

# Admittance-based Upper Limb Robotic Active and Active-assistive Movements

Regular Paper

---

Cristóbal Ochoa Luna<sup>1\*</sup>, Mohammad Habibur Rahman<sup>1,2</sup>, Maarouf Saad<sup>1</sup>, Philippe S. Archambault<sup>2,3</sup> and Steven Bruce Ferrer<sup>1</sup>

1 Electrical Engineering Department, École de technologie supérieure, Montreal, Canada

2 School of Physical & Occupational Therapy, McGill University, Montreal, Quebec, Canada

3 Centre of Interdisciplinary Research in Rehabilitation (CRIR), Montreal, Quebec, Canada

\*Corresponding author(s) E-mail: cristobal.ochoa-luna.1@ens.etsmtl.ca

Received 06 October 2014; Accepted 19 February 2015

DOI: 10.5772/60784

© 2015 Author(s). Licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

---

## Abstract

This paper presents two rehabilitation schemes for patients with upper limb impairments. The first is an active-assistive scheme based on the trajectory tracking of predefined paths in Cartesian space. In it, the system allows for an adjustable degree of variation with respect to ideal tracking. The amount of variation is determined through an admittance function that depends on the opposition forces exerted on the system by the user, due to possible impairments. The coefficients of the function allow the adjustment of the degree of assistance the robot will provide in order to complete the target trajectory. The second scheme corresponds to active movements in a constrained space. Here, the same admittance function is applied; however, in this case, it is unattached to a predefined trajectory and instead connected to one generated in real time, according to the user's intended movements. This allows the user to move freely with the robot in order to track a given path. The free movement is bounded through the use of virtual walls that do not allow users to exceed certain limits. A human-machine interface was developed to guide the robot's user.

**Keywords** Upper limb rehabilitation, active and active-assistive rehabilitation, admittance, 7DOF exoskeleton, human-machine interface

---

## 1. Introduction

Upper extremity paralysis (the inability of a muscle or group of muscles to move voluntarily) occurs as a consequence of lesions such as injuries to the central or peripheral nervous system. Causes of injuries include strokes, sports mishaps and car or occupational accidents, among others. According with the US National Stroke Association®, approximately 80% of stroke survivors experience hemiparesis, which causes weakness or an inability to move one side of the body. Spasticity affects roughly 40% of stroke survivors and is characterized by stiff or tight muscles that constrain movement [1]. The primary treatment for these cases is rehabilitation, i.e., the process of helping an individual achieve the highest level of independence and quality of life possible, physically, emotionally, socially

and spiritually. Where rehabilitation is possible, it involves many hours of highly skilled procedures. The number of people capable of providing such support is insufficient for adequately covering all patients that need help. Recovery progression varies greatly and while certain people recover relatively quickly, for others, recovery can take a long time and can even be a lifelong process.

The use of robotic assistance in rehabilitation is an important point of interest that is attracting significant attention [2, 3]. Many studies have shown that robotic rehabilitation can be effective at improving arm function following a stroke [4-8]. However, in studies where participants received an equivalent amount of either robotic or conventional therapy, the improvements in arm function were no longer significant [8]. One reason for this may be that earlier studies used robotic devices with a limited number of degrees of freedom, such as planar manipulandums. Indeed, a recent study [9] demonstrated a small but significant improvement in arm function following therapy involving a 7-degree-of-freedom robot (shoulder, elbow and hand opening) when compared to an equivalent amount of conventional therapy. Thus, there is a clinical need for continuing the development of robotic devices that can assist with naturalistic arm movements.

The ETS - Motion Assistive Robotic-exoskeleton for Superior Extremity (ETS-MARSE) was developed in our laboratory [10-13] in order to provide rehabilitation assistance for the upper limbs. Currently, it comprises a seven-degree-of-freedom (DOF) exoskeleton, designed to cope with the full motion capabilities of the human arm; that is, at the shoulder, elbow and wrist levels, combined or individually. Additionally, it contains a graphical user interface that uses a virtual environment with different rehabilitation exercises. To date, work in this regard has been developed to the point of passive rehabilitation, which means that the exoskeleton executes movements along predefined trajectories, while moving the subject's arm with it to help improve the range of movement. This paper describes the next rehabilitation step: active-assistive and active rehabilitation [14-16].

The methodology presented in this paper combines the advantages of the trajectory tracking control as the inner loop control of the robot and an admittance-based trajectory modifier that performs the outer force control, and which is responsible for the interaction between the robot and user. The admittance is used as a means of modifying the desired trajectory [17, 18]. Due to the nonlinear nature of the ETS-MARSE, a nonlinear control technique that has inherent adaptive characteristics has been used, i.e., virtual decomposition control (VDC). VDC is a relatively new approach that can reduce the mathematical complexity of multi-degree of freedom robots and has an adaptive nature that helps to deal with different patients' dynamics [19]. As robots become more complex, the computation of robot dynamics is proportional to the fourth power of the number

of DOF in Lagrangian dynamics [20]. VDC uses the dynamics of virtually created subsystems to conduct control design; this decomposition yields simpler dynamic equations in formulation and implementation [21]. In previous research with ETS-MARSE, we used different non-linear control strategies, such as computed torque (CTC) [10] and sliding mode control (SMC) [12]. Though SMC performed well, it showed an approximately 1.5 times increase in end-effector tracking error when conditions of the subject changed and perturbations were presented, while the same exercise was performed. It also presented the associated chattering, which can be reduced but not eliminated [12]. Conversely, an advantage of the VDC lies in its inherent adaptive capabilities, in the form of internal parameter estimation. It was shown in our previous studies that in direct comparison with CTC and PID, VDC can successfully cope with the physiological variations between different subjects [22]. Although differences in tracking error were minimal and below the needs dictated by the application, PID showed an average RMS error standard deviation more than three times larger than for VDC; for CTC, this was more than 10 times that of VDC. VDC however, due to control gains and parameter estimation gains, involves a complicated process of initial tuning during the controller design process.

In order to obtain the user's motion intention and modify the robot's trajectory tracking accordingly, the admittance concept is used. This approach has also been proven adequate for high ratio transmissions systems [18, 23]. The robot takes information from the human-machine interaction, in this case through contact force feedback between the structure and the person's arm and uses it to actively modify its rehabilitation task.

