

An Upper Limb Kinematic Graphical Model for the Prediction of Anthropomorphic Arm Trajectories

B. A. P. S. Noronha, M. Wessels, A. Q. L. Keemink, A. Bergsma, H. F. J. M. Koopman

Abstract—This study approaches the use of Bayesian networks to model the human arm movement in an anthropomorphic manner for the control of an upper limb assistive robot. The model receives as input a desired wrist position and outputs three angles, the swivel angle (i.e. the angle that represents the rotation of the plane formed by the upper and lower arm around the axis that passes through the shoulder and wrist) and two angles corresponding to two degrees of freedom of the sternoclavicular joint (elevation/depression and protraction/retraction). These angles, together with the wrist position, fully describe the position of the shoulder and the elbow. A set of recording sessions was conducted to acquire human motion data to train the model for four different activities of daily living. Performance was measured by the elbow and shoulder joints' end-point errors and Pearson's r . The model was able to predict accurately elbow movement (mean error 0.021 ± 0.020 m, Pearson's r 0.86–0.99) and shoulder movement (mean error 0.014 ± 0.011 m, Pearson's r 0.52–0.99) for wrist trajectories that fall in the set of training data. It was also able to create new motions that were not in the set of training data, with a better accuracy for the elbow joint (mean error 0.042 ± 0.025 m, Pearson's r 0.59–0.99) and an average accuracy for the shoulder joint (mean error 0.026 ± 0.012 m, Pearson's r -0.12–0.99). The proposed model presents a novel method to solve the inverse kinematics problem in the scope of the human upper limb. It can also create movement out of its training data, although not highly correlated with the trajectory performed by a human.

I. INTRODUCTION

Understanding the control of arm movements by the central nervous system (CNS) is one of the most investigated areas in the study of human motor control. One of the main obstacles is the Bernstein problem [1], which poses the question: how can the CNS coordinate such a complex and redundant motor system with so many degrees of freedom (DOFs) to interact with the surrounding environment so efficiently? This problem can be regarded as an inverse kinematics (IK) problem, which involves calculating the joints angles of a kinematic chain from the end-effector position [2]. The principal issue with the IK problem is that, when considering a redundant kinematic chain, there are infinite solutions. Several models and methods have been proposed to try to explain and replicate how the CNS resolves kinematic redundancy and to provide a solution to the IK problem. Such methods involve using the Jacobian in

Dept. of Biomechanical Engineering, Faculty of Engineering Technology, University of Twente, Drienerloolaan 5, 7522 NB The Netherlands, m.wessels@utwente.nl

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 644000. This result reflects only the author's view and the Commission is not responsible for any use of the information in this result.

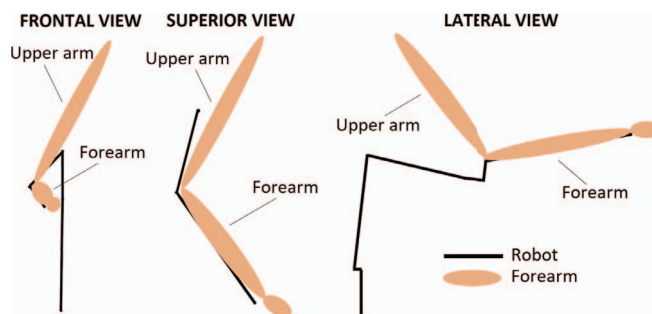


Fig. 1: Graphical representation of the robot used. Only the forearm of the subject is mechanically supported.

numerical iterative techniques [3–5] or virtual spring-damper systems [6], probabilistic approaches [7] or using cost functions [8–15]. However, these methods have some limitations and drawbacks such as not reproducing the movement in an anthropomorphic manner, suffering from singularities, having a slow convergence to a final posture or not being applicable to a wide range of tasks [3–15].

An important application of these models is the control of robots for assisting humans in daily life (e.g. helping patients that have lost motor function to recover it through rehabilitation). The work presented here was done under the scope of the eNHANCE project [16]. Its goal is to develop an interface that combines multiple inputs (e.g. user intention, electromyography (EMG) and mechanomyography (MMG) signals) to deliver support by means of an assistive upper limb robot (Fig. 1). The system can be used to aid patients that have partially lost the ability to move their upper right limb. The robot has five motors that actuate five DOFs. These are not DOFs of the arm itself because the robot supports only translation and rotation (no wrist supination) of the forearm. Specifically, the forearm is supported on the robot (the robot is not an exoskeleton, it is not worn), being the connecting link between the latter and the upper limb. The arm movement can be represented by four rotational DOFs: three in the shoulder and one in the elbow.

