [7] 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.
- [8] J. R. Cram and G. S. Kasman, *Introduction to Surface Electromyography*. Gaithersburg, MD: Aspen, 1998.
- [9] I. T. Jolliffe, *Principal Component Analysis*. New York: Springer-Verlag, 2002.
- [10] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, NJ: Prentice-Hall, 1999.
- [11] 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," presented at the IEEE/ASME Int. Conf. Adv. Intell. Mechatronics, Switzerland, 2007.
- [12] 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.
- [13] J. M. Carmena, M. A. Lebedev, R. E. Crist, J. E. O'(tm)Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, 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, pp. 001–016, 2003.