In recent years, admittance control for rehabilitation has come to represent a well-known approach and different schemes have been proposed [17, 24-26]. One of the main advantages of this type of control is that it offers the possibility of active rehabilitation. This is because it contributes to the rehabilitation process by allowing task repeatability and provides relevant data for the quantification of different outcomes, such as improvement, effort, etc. However, this approach is limited by the number of degrees of freedom of the robotic device [24] or has only been employed with end-effector type robots [17, 25, 26]. The contribution of this paper is to present both active-assistive and active rehabilitation task schemes for the 7DOF ETS-MARSE exoskeleton. These tasks are based on path-tracking game-like exercises that have been successfully employed in neuromuscular rehabilitation [27]. The rehabilitation robot works in two different scenarios, both of which make use of a virtual environment as an interface to visualize the movement to be performed. In the first scenario (active-assistive rehabilitation), the robot follows a predefined movement path; however, the system measures the user's force opposing the predetermined move-

ment and allows a certain adjustable degree of trajectory variation. In the active mode, the robot allows the user to freely move in the workspace while trying to achieve a target. In this case, the user’s movement is constrained by means of virtual walls that restrain him from exceeding certain limits and mark the point where the system will take corrective or assistive actions to help resolve a collision conflict. At all times, the robot appears weightless to the user and helps to support their arms if they stop in the middle of an exercise. These schemes may encourage the patient to actively participate in the exercise by attempting to follow the target trajectory and motivating them to improve their level of accomplishment.

This paper is organized as follows: section 2 describes the ETS-MARSE and the setup of the system. In section 3, a general description of the controller (VDC) is given. In section 4, the implementations of the active-assistive and active rehabilitation schemes are described. Section 5 summarizes the experimental setup and presents the results. Finally, section 6 provides conclusions and future work perspectives.

## 2. System Overview

The exoskeleton robot arm ETS-MARSE is shown in Figure 1. It is an open chain, all-revolute, 7DOF wearable manipulator-like robot; its joints were designed to correspond to the principal degrees of freedom of the human arm. These are:

- 3DOF at shoulder level for horizontal flexion/extension, vertical flexion/extension and internal/external rotation
- 1DOF for elbow flexion/extension
- 1 DOF for forearm pronation/supination
- 2DOF at the wrist level for radial/ulnar deviation and flexion/extension



Figure 1. Subject wearing the ETS-MARSE robotic arm

The exoskeleton structure has a six-axis force sensor mounted on the end-effector (tip) of the robot, which is at the base of the handle where the user’s hand must be

positioned. Each joint is driven by a brushless DC motor that has a Hall effect sensor used for position feedback of the joints. The motors are incorporated with harmonic drives; the specifications for the motors and harmonic drives can be found in Appendix A. A backplane card collects analogue and digital signals and connects through the proper interface cards to an NI PXI-7813R (remote input-output card) placed on an NI PXI-1031 chassis. The card has an integrated FPGA in which low-level control is performed, a PI controller for the current loop of the motors, position feedback via the Hall effect sensors and the collection of force sensor inputs. A PXI-8108 controller, which performs high-level control (the robot operating system and the trajectory tracking controller) is also mounted on the chassis. In this case, we are using the nonlinear control technique known as virtual decomposition control (VDC) [19], with an update rate of 1 ms. A description of the implementation of the VDC is outlined in section 3 of this paper.

A PC runs the human-machine interface virtual environment in which the robot wearer visualizes the rehabilitation exercise. The current architecture of the system is depicted in Figure 2. Since the system is a human-machine interface, safety is a major concern and for this reason, a hardware emergency stop button is built into it.

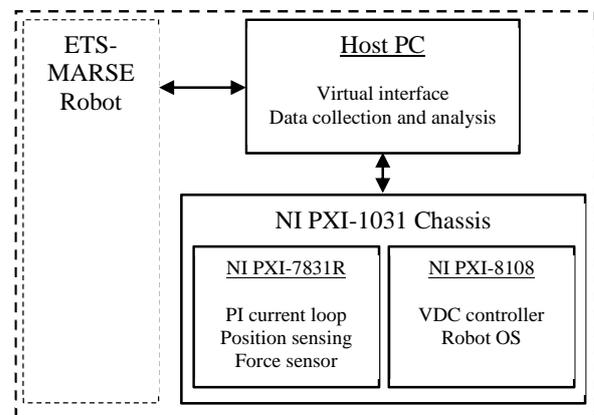


Figure 2. ETS-MARSE system hardware overview

## 3. Virtual Decomposition Control of ETS-MARSE

The control technique we used is presented in this section, but only the steps required to implement VDC are listed. This technique presents an effective approach for dealing with high DOF robots and multi-robot control. A detailed analysis is beyond the scope of this paper; readers interested in such an analysis are referred to other works that address this technique [28, 29]. The frame attachment for the DOF of the ETS-MARSE, following the method summarized in [30], is shown in Figure 3. The resulting Denavit-Hartenberg modified parameters are obtained, as described in [30] and are summarized in Table 1.

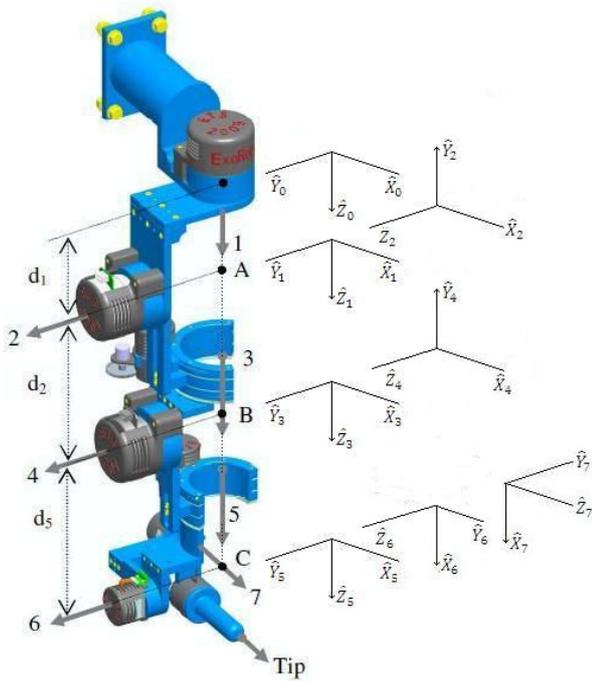


Figure 3. 7DOF exoskeleton robot arm

$i$	$\alpha_{i-1}$	$a_{i-1}$	$d_i$	$\theta_i$
1	$0^\circ$	0	$d_1$	$\theta_1$
2	$-90^\circ$	0	0	$\theta_2$
3	$90^\circ$	0	$d_3$	$\theta_3$
4	$-90^\circ$	0	0	$\theta_4$
5	$90^\circ$	0	$d_5$	$\theta_5$
6	$-90^\circ$	0	0	$\theta_6-90^\circ$
7	$-90^\circ$	0	0	$\theta_7$

Table 1. ETS-MARSE Denavit-Hartenberg modified parameters

### 3.1 Robot Subsystem Decomposition

The first step in the VDC analysis covers the virtual decomposition of the robot; in our case, we can view the exoskeleton as a single open chain. This results in decomposition into fourteen subsystems, as illustrated in Figure 4.