In this paper, we present an upper limb computational model to be used in the control of the robot. We extended the probabilistic approach developed by Artemiadis et al. [7] to tackle the IK problem. From collected joints' position data, they constructed a graphical Bayesian model to describe the probability distributions of the joints angles. Subsequently, an objective function that places the arm in a configuration such that its joint probabilities are maximised is defined. This function is used in an IK closed-loop algorithm for moving a robot arm. Their model presents several advantages: negligible computational time; anthropomorphic resulting con-

figuration; generation of anthropomorphic motions outside the set of training data; no knowledge about the muscle and joints dynamic properties (viscosity, stiffness and co-contraction) is required. A limitation to their model is, however, that it cannot generate task-specific motion. The graphical model is constructed only from joints angles, hence its input is a joint angle, which in the 3D space does not specify an end-effector position. We built upon their approach by using the wrist's position as input and receiving as output the corresponding arm configuration. The resulting model shall be used to predict the trajectories the robot arm must carry to perform activities of daily life.

II. METHODS

In this section, we describe how the model was constructed and how its performance was evaluated. There are four main steps: a) Collecting motion data from healthy subjects for training the graphical model and evaluating its performance; b) Constructing the model and defining the upper limb kinematics; c) Determining the arm trajectory given the wrist trajectory; d) Evaluate the model performance.

A. Motion capture and performed experiments

First, we placed reflective markers on the subjects' joints to be able to record their positions. Afterwards, the subjects performed four tasks, thus obtaining the motion data corresponding to each task.

Motion capture and marker placement: The system used for measuring the motion of the subjects was OptiTrack Motive ®. The hardware of this system comprises a set of infra-red cameras and reflective markers. The markers were placed directly on the skin and on bony landmarks of each joint to decrease artefacts due to skin movement/stretch. Ten markers were used in total: three on the wrist, three on the elbow, three on the shoulder and one on the sternoclavicular joint (Fig. 2a). The motion capture system returns each marker's 3D position. The joint's position is taken as the average position of each joint's three markers.

Subjects: Ten subjects (2 females, 8 males, age range 21–27, mean age 24.1 ± 1.5) were recruited by word of mouth through the research personnel at the University of Twente. Subjects were excluded if they had motor impairment(s).

Tasks: A set of four tasks was selected based on their use in upper limb rehabilitation and usefulness in everyday life [17], [18]:

- Task 1: Reach and grasp laterally (Fig. 2b) 5 mugs placed equidistantly on a semi-circle in front of the subject, clockwise (2×). After each grasp, return the hand to its initial position.
- Task 2: Similar to Task 1, except the mugs were grasped at the top (Fig. 2c).
- Task 3: Reach and grasp laterally an open bottle of water in front of the subject and drink a small quantity from it (5×). After each grasp, return the hand and the bottle to their original positions.
- Task 4: Turn the pages of a magazine using the right hand (during 30 s).



Fig. 2: (a) marker placement positions; (b) example of lateral grasp; (c) example of grasp over the top.

The subjects were sitting at a table and instructed to maintain their back straight and perform the movements naturally.

B. Constructing the upper limb model

Kinematic model of the upper limb: A model with seven DOF was considered (Fig. 3): two in the sternoclavicular (SC) joint ($\theta_{1,2}$); three in the glenohumeral (GH) joint ($\theta_{3,4,5}$); two in the elbow (E) joint ($\theta_{6,7}$). For representing the DOFs, Denavit-Hartenberg (DH) parameters [19], the wrist position and the swivel angle were used. The swivel angle φ_s represents the angular rotation of the elbow around an axis that passes through the shoulder and wrist joints, relative to a downward pointing vertical. The first two DOFs were represented with the DH parameters (Table I), the 3rd, 4th and 5th using φ_s , and the 6th and 7th using φ_s together with the wrist position. It was decided that joints angles θ_{3-6} would not be used in the graphical model to avoid gimbal lock, and θ_7 because wrist pronosupination is not supported by the robot.