### 3.2 Kinematics

From the resulting parameters shown in Table 1 and the decomposition shown in Figure 4, the homogeneous transformation matrices of the system are obtained. From these, the force/moment transformation matrices of the robot are calculated as follows:

$${}^A U_B = \begin{bmatrix} {}^A R_B & 0_{3 \times 3} \\ ({}^A r_{AB} \times) {}^A R_B & {}^A R_B \end{bmatrix} \in \mathbb{R}^{6 \times 6} \quad (1)$$

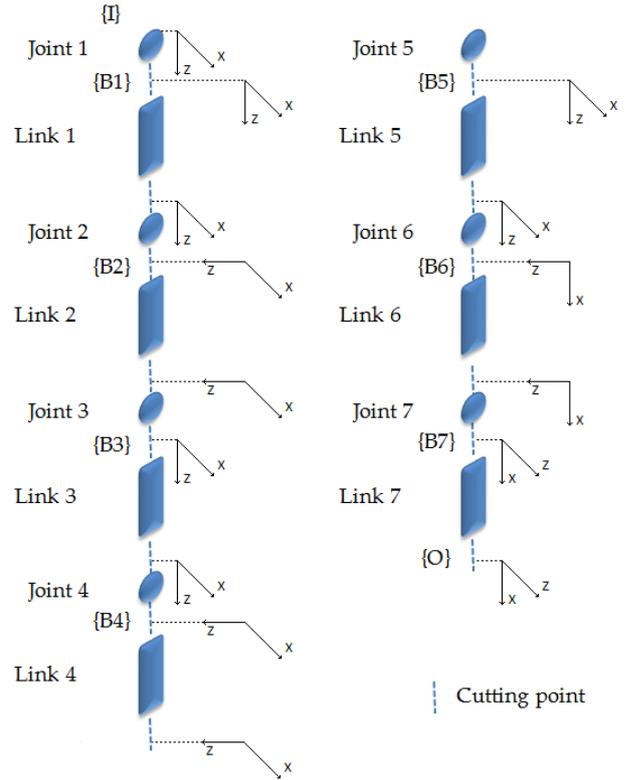


Figure 4. Virtual decomposition in 14 subsystems

where  ${}^A R_B$  represents the rotation matrix from one frame to the next and  ${}^A r_{AB}$  is the distance between frames, and:

$$({}^A r_{AB} \times) = \begin{bmatrix} 0 & -{}^A r_{AB}(3) & {}^A r_{AB}(2) \\ {}^A r_{AB}(3) & 0 & -{}^A r_{AB}(1) \\ -{}^A r_{AB}(2) & {}^A r_{AB}(1) & 0 \end{bmatrix} \quad (2)$$

where  ${}^A r_{AB}(i)$  represents the  $i_{th}$  term of the  ${}^A r_{AB}$  vector.

We calculate each force/moment transformation matrix  ${}^{B_i} U_{B_{i+1}}$  for  $i=1, \dots, 6$ .

The next step is to calculate  ${}^{B_i} V$ , the linear/angular velocity vectors of the  $B_i$  frames (with  $i=1, \dots, 7$ ) as follows:

$${}^{B_i} V = \begin{bmatrix} {}^{B_i} v \\ {}^{B_i} \omega \end{bmatrix} \in \mathbb{R}^6 \quad (3)$$

where  ${}^{B_i} v$  is the linear velocity vector and  ${}^{B_i} \omega$  is the angular velocity vector of the corresponding frame. With the previous linear/angular velocity vectors, the augmented velocity vector is formed as:

$$\mathcal{V}_7 = [\dot{q}_1 \quad \dots \quad \dot{q}_7 \quad {}^{B_1} V^T \quad \dots \quad {}^{B_7} V^T]^T \quad (4)$$

Equation (4) can be written in the following form:

$$\mathcal{V}_7 = J_7^* \dot{q} \quad (5)$$

where  $\dot{q} = [\dot{q}_1 \ \dots \ \dot{q}_7]^T$  is the velocity vector and  $J_7$  is the VDC-Jacobian matrix of the system:

$$J_7 = \begin{bmatrix} & I_7 & & & \\ z & 0_6 & \dots & 0_6 & \\ {}^{B_1}U_{B_2}^T z & z & \ddots & \vdots & \\ \vdots & \ddots & \ddots & 0_6 & \\ {}^{B_1}U_{B_7}^T z & \dots & {}^{B_6}U_{B_7}^T z & z & \end{bmatrix} \quad (6)$$

with  $z = [0 \ 0 \ 0 \ 0 \ 0 \ 1]^T$ ,  $I_7$  is a  $7 \times 7$  identity matrix and  $0_6$  a  $6 \times 1$  zero vector.

Finally, the required velocities need to be calculated as a function of the control requirements. Since in our case the robot is required to track a control trajectory, we incorporate the desired position in the vector of required velocities:

$$\dot{q}_r = \dot{q}_d + \lambda(q_d - q) \quad (7)$$

where  $\dot{q}_r = [\dot{q}_{1r} \ \dots \ \dot{q}_{7r}]^T$  is the required velocity vector,  $\dot{q}_d = [\dot{q}_{1d} \ \dots \ \dot{q}_{7d}]^T$  is the desired velocity vector,  $\lambda > 0$  is a control parameter,  $q_d = [q_{1d} \ \dots \ q_{7d}]^T$  is the desired position vector and  $q = [q_1 \ \dots \ q_7]^T$  is the position vector. From (5) to (7), we obtain the augmented required velocity vector as follows:

$$\mathcal{V}_7 = J_7^* \dot{q}_r \quad (8)$$

### 3.3 Dynamics

In this paper, only the inverse dynamics that take part in the control of the robot are described. To obtain the inverse dynamics, it is necessary to calculate the forces required for the links and then the torques required by the joints. The first step for the link dynamics is to obtain the required net force/moment vectors using the following equation:

$${}^{B_i}F_r^* = Y_i \hat{\theta}_i + K_{si} ({}^{B_i}V_r - {}^{B_i}V) \quad (9)$$

for  $i=1, \dots, 7$ , where  $K_{si}$  is a symmetric positive-definite gain matrix; the velocities  ${}^{B_i}V_r$  and  ${}^{B_i}V$  are defined in relations (3) and (8), respectively;  $Y_i \in \mathbb{R}^{6 \times 13}$  is a regressor matrix and  $\hat{\theta}_i \in \mathbb{R}^{13}$  is the estimated parameters vector composed of the mass of the link, the elements of the vector pointing from the origin of the reference frame of the link

toward its mass centre, as well as the elements of the inertia tensor of the body. The exact representation of each element of  $\hat{\theta}_i$  is given in Appendix B. To estimate the parameters, the following relation is used:

$$\hat{\theta}_{i\gamma} = \mathcal{P}(s_{i\gamma}(t), \rho_{i\gamma}, a_{i\gamma}(t), b_{i\gamma}(t), t) \quad (10)$$

for  $i=1, \dots, 7$  and  $\gamma=1, \dots, 13$ . Where  $\hat{\theta}_{i\gamma}$  represents the  $\gamma$ th parameter of the  $i$ th link,  $\rho_{i\gamma}$  is the parameter update gain,  $a_{i\gamma}(t)$  and  $b_{i\gamma}(t)$  are the lower and upper bounds of  $\hat{\theta}_{i\gamma}$ , respectively, and  $s(t) \in \mathbb{R}$  is a scalar variable given by:

$$s_i = Y_i^T ({}^{B_i}V_r - {}^{B_i}V) \quad (11)$$

The projection function  $\mathcal{P}$  is a differentiable scalar function defined for  $t \geq 0$ , such that its time derivative is governed by:

$$\dot{\mathcal{P}} = \rho s(t) \mathcal{K} \quad (12)$$

with:

$$\mathcal{K} = \begin{cases} 0 & \text{if } \mathcal{P} \leq a(t) \text{ and } s(t) \leq 0 \\ 0 & \text{if } \mathcal{P} \geq b(t) \text{ and } s(t) \geq 0 \\ 1 & \text{otherwise} \end{cases}$$