In the DH convention, there is a frame associated to each DOF. Each frame's position and orientation can be determined using the DH parameters and considering homogeneous transformations between frames. The homogeneous transformation between frame $i-1$ and i can be represented

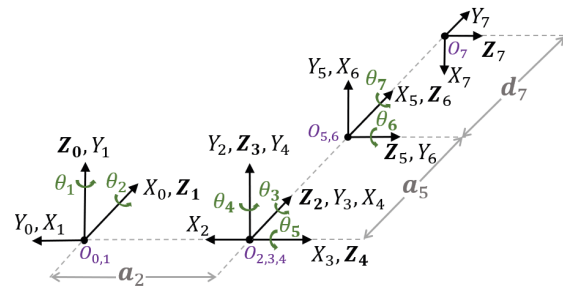


Fig. 3: Kinematic chain of the upper limb. $O_{0,1}$ corresponds to the centre of the SC joint, $O_{2,3,4}$ to the centre of the GH joint, $O_{5,6}$ to the centre of the E joint and O_7 to the centre of the W joint. a_2, a_5 and d_7 represent the length of the clavicle, upper arm and forearm, respectively.

TABLE I: Relevant DH parameters for the kinematic chain in Fig. 3.

i	a_i	α_i (°)	d_i	θ_i
1	0	90	0	θ_1
2	a_2	0	0	θ_2

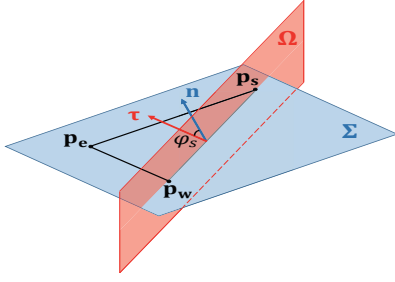


Fig. 4: Illustration of planes Σ and Ω and of the swivel angle φ_s .

by

$$T_{i-1}^i = \begin{bmatrix} c_i & -s_i c_{\alpha_i} & s_i s_{\alpha_i} & a_i c_i \\ s_i & c_i c_{\alpha_i} & -c_i s_{\alpha_i} & a_i s_i \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where $c_i = \cos \theta_i$, $s_i = \sin \theta_i$, $c_{\alpha_i} = \cos \alpha_i$, $s_{\alpha_i} = \sin \alpha_i$. The transformation T_j^i between frames j and i , where $j \neq i-1$, is obtained by the chain multiplication of the transformation matrices of consecutive frames. We shall now explore how to calculate the necessary angles given the joints positions and vice-versa.

The position of the SC, GH, E and W joints are represented by vectors $\mathbf{p}_{SC}, \mathbf{p}_s, \mathbf{p}_e, \mathbf{p}_w \in \mathbb{R}^3$, respectively. The coordinates of these vectors are in the 0-frame coordinate system. Vector \mathbf{p}_s coincides with the position of frames 2, 3 and 4. Therefore, the last column of matrices T_0^i , $i = \{2, 3, 4\}$ coincides with \mathbf{p}_s . By equating that column to $[\mathbf{p}_s^T \ 1]^T$, we obtain expressions for the angles θ_1 and θ_2

$$\theta_1 = \arctan2(p_{sy}, p_{sx}) \quad (2)$$

$$\theta_2 = \arctan2(p_{sz}, \sqrt{p_{sx}^2 + p_{sy}^2}) \quad (3)$$

The plane Σ that is defined by the GH, E and W joints has normal vector \mathbf{n}

$$\mathbf{n} = \frac{(\mathbf{p}_e - \mathbf{p}_s) \times (\mathbf{p}_w - \mathbf{p}_s)}{\|(\mathbf{p}_e - \mathbf{p}_s) \times (\mathbf{p}_w - \mathbf{p}_s)\|} \quad (4)$$

The plane Ω that passes through the GH and the W joints and is vertical, i.e., contains the vector $\mathbf{u}_z = [0, 0, 1]^T$, has normal vector $\boldsymbol{\tau}$

$$\boldsymbol{\tau} = \boldsymbol{\rho} \times \mathbf{u}_z \quad (5)$$

where

$$\boldsymbol{\rho} = \frac{\mathbf{p}_w - \mathbf{p}_s}{\|\mathbf{p}_w - \mathbf{p}_s\|}, \quad (6)$$

The φ_s is defined as the angle between \mathbf{n} and $\boldsymbol{\tau}$ and can be calculated as

$$\varphi_s = \arccos(\mathbf{n} \cdot \boldsymbol{\tau}) \quad (7)$$

Vector \mathbf{p}_s can be calculated from θ_1 and θ_2 by equating the last column of T_0^2 to $[\mathbf{p}_s^T \ 1]^T$, obtaining:

$$p_{sx} = a_2 c_1 c_2 \quad (8)$$

$$p_{sy} = a_2 s_1 c_2 \quad (9)$$

$$p_{sz} = a_2 s_2 \quad (10)$$

The elbow moves according to two separate constraints, (a) the connection to the shoulder through the upper arm and (b) the connection to the hand through the forearm. Therefore, it moves on two spheres, one centred around \mathbf{p}_s with radius r_{ua} (upper arm length) and another centred around \mathbf{p}_w with radius r_{fa} (forearm length), described by