Once this step is completed, the required force/moment vectors at the cutting points are obtained as follows:

$${}^{B_7}F_r = {}^{B_7}F_r^*$$

$${}^{B_i}F_r = {}^{B_i}F_r^* + {}^{B_i}U_{B_{i+1}} {}^{B_{i+1}}F_r \quad (13)$$

for  $i=6, \dots, 1$ . The next step is the calculation of the dynamics of the joints. Some relations and variables used to calculate the dynamics of the joints were also used in the calculation of the dynamics of the links; therefore, a sub-index,  $a$ , is used to refer to the joints' dynamic equations and variables. For  $i=1, \dots, 7$  in (14) to (18), the following vectors are defined:

$$Y_{ai} = \begin{bmatrix} - \\ q_{ir} & \text{sign}(\dot{q}_{ir}) & \dot{q}_{ir} & 1 \end{bmatrix} \quad (14)$$

$$\theta_{ai} = [J_{mi} \ k_{ci} \ k_{vi} \ c_i]^T \quad (15)$$

where  $J_{mi}$  is the equivalent mass or moment of inertia,  $k_{ci} > 0$  denotes the Coulomb friction coefficient,  $k_{vi} > 0$  denotes the

viscous friction coefficient and  $c_i$  denotes an offset that accommodates asymmetric Coulomb frictions. As in the case of the links, the estimation of the parameters defined in (15) is performed as follows:

$$\hat{\theta}_{ai\gamma} = \mathcal{P}(s_{ai\gamma}(t), \rho_{ai\gamma}, a_{ai\gamma}(t), b_{ai\gamma}(t), t) \quad (16)$$

However, in this case,  $\gamma=1, \dots, 4$  and the scalar function  $s$  is defined for the joints as:

$$s_{ai} = Y_{ai}^T (\dot{q}_{ir} - \dot{q}_i) \quad (17)$$

Thereafter, the net torque for the joint becomes as follows:

$$\tau_{ir}^* = Y_{ai} \hat{\theta}_{ai} + k_{ai} (\dot{q}_{ir} - \dot{q}_i) \quad (18)$$

where  $k_{ai}$  denotes a feedback gain.

Finally, the torques obtained from the link and joint analyses are combined. First, we extract the torque of the link force/moment vector.

$$\tau_{air} = z^T B_i F_r \quad (19)$$

for  $i=1, \dots, 7$ . With  $z$  defined after (6), the control torque is then designed for  $i=1, \dots, 7$  as:

$$\tau_{Ci} = \tau_{ir}^* + \tau_{air} \quad (20)$$

For a complete stability analysis of VDC, the reader is referred to [19, 28, 31]. Figure 5 presents a flow diagram of the VDC architecture applied in the inner position control loop. The interaction between the robot and the human is represented by the human joint position  $q_h$ , causing the robot to exert torque  $\tau_h$ . The human also applies force,  $\mathcal{F}$ , which will be used in the outer force control loop, as described in the next section.

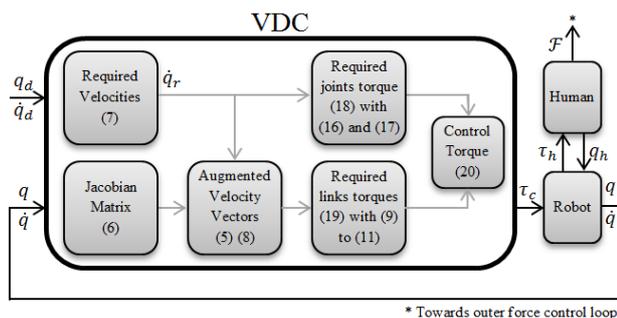


Figure 5. Block diagram of the VDC

#### 4. Active-Assistive and Active Rehabilitation

In robotics, a trajectory defines movement of a robot in a multidimensional space. This trajectory contains information concerning the position, velocity and acceleration for each degree of freedom in time [30]. To control the movement of the robot, the user defines a desired trajectory that includes this information and is usually generated by means of a trajectory planner.

##### 4.1 Trajectory Planner

The trajectory planner for ETS-MARSE is based on joint-space specification. It is performed in real time on a dedicated processor (PXI-8108), described in Section 2. Some exercises can comprise different trajectory segments, each specified by initial and final positions for each joint. Each segment of the trajectory for each joint is calculated using a cubic polynomial [30] of the form:

$$q(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (21)$$

where  $a_i$  for  $i=0, \dots, 3$  are the coefficients of the cubic polynomial to be specified. With initial and final velocities of zero, the initial position defined as  $q(0)=q_0$  and the final position defined as  $q(t_f)=q_f$ , equation (21) reduces as follows:

$$q(t) = q_0 + \frac{3(q_f - q_0)}{t_f^2} t^2 - \frac{2(q_f - q_0)}{t_f^3} t^3 \quad (22)$$

In order to obtain information on the velocity and acceleration of the trajectory, the corresponding first and second derivatives of (22) are used.

In rehabilitation, it is often useful to specify a desired trajectory in Cartesian space in order to specify movements with a physical interpretation. This can be as simple as a straight line, or, for example, following a geometrical shape such as a triangle, a rectangle, etc. In the case of the ETS-MARSE robot, the trajectory can be specified either in the joint space or in Cartesian space (robot workspace); whatever the case, the trajectory tracking is always executed in joint space. Thus, it is necessary, given a position and orientation of the end-effector in Cartesian space, to find the joint positions of the robot that can accomplish this desired configuration (inverse kinematics problem). In this case, the pseudo-inverse of the Jacobian method is used.