$$\|\mathbf{p}_e - \mathbf{p}_s\| = r_{ua} \quad (11)$$

$$\|\mathbf{p}_e - \mathbf{p}_w\| = r_{fa} \quad (12)$$

The intersection of these two spheres is a circle that contains the points where the elbow is allowed to move. The plane that contains this circle is plane Π , with normal vector $\boldsymbol{\rho}$ and is described by

$$[\boldsymbol{\rho}^T \ \sigma] \begin{bmatrix} \mathbf{p}_e \\ 1 \end{bmatrix} = 0 \quad (13)$$

where

$$\sigma = \frac{r_{fa}^2 - r_{ua}^2 + \|\mathbf{p}_s\|^2 - \|\mathbf{p}_w\|^2}{2\|\mathbf{p}_w - \mathbf{p}_s\|} \quad (14)$$

The normalised projection of $-\mathbf{u}_z$ on plane Π is

$$\boldsymbol{\zeta} = \frac{\boldsymbol{\rho} \times \boldsymbol{\tau}}{\|\boldsymbol{\rho} \times \boldsymbol{\tau}\|} \quad (15)$$

Plane Π is hence spanned by two normalised vectors, $\boldsymbol{\zeta}$ and $\boldsymbol{\tau}$. The vector \mathbf{n} can now be described as

$$\mathbf{n} = -\sin(\varphi_s) \boldsymbol{\zeta} + \cos(\varphi_s) \boldsymbol{\tau} \quad (16)$$

and the equation that describes plane Σ is then

$$[\mathbf{n}^T \ -(\mathbf{n} \cdot \mathbf{p}_s)] \begin{bmatrix} \mathbf{p}_e \\ 1 \end{bmatrix} = 0 \quad (17)$$

The intersection of planes Π and Σ is a line L described by

$$\begin{bmatrix} \boldsymbol{\rho}^T \\ \mathbf{n}^T \end{bmatrix} \mathbf{p}_e = \begin{bmatrix} -\sigma \\ -(\mathbf{n} \cdot \mathbf{p}_s) \end{bmatrix} \quad (18)$$

If we consider the vectors \mathbf{V} and \mathbf{U} such that

$$\mathbf{V} = -(\mathbf{n} \cdot \mathbf{p}_s) \boldsymbol{\rho} - \sigma \mathbf{n} \quad (19)$$

$$\mathbf{U} = \mathbf{n} \times \boldsymbol{\rho} \quad (20)$$

then the solution of (18) is given by

$$p_{ex} = \frac{V_z + p_{ey} U_x}{U_y} \quad (21)$$

$$p_{ez} = -\frac{V_x - p_{ey} U_z}{U_y} \quad (22)$$

Plugging eqs. (21) and (22) in either eq. (11) or eq. (12) gives us a quadratic equation with one single variable, p_{ey} . Choosing the first sphere equation, the coefficients of the quadratic equation $ap_{ey}^2 + bp_{ey} + c = 0$ become

$$a = \|\mathbf{U}\|^2 \quad (23)$$

$$b = 2(U_x V_z - U_z V_x - U_y (\mathbf{U} \cdot \mathbf{p}_w)) \quad (24)$$

$$c = U_y^2 (\|\mathbf{p}_s\|^2 - r_{fa}^2) + 2U_y (V_x p_{wz} - V_z p_{wx}) + V_x^2 + V_z^2 \quad (25)$$

and the solution that results in an anatomically correct elbow position is chosen, i.e. so that the elbow does not flex inwards instead of outwards.

Graphical model: A graphical model is a diagrammatic representation of a probability distribution by means of a graph. In a probabilistic graphical model, the nodes of the graph represent random variables and the edges represent probabilistic relationships between them. A graph whose edges have directionality is known as a Bayesian Network (BN). In this study, an acyclical BN is used. The joint probability distribution of a BN is given by

$$p(X_1, \dots, X_K) = \prod_{k=1}^K p(X_k | F_{X_k}) \quad (26)$$

where F_{X_k} denotes the set of parents of node X_k . The parents of a node X_i are the nodes X_j, \dots, X_K that have an edge directed from them to X_i . If the node X_i does not have parents, it is called a root node and $p(X_i | F_{X_i}) = p(X_i | \emptyset) = p(X_i)$.

The random variables whose joint probabilities the model aims to approximate are $p_{w_x}, p_{w_y}, p_{w_z}, \varphi_s, \theta_1, \theta_2$, hence they are the nodes of the model. The value of each node can be calculated using inference methods: the root nodes are observed nodes, meaning their value is not resultant from any calculation but from an observation, and the other nodes' values are calculated by propagating throughout the network.

The BN is constructed by adapting the method presented by Lam et al. [20]. The more complex the model, the more accurate it is. However, more complex models are computationally expensive, and they may also overfit the data. Therefore, Lam et al. use the Minimum Descriptive Length (MDL) principle, which avoids this by making a trade-off between accuracy and complexity. The model's accuracy is the sum of the weight W of every node in the model

$$W(X_i, F_{X_i}) = \sum_{X_i, F_{X_i}} \left[p(X_i, F_{X_i}) \log_2 \left(\frac{p(X_i, X_j)}{p(X_i)p(F_{X_i})} \right) \right] \quad (27)$$

We added to the method of Lam et al. the restrictions that the maximum number of parents a node can have is two and the root nodes can only be $p_{w_x}, p_{w_y}, p_{w_z}$, which are the input to the model. A pair of graphical models was created for each of the performed tasks, given that each task implies a specific pattern of motion that is unique. In this way, the specificity of the network is improved depending on the desired motion. The pair of models consisted of a Trade-off (T-O) model, which is the one calculated following the MDL principle, and a Heavy model, represented by the maximum-weight BN. The data of one subject was used as validation, whereas the data of the rest of the subjects was used to train the model.

From a discrete to a continuous model: The probabilities that are used by the graphical model are empirically determined from discrete measurements and stored in large matrices, hence searching them is a time-consuming process. To avoid this, we used Gaussian Mixture Models (GMM) to approximate the discrete joint probabilities with continuous functions. A GMM is a linear combination of independent Gaussian distributions, where each Gaussian is a component

with an associated weight [21]. The GMMs are fitted using MATLAB's ® function `fitgmdist`. Given we do not know the underlying probability distribution, we do not know how many components each of the fitted GMMs should have. This is determined using the Akaike information criterion [22].

Used software: The software used in this study was MATLAB ®.

C. Reconstructing the arm configuration

The arm configuration is calculated step by step, i.e. for each discrete value of \mathbf{p}_w . The procedure taken in each step is as follows (Fig. 5):

1. Give the \mathbf{W} joint position as input to the graphical model.
2. Calculate the values of $\{\varphi_s, \theta_1, \theta_2\}$ by propagating through the calculated network.
3. Calculate the GH (eq. (8)) and E joint positions (eqs. (21), (22) and (23)).

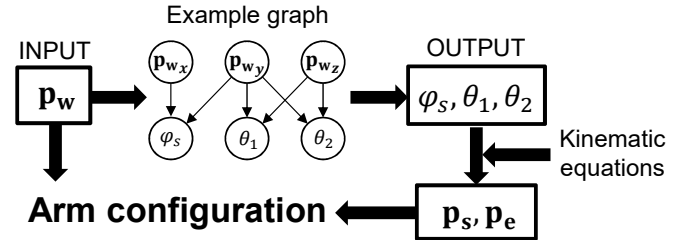


Fig. 5: Scheme of how the arm configuration is reconstructed. The graph shown is merely an example.

D. Performance evaluation

The performance of the models was evaluated regarding the elbow and shoulder joints positions, and not using the angles' values. This was because it is more intuitive to evaluate positions errors than angles errors (mainly when we are dealing with the swivel angle, which is not an angle easy to visualise). In addition, there are three angles and only two joints, making it easier to analyse the joints' positions.

The values of these variables given a wrist trajectory performed by a human were compared to those predicted by the models for the same wrist trajectory, for each task. Two measures were analysed: (1) mean error (Euclidean distance between the measured and predicted positions, averaged for all subjects within the same task) and (2) Pearson correlation coefficient between the measured and predicted values. These two measures complement each other in terms of the information they provide: the mean error indicates how far-off are the values compared (in average), whereas Pearson's r indicates how similar are both trajectories. An error of 5 cm was considered the threshold between a good and a poor error. This value was chosen because it is not large enough to prevent performing the task at hand.

We investigated whether the choice of a simpler, but lighter model (as stated by the MDL principle) is corroborated by the results. For this, we compared the computation time (independent-samples t-test) and performance of the T-O and the Heavy models, for each tasks' elementary motion.

We also evaluated whether the model was able to generate new motion by evaluating the performance for a validation trajectory collected from one subject not in the training set.