In the dynamic relations of the links of a robot, the reference frame of each link has linear and angular velocities, with respect to the other links and to the robot reference frame. The Jacobian matrix of a robot relates the joint velocities with linear and angular velocities of the Cartesian space as follows:

$$\dot{X} = J(q)\dot{q} \quad (23)$$

where  $\dot{X} \in \mathbb{R}^6$  is the velocity vector of the end-effector,  $\dot{q} \in \mathbb{R}^n$  is the joint velocities vector and  $J(q) \in \mathbb{R}^{6 \times n}$  is the robot Jacobian matrix. For the inverse kinematics problem presented, one possible solution can be obtained as follows [32]:

$$\dot{q} = J(q)^+ \dot{X} \quad (24)$$

where  $J(q)^+ = J(q)^T (J(q)J(q)^T)^{-1}$  is the generalized pseudo-inverse. The method can be enriched by adding null space characteristics to the right side of the equation with the term  $(J(q)^+J(q) - I)\xi$ , where  $\xi \in \mathbb{R}^n$  is an arbitrary vector that can be used for tasks such as obstacle avoidance and  $I$  is the  $6 \times 6$  identity matrix. Being a redundant robot, the ETS-MARSE has different solutions for its inverse kinematics problem; currently, to solve that problem, we are developing an inverse kinematics solution (analytical, geometrical or numerical) that will ensure anthropomorphic configurations, as in [17, 33]. In the rehabilitation tasks presented here, solving the null space is not necessary, since only predefined Cartesian trajectories were used. These trajectories were simulated and tested to constantly ensure a configuration of the robot not presenting any non-ergonomic positions. Additionally, singularity avoidance is secured with joint angle limits, both in software and hardware (mechanical stoppers).

#### 4.2 Active-assistive Rehabilitation Based on an Admittance-modified Trajectory Planner

In the case of active-assistive mode, the robot helps the user, at an adjustable level, to achieve the tracking of a trajectory shown in the virtual interface. To do this, the assistive controller somewhat modifies the predefined trajectory in relation to the patient's movement limitations. The system measures the force exerted by the user at the end-effector of the robot, which can be interpreted as the difference between the user's trajectory and the goal trajectory.

The first step is to transform this information into a quantity that specifies how the subject's opposition force affects all the DOF of the exoskeleton (that are related to the human arm joints). As a consequence of the virtual work principle, the Jacobian matrix relates static forces, i.e., it transpose assigns Cartesian forces to joint torques as follows:

$$\tau = J^T \mathcal{F} \quad (25)$$

where  $\tau$  is the  $7 \times 1$  vector of joint torque and  $\mathcal{F}$  is the  $6 \times 1$  measured force-moment Cartesian vector. This relation gives us the user's opposition force in terms of joint torques.

The second step is to transform these torques into meaningful information for the trajectory that is being followed, that is, information relating to changes in position. Impedance can be defined as a transfer function between the external force acting on the manipulator and its displacement [34]. Knowing the result from (25) and applying the impedance definition in the form of an admittance relation, which directly relates an input force to an output position, we have a direct form with which to modify the desired trajectory as follows:

$$q_a = q_d + \tau \left( \frac{1}{K + Cs} \right) \quad (26)$$

where  $q_a$  is the  $7 \times 1$  vector of the new desired trajectory defined by the admittance,  $q_d$  is the  $7 \times 1$  vector with the original desired trajectory from the trajectory planner and the final right term of the equation in the parenthesis, the chosen admittance function. It can be seen that the latter has the form of a spring-damper function, in which  $\tau$  is the  $7 \times 1$  vector of torques exerted on the robot's joints by the subject, calculated from (25) and  $K$  and  $C$  are the  $7 \times 7$  gain matrices corresponding to the spring and damper constants, respectively. By adjusting these constants to provide higher or lower opposition to the subjects' disturbances and constraints over movement, this admittance function allows for variation in the amount of help to be provided by the robot. This particular admittance function by means of the damping term also provides smooth movement.

#### 4.3 Active Rehabilitation Based on an Admittance-modified Trajectory Planner

Another application of an admittance function modifying a predefined trajectory involves performing a free trajectory movement. If the spring-damper system introduced by the admittance function in (26) is detached from a desired trajectory, it becomes a means of moving the robot freely within its workspace. In this case, the subject is allowed to move in the environment presented by the virtual interface and can try to follow the trajectory without any robotic aid. This rehabilitation option, corresponding to active (unassisted) rehabilitation, is achieved by changing the desired trajectory of the robot, that is,  $q_d$  in (26). Instead of coming from a trajectory planner, the desired trajectory comes from the robot's previous position, as follows:

$$q_a = q + \tau \left( \frac{1}{K + Cs} \right) \quad (27)$$

where  $q$  is the  $7 \times 1$  vector of the joint positions. If the user stops exerting forces on the force sensor,  $\tau \rightarrow 0$ , causing the robot to reduce its movement and gradually, when  $q_a = q$ , the robot remains in its most recent position. With this

mode, the robot is allowed to follow the user's movements, while the trajectory tracking ensures the weightless sensation of the robot structure to the subject and provides gravity compensation. In Figure 6, a block diagram showing the control architecture is presented. The position of the selector shown in the bottom row of the system corresponds to the case of active-assistive rehabilitation (left position) and active rehabilitation (right position) described earlier. In this case, the desired trajectory that inputs the VDC (see Figure 5) is the admittance modified trajectory  $q_a$  from (26) or (27) and  $\Delta q$  is the second term of the right-hand side of the same relations.

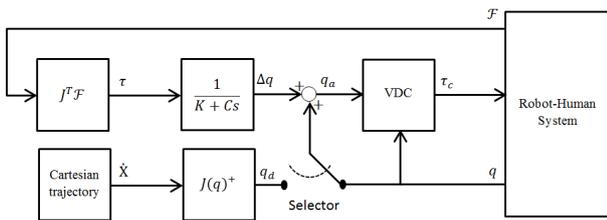


Figure 6. Block diagram of the system

The ETS-MARSE works in two different active rehabilitation modes, which use virtual walls added to the environment in order to constrain movement to a confined space. In the first mode, the system restrains movement beyond the wall and only allows movement that corrects the collision situation, i.e., the robot will not allow the user "to slide" through the wall to complete the exercise. The user must return to the inner part of the space in order to continue with the movement. The algorithm used in this rehabilitation mode is presented in Figure 7. The second mode of operation is intended for users that need more assistance once there is a deviation from the main trajectory. Once the user hits a wall, the robot stops for a predetermined time to let the user notice that movement has been stopped by the system. It then goes into passive mode, which means that the robot will help the user by returning automatically to the closest point on the desired trajectory, as illustrated in Figure 8. Here,  $P_w$  represents the point where the user hits the wall,  $P_i$  the trajectory initial point,  $P_f$  the final point and  $P_r$  the point of the position reset. The last one, the closest point to the trajectory, is obtained as follows: first, the cosine of angle  $\alpha$  is calculated:

$$\cos \alpha = \frac{|u_1 \cdot v_1 + u_2 \cdot v_2 + u_3 \cdot v_3|}{\sqrt{u_1^2 + u_2^2 + u_3^2} \cdot \sqrt{v_1^2 + v_2^2 + v_3^2}} \quad (28)$$

where  $u \in \mathbb{R}^3$  and  $v \in \mathbb{R}^3$  are the direction vectors of the lines  $(\overline{P_i P_w})$  and  $(\overline{P_w P_r})$ , respectively, and  $u_i$  and  $v_i$  are the  $i_{th}$  elements of the corresponding direction vector. Then, the reset point is calculated as:

$$P_r = \cos \alpha \cdot \sqrt{u_1^2 + u_2^2 + u_3^2} + P_i \quad (29)$$

With the final point obtained from (29) and the initial point as  $P_w$ , the passive mode of the system conducts the robot end-effector to this point by means of a trajectory generated as described in subsection 4.1. Finally, after another predetermined time to allow the user to be ready, the system allows them to continue. The algorithm followed by this rehabilitation mode is presented in Figure 9.

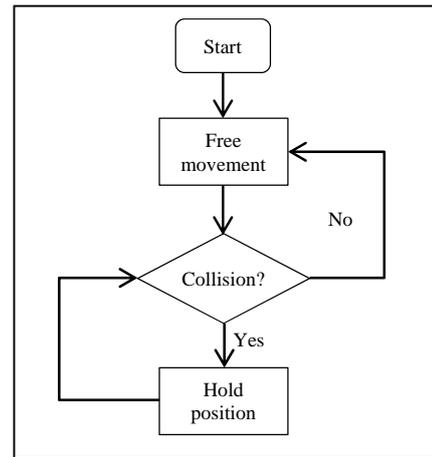


Figure 7. Active rehabilitation with walls

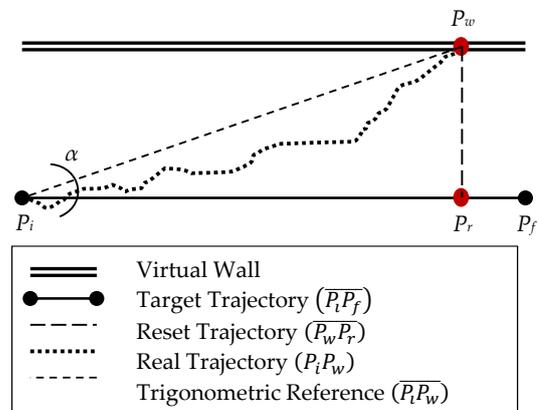


Figure 8. Virtual wall collision and reset path

## 5. Experimental Results

In this section, we present one scenario for each of the two rehabilitation modes. The first comprises an experiment performed for active-assistive rehabilitation and the second, two tests for active rehabilitation.

### 5.1 Active-assistive Rehabilitation

This first scenario comprises three tests based on a tridimensional square trajectory, defined below. The trajectory involves the movement of the seven degrees of freedom of

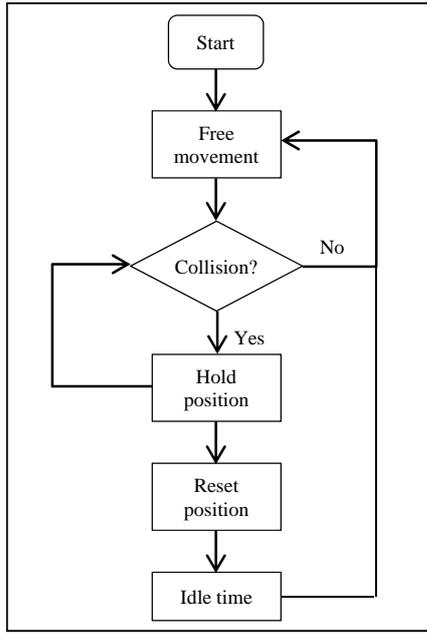


Figure 9. Active-assisted rehabilitation with walls

the robot. Test number one is given as a reference of the behaviour of the robot, i.e., without modifying the trajectory tracking. This scenario corresponds to the passive rehabilitation mode in which the patient is guided through a predefined rehabilitation exercise. The result of this test is shown in Figure 10. It can be seen that the tracking error is very small and this will be used as a reference value. The initial position of the end-effector of the robot corresponds to point A; it moves to points B – C – D and then returns to A. The points are listed in Table 2.

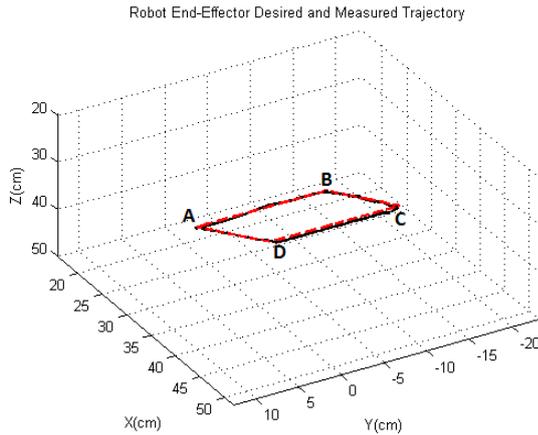


Figure 10. Reference trajectory tracking

The next two tests constitute the central point of this approach of rehabilitation (active-assistive). In these cases, the tracking is modified based on the measurements of the force sensor. The reference trajectory is the same as that presented in Figure 10, but modified according to equation (26). To simulate stiffness or spasticity from the user, an

Point	X(cm)	Y(cm)	Z(cm)
A	28.67	2.51	37.08
B	28.67	-12.51	37.08
C	43.67	-12.51	27.08
D	43.67	2.51	27.08

Table 2. Robot active-assistive rehabilitation reference trajectory

elastic band is tied from the handle of the robot (which holds the force sensor) to an external anchor point. This provides a systematic comparison, depending only on the modification of the admittance parameters. The setup allows the variation to be independent from the test subject, whose behaviour can change from one test to another. For the first test, the admittance parameters were selected to provide a high degree of help, specifically with the main diagonal of  $K=[0.05 \ 0.06 \ 0.06 \ 0.05 \ 0.02 \ 0.02 \ 0.02]^T$  and the remaining elements equal to 0, and the main diagonal of  $C=[5 \ 1 \ 5 \ 10 \ 5 \ 5 \ 5]^T$ . The remaining elements were also zero. The second test was configured to provide a low degree of help from the robot, i.e., with the main diagonal of  $K=[0.1 \ 0.12 \ 0.1 \ 0.04 \ 0.04 \ 0.04]^T$  and the main diagonal of  $C=[5 \ 1 \ 5 \ 10 \ 5 \ 5 \ 5]^T$ , and the remaining elements being zero.

The results of these tests are shown in Figures 11 and 12 for high and low degrees of help, respectively. Because the opposition force to this movement is exerted on the robot, the actual trajectory is deflected from the predefined one. In these figures, the dotted line represents the desired trajectory and the solid line the actual trajectory followed by the robot, which was perturbed using the elastic band.