III. RESULTS

Out of the ten subjects, one was excluded due to failure of returning the hand to the original position and another due to data corruption. Therefore, the data of 7 subjects was used for training the model, and 1 subject for validation.

The computation time of each model is presented in Table II together with the respective weight scores. The Heavy models took, on average, 11% longer than their T-O counterpart ($P < 0.0001$).

Table III shows the mean errors for all the analysed variables. The mean error for the Heavy model was, on average, 0.011 m lower than for the T-O model.

Table IV shows Pearson's r . One coefficient was calculated for each of the position vectors' components. The Pearson's r of the Heavy model were, on average, 0.192 larger than those of the T-O model.

Fig. 6 shows the error variation throughout the first round of reaching movements in Task 1, in the validation case. It can be seen that the error increases during the reaching movement relatively to the resting position.

TABLE II: Average trajectory computation time (s) for each of the calculated models, for each task. The average was taken over 100 separate trajectory computations. Between parenthesis is the respective model weight.

COMPUTATION TIMES (s)				
Task				
	1	2	3	4
T-O	0.87 (11.53)	0.78 (8.75)	3.10 (8.73)	1.03 (8.20)
Heavy	0.92 (11.88)	0.92 (13.27)	3.40 (10.52)	1.16 (12.57)

TABLE III: Mean errors (m) for each trajectory, for the T-O and Heavy (H) models, averaged for all tasks. T – training trajectory, V – validation trajectory.

MEAN ERRORS									
Task									
		1		2		3		4	
		T-O	H	T-O	H	T-O	H	T-O	H
T	p_s	0.021 ± 0.018	0.018 ± 0.014	0.029 ± 0.037	0.017 ± 0.014	0.017 ± 0.010	0.013 ± 0.009	0.011 ± 0.007	0.009 ± 0.008
	p_e	0.026 ± 0.018	0.026 ± 0.026	0.041 ± 0.026	0.024 ± 0.021	0.021 ± 0.015	0.019 ± 0.014	0.043 ± 0.032	0.016 ± 0.016
V	p_s	0.040 ± 0.031	0.032 ± 0.016	0.047 ± 0.045	0.033 ± 0.017	0.027 ± 0.009	0.019 ± 0.010	0.019 ± 0.008	0.020 ± 0.007
	p_e	0.057 ± 0.038	0.046 ± 0.030	0.056 ± 0.066	0.030 ± 0.029	0.048 ± 0.020	0.035 ± 0.014	0.069 ± 0.024	0.058 ± 0.029

TABLE IV: Correlation between measured and predicted trajectories for T-O and Heavy (H) models, for each task. T – training, V – validation.

PEARSON'S r									
		Task							
		1		2		3		4	
		T-O	H	T-O	H	T-O	H	T-O	H
T	p_{s_x}	0.73	0.85	0.23	0.85	0.69	0.86	0.93	0.91
	p_{s_y}	0.79	0.82	0.80	0.86	0.89	0.91	0.83	0.82
	p_{s_z}	0.65	0.71	0.59	0.62	0.48	0.52	0.53	0.77
	p_{e_x}	0.99	0.99	0.98	0.99	0.99	0.99	0.97	0.99
	p_{e_y}	0.98	0.97	0.96	0.97	0.89	0.88	0.33	0.91
	p_{e_z}	0.95	0.92	0.84	0.86	0.94	0.97	0.80	0.97
V	p_{s_x}	0.68	0.89	-0.45	0.82	0.07	0.80	0.27	0.38
	p_{s_y}	-0.09	-0.12	0.10	-0.23	0.49	0.44	0.00	0.16
	p_{s_z}	0.67	0.81	0.45	0.53	-0.45	-0.24	-0.08	0.44
	p_{e_x}	0.96	0.99	0.92	0.99	0.99	0.98	0.59	0.68
	p_{e_y}	0.99	0.98	0.92	0.99	0.72	0.89	0.27	0.60
	p_{e_z}	0.92	0.92	0.78	0.80	0.97	0.99	-0.07	0.59

IV. DISCUSSION

A. The proposed model

An upper limb kinematic graphical model is proposed as an approach to resolve the redundancy in the IK problem of human arm motion. The model takes as input a desired wrist trajectory and predicts how a human would position the elbow and shoulder joints accordingly.

This is the first biomechanical graphical model that, given a desired wrist joint position, outputs a human-like arm configuration. In addition to the advantages stated in Sec. I, our model does not try to mimic a predefined human motion, but instead uses it to learn what can be categorised as anthropomorphic. Therefore, it can generalise for motions outside the training data, even though with lower similarity to real human motion. This is one of the great advantages of using a graphical model for modelling the upper limb kinematics.