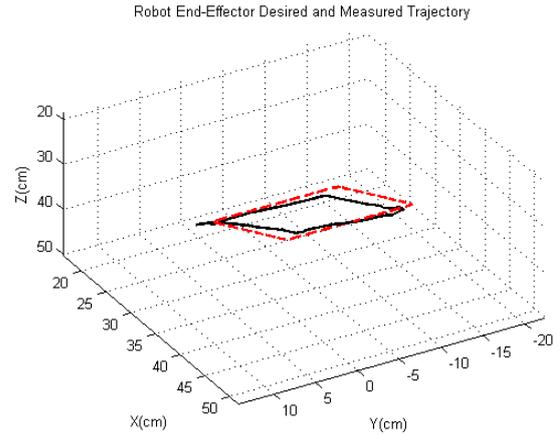


Figure 11. Trajectory modified with a high degree of help

For the final two tests (Figures 11 and 12), the force and torque measured with the force sensor are shown in Figure 13. We can see how the force input from the elastic band is very similar in both tests; nevertheless, the modification of the trajectory is different, allowing us to see the effects of modifying the admittance function coefficients.

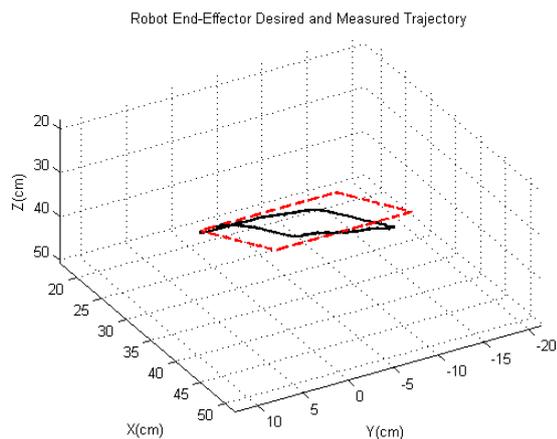


Figure 12. Trajectory modified with low degree of help

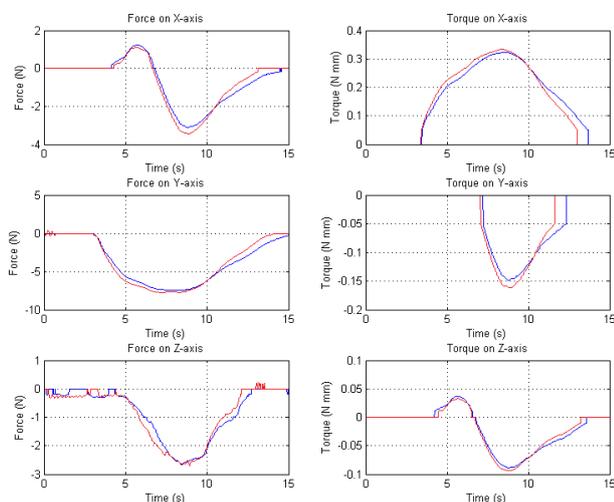


Figure 13. Comparison of forces-torques between high and low degree of assistance from the robot

### 5.2 Active Rehabilitation, Constrained Free Movement

In this rehabilitation mode, the user follows the trajectory by means of a virtual interface that provides the tracking of movement and incorporates a variety of exercises in a game-based environment. The active rehabilitation mode with virtual walls has two different tests: the first with the wall as an obstacle and the second, where collision with the wall indicates that the robot will assist the user, as described in section 4. Both setups require the subject (32-year old healthy male subject with a mass of 91 kg and height of 182 cm, in seated position) to track a straight line; the virtual wall is located parallel to the desired trajectory, 10 cm to the right.

The result for the first experiment is shown in Figure 14. In this mode, once the user hits the wall, the system only allows movement towards the desired trajectory. In the plots at the bottom of the figure, the filtered force in the Y-axis of the force sensor and the position of the end-effector in the Y-axis of the reference frame are shown as a function

of time. Both axes are the Y-axis, because the wall is set in the Y-direction. The Y-axis of the force sensor is parallel to the Y-axis of the reference frame, but in the opposite direction, which provides a good reference point for the comparison. Nevertheless, it is very important to remember that the force and its modification follow the transformation described in (25) and (27). It can be seen that, when the subject starts to apply force to the sensor at 12.86 seconds, the displacement of the robot starts. At 17.05 seconds, the wall is hit; the virtual wall restricts the movement of the end-effector beyond this point (at approximately 0.08 m), irrespective of an increase in the user's produced force. Finally, at 20.40 seconds, when the user starts to apply force in the opposite direction (away from the wall), the system allows the robot to continue with the movement and completes the exercise.

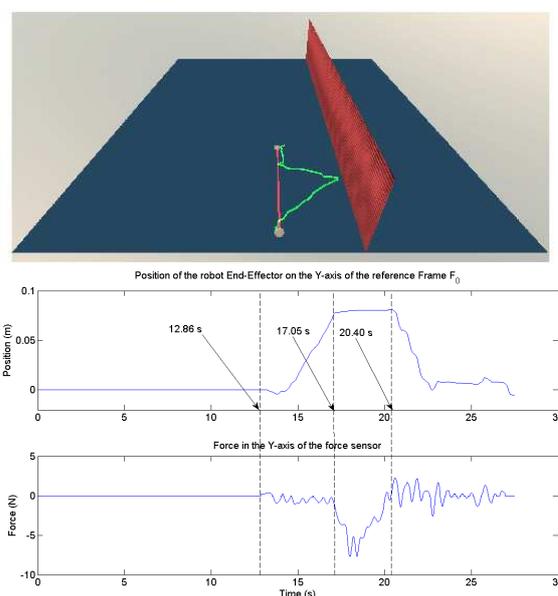
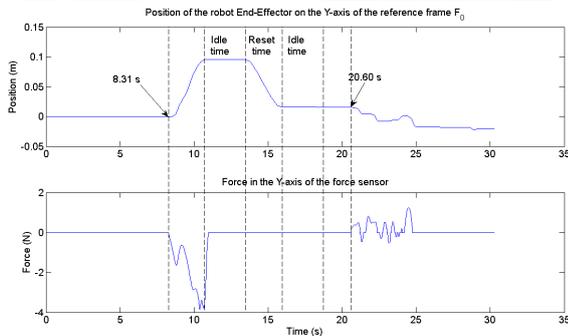
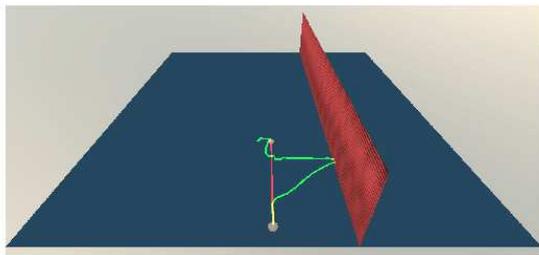


Figure 14. Virtual walls as obstacles for active rehabilitation

The final test shows the assisted wall return. It can be seen in Figure 15 that the user starts the movement by applying force to the robot at 8.31 seconds. Once the wall is hit, at approximately 10.90 seconds, it can be seen that the movement of the robot stops, even though force continues to be exerted for some time. The system, after an idle waiting time (three seconds), resets the position to the closest point in the trajectory (two seconds), followed by another idle time (three seconds), allowing the user to be ready. After this, the user can continue the exercise when ready, in this case, at 20.06 seconds. It is important to note that in the two idle states and in the reset state, there is no input from the user to the force sensor. This shows that assistance is provided for the system without any action needed from the user. The small delays that can be observed in the state changes, especially from the applied force to the start of movement, are due to the time it takes to overcome the static friction and the filtering in the force sensor's raw data; this delay is imperceptible to the user.