B. Performance analysis

It was observed that for all tasks, the Heavy model performed better than the T-O model. Although it took longer for predicting the same trajectory, all computation times are low and in the same order of magnitude pairwise, hence the Heavy model can be preferred over the T-O. The reason why the MDL principle does not seem to apply here is that the models do not contain a vast set of stochastic variables. The rest of the discussion shall be regarding the Heavy model.

The performance when predicting a trajectory that was included in the training set was always better when compared to validation. The elbow trajectory prediction was very good for both trajectories, whereas the shoulder position prediction, although good for the training set, was not as good for the validation set due to its low Pearson's r . One might think this was due to overfitting. However, the prediction made by the T-O model, which accounts for overfitting, was considerably worse, meaning overfitting was not the issue. Nonetheless, the errors were still low (< 8 cm). This disparity

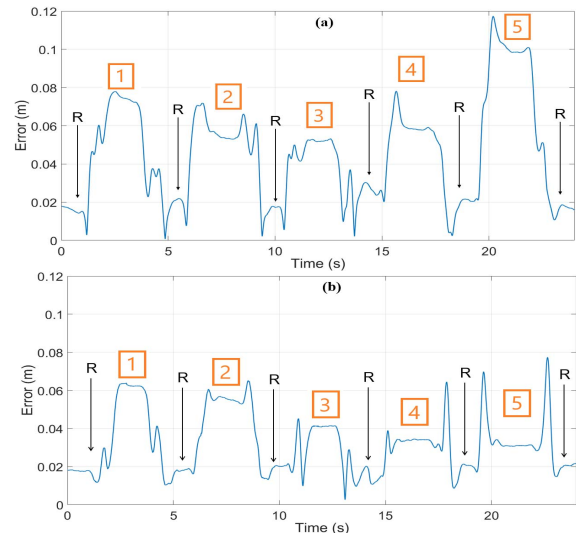


Fig. 6: Error of the predicted trajectory for Task 1 during validation. The "R" indicates resting positions and the orange numbers indicate the reaching movement. (a) – E joint; (b) – GH joint.

in performance between training and validation trajectories was expected due to a high variability in human motion, which results in differences in the motions performed by the validation and the training subjects.

It is interesting to note there was a lower error for the shoulder trajectory in spite of its lower correlation. This might be due to the lower range of motion of the shoulder, which does not allow for errors as large as the elbow's. The variability between subjects was not analysed.

We also observed a contrast between tasks in the achieved accuracy. Task 3 had the lowest error and variability for both the joints positions, probably due to the repetition of the same movement, as opposed to the rest of the tasks. Tasks 1 and 2 had a high variability because there was only one repetition of each reaching movement, therefore there is a higher variability in the values that were used in the training set, and it is more difficult for the model to construct an accurate network. The model had a higher error for the wrist positions where the hand was farther from the initial position (Fig. 6). This can be explained by the fact that the resting position was much more observed than the others, hence better learned by the model. Furthermore, Task 4 had very low correlation with the real trajectory, which, even if the mean errors are low, results in the worst performance. This is probably because flipping pages is an activity that greatly varies between subjects.

C. Limitations and improvements

The two main limitations observed were the accuracy of the motion capture system, which is prone to skin artefacts, and the variability in performance of Task 4.

The main aspects that can further improve the performance of the created model are: (1) increase the variability of movements included in the set to allow for better generalisation; (2) include dependence with the previous arm configuration so that a whole arm trajectory is predicted, instead of having each consecutive point predicted independently; (3) consider the SC joint as not fixed [23]; (4) create a single model that can be used for every task, therefore avoiding the need to specify the task prior to its realisation.

The current implementation did not have in account safety limitations (e.g. limiting range of motion). Although no extreme movements were observed, this should be taken into account when using this model. Furthermore, no studies were performed regarding performance for tasks that were not used in training.

The model can be further enhanced by categorising each subject according to its characteristics, e.g. arm lengths, arm width, height, etc. In this way, personalised networks can be created by collecting a large amount of data, improving the model's performance. Furthermore, a comparison with more established methods already referred in the Introduction would bring robustness to the method presented here.

V. CONCLUSION

The model developed in the present study presents a novel alternative for solving the upper limb inverse kinematics

problem. It was able to predict with a high accuracy the movement of a human arm contained in its training data, and to also generalize to motions not seen in the training data, albeit with a lower match to the observed human motion.

REFERENCES

- [1] N. A. Bernstein, *The Co-ordination and regulation of movements*, 1st ed., Pergamon Press Ltd., 1967.
- [2] D. Tolani, A. Goswami, and N. I. Badler, "Real-Time Inverse Kinematics Techniques for Anthropomorphic Limbs," *Graphical Models*, vol. 62, no. 5, pp. 353–388, 2000.
- [3] B. Kenwright, "Inverse Kinematics – Cyclic Coordinate Descent (CCD)," *Journal of Graphics Tools*, vol. 16, no. 4, pp. 177–217, 2012.
- [4] A. Aristidou and J. Lasenby, "FABRIK: A fast, iterative solver for the Inverse Kinematics problem," *Graphical Models*, vol. 73, no. 5, pp. 243–260, sep 2011.
- [5] S. R. S. Buss, "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods," *IEEE Transactions on Robotics and Automation*, vol. 132, no. 4, pp. 1–19, 2004.
- [6] S. Arimoto and M. Sekimoto, "Human-like movements of robotic arms with redundant DOFs: Virtual spring-damper hypothesis to tackle the bernstein problem," *IEEE International Conference on Robotics and Automation*, vol. 2006, no. May, pp. 1860–1866, 2006.
- [7] P. K. Artemiadis, P. T. Katsiaris, and K. J. Kyriakopoulos, "A biomimetic approach to inverse kinematics for a redundant robot arm," *Autonomous Robots*, vol. 29, no. 3, pp. 293–308, 2010.
- [8] E. Todorov, "Optimality principles in sensorimotor control," *Nature Neuroscience*, vol. 7, no. 9, pp. 907–915, 2004.
- [9] D. M. Wolpert, J. Diedrichsen, and J. R. Flanagan, "Principles of sensorimotor learning," *Nature Reviews Neuroscience*, vol. 12, no. 12, pp. 739–51, 2011.
- [10] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," *The Journal of Neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.
- [11] J. M. Hollerbach and K. C. Suh, "Redundancy Resolution of Manipulators through Torque Optimization," *IEEE Journal on Robotics and Automation*, vol. 3, no. 4, pp. 308–316, 1997.
- [12] Y. Uno, M. Kawato, and R. Suzuki, "Formation and control of optimal trajectory in human multijoint arm movement," *Biological Cybernetics*, vol. 61, no. 2, pp. 89–101, 1989.
- [13] J. Rasmussen, M. Damsgaard, and M. Voigt, "Muscle recruitment by the min/max criterion - A comparative numerical study," *Journal of Biomechanics*, vol. 34, no. 3, pp. 409–415, 2001.
- [14] J. Diedrichsen, R. Shadmehr, and R. B. Ivry, "The coordination of movement: optimal feedback control and beyond," *Trends in Cognitive Sciences*, vol. 14, no. 1, pp. 31–39, 2010.
- [15] C. M. Harris and D. M. Wolpert, "Signal-dependent noise determines motor planning," *Nature*, vol. 394, no. 6695, pp. 780–4, 1998.
- [16] "eNHANCE Project." [Online]. Available: <http://www.enhance-motion.eu/>
- [17] B. H. Dobkin, "Strategies for stroke rehabilitation," *Lancet Neurology*, vol. 3, no. 9, pp. 528–536, 2004.
- [18] R. J. Nudo, B. M. Wise, F. SiFuentes, and G. W. Milliken, "Neural Substrates for the Effects of Rehabilitative Training on Motor Recovery After Ischemic Infarct," *Science*, vol. 272, no. 22, pp. 1791–4, 1996.
- [19] M. W. Spong, S. Hutchinson, and M. Vidyasagar, "Forward Kinematics: The Denavit-Hartenberg Convention," in *Robot dynamics and control*, 2nd ed., 2004, ch. 3, pp. 61–78.
- [20] W. Lam and F. Bacchus, "Learning bayesian belief networks: An approach based on the MDL principle," *Computational Intelligence*, vol. 10, no. 3, 1994.
- [21] C. M. Bishop, *Pattern Recognition And Machine Learning*, 1st ed., Springer, 2006.
- [22] K. P. Burnham and D. R. Anderson, "Multimodel Inference: Understanding AIC and BIC in Model Selection," *Sociological Methods & Research*, vol. 33, no. 2, pp. 261–304, 2004.
- [23] S. V. Adamovich, P. S. Archambault, M. Ghafouri, M. F. Levin, H. Poizner, and A. G. Feldman, "Hand trajectory invariance in reaching movements involving the trunk," *Experimental Brain Research*, vol. 138, no. 3, pp. 288–303, 2001.