**Figure 15.** Virtual walls as obstacle and starting point for robotic assistance in active rehabilitation

## 6. Conclusions

The experimental results confirm that the present work can actively help in different rehabilitation scenarios. For active-assistive rehabilitation, with the guidance of the ETS-MARSE, a subject can accomplish a specified trajectory. The robot can vary the assistance provided according to the evolution and the needs of future patients. It is important to note that the robot will not produce forces that are beyond the capabilities of the subject in order to fulfil a trajectory. A qualified therapist must adjust the limits according to the user's requirements. The amount of variation of the trajectory can be quantified as an excellent measurement of progression of the subject. In the case of active rehabilitation, it is shown how by means of the same admittance function, the system can work in a free movement mode. This rehabilitation mode, combined with limits (virtual walls), could indicate when patients will need assistance from the robot to complete their rehabilitation tasks. As the next step for active rehabilitation, the system will incorporate the use of electromyographic signals in order to complete the readings of the force sensor and to provide better tracking of the subject's movement intentions, and will be used on its own for subjects where the force sensor may be not enough to determine the intended movement.

## 7. Acknowledgements

Cristóbal Ochoa Luna would like to thank the National Council on Science and Technology of Mexico (CONACYT) for the support under grant No. 211951.

This article is a revised and expanded version of a paper titled "Robotic assisted trajectory tracking for human arm

rehabilitation," presented at the 2014 International Conference on Mechatronics and Robotics, Structural Analysis (MEROSTA 2014) in the session: Advances in Robotics, Mechatronics and Circuits. Santorini, Greece, 2014.

## 8. References

- [1] National Stroke Association®. Effects of Stroke - Paralysis [Internet]. 2012. Available from: <http://www.stroke.org/site/PageServer?pagename=paralysis>. Accessed on 06 Oct 2014.
- [2] Feriančík M, Líška O. Automatic rehabilitation device based on robot and machine vision. In: Proceedings of the 2014 International Conference on Mechatronics and Robotics, Structural Analysis (MEROSTA 2014); 17-21 July 2014; Santorini Island, Greece. p. 201-203.
- [3] Kurnicki A, Stanczyk B, Kania B. Manipulator development for telediagnosics. In: Proceedings of the 2014 International Conference on Mechatronics and Robotics, Structural Analysis (MEROSTA 2014); 17-21 July 2014; Santorini Island, Greece. p. 214-218.
- [4] Masiero S, Armani M, Rosati G. Upper-limb robot-assisted therapy in rehabilitation of acute stroke patients: focused review and results of new randomized controlled trial. *Journal of Rehabilitation Research & Development*. 2011; 48:355-366. DOI: 10.1682/JRRD.2010.04.0063.
- [5] Hesse S, Schmidt H, Werner C. Machines to support motor rehabilitation after stroke: 10 years of experience in Berlin. *Journal of Rehabilitation Research & Development*. 2006; 43:671-678. DOI: 10.1682/JRRD.2005.02.0052.
- [6] Riener R, Nef T, Colombo G. Robot-aided neurorehabilitation of the upper extremities. *Medical & Biological Engineering & Computing*. 2005; 43:2-10. DOI: 10.1007/BF02345116.
- [7] Kwakkel G, Kollen B J, Krebs H I. Effects of Robot-Assisted Therapy on Upper Limb Recovery after Stroke: A Systematic Review. *Neurorehabilitation and Neural Repair*. 2008; 22:111-121. DOI: 10.1177/1545968307305457.
- [8] Norouzi-Gheidari N, Archambault P S, Fung J. Effects of robot-assisted therapy on stroke rehabilitation in upper limbs: systematic review and meta-analysis of the literature. *Journal of Rehabilitation Research & Development*. 2012; 49:479-496. DOI: 10.1682/JRRD.2010.10.0210.
- [9] Klamroth-Marganska V, Blanco J, Campen K et al. Three-dimensional, task-specific robot therapy of the arm after stroke: a multicentre, parallel-group randomised trial. *The Lancet Neurology*. 2014; 13:159-166. DOI: 10.1016/S1474-4422(13)70305-3.
- [10] Rahman M H, Kittel-Ouimet T, Saad M et al. Dynamic Modeling and Evaluation of a Robotic

- Exoskeleton for Upper-limb Rehabilitation. *International Journal of Information Acquisition*. 2011; 8:83-102. DOI: 10.1142/S0219878911002367.
- [11] Rahman M H, Saad M, Kenné J P et al. Development of a 4DoFs exoskeleton robot for passive arm movement assistance. *International Journal of Mechatronics and Automation*. 2012; 2:34-50. DOI: 10.1504/IJMA.2012.046587.
- [12] Rahman M H, Saad M, Kenné J P et al. Nonlinear Sliding Mode Control Implementation of an Upper Limb Exoskeleton Robot to Provide Passive Rehabilitation Therapy. In: C.-Y. Su, S. Rakheja and H. Liu, editors. *Intelligent Robotics and Applications*. Springer Berlin Heidelberg; 2012. p. 52-62. DOI: 10.1007/978-3-642-33515-0\_6.
- [13] Ochoa Luna C, Rahman M H, Saad M et al. Robotic assisted trajectory tracking for human arm rehabilitation. In: *Proceedings of the 2014 International Conference on Mechatronics and Robotics, Structural Analysis (MEROSTA 2014)*; 17-21 July 2014; Santorini Island, Greece. 2014. p. 51-55.
- [14] Brokaw E B, Murray T, Nef T et al. Retraining of interjoint arm coordination after stroke using robot-assisted time-ndependent functional training. *Journal of Rehabilitation Research & Development*. 2011; 48:299-316. DOI: 10.1682/JRRD.2010.04.0064.
- [15] Krebs H I, Palazzolo J J, Dipietro L et al. Rehabilitation Robotics: Performance-Based Progressive Robot-Assisted Therapy. *Autonomous Robots*. 2003; 15:7-20. DOI: 10.1023/A:1024494031121.
- [16] Dipietro L, Ferraro M, Palazzolo J J et al. Customized interactive robotic treatment for stroke: EMG-triggered therapy. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 2005; 13:325-334. DOI: 10.1109/TNSRE.2005.850423.
- [17] Culmer P R, Jackson A E, Makower S et al. A Control Strategy for Upper Limb Robotic Rehabilitation With a Dual Robot System. *IEEE/ASME Transactions on Mechatronics*. 2010; 15:575-585. DOI: 10.1109/TMECH.2009.2030796.
- [18] Seraji H. Adaptive admittance control: an approach to explicit force control in compliant motion. In: *Proceedings 1994 IEEE International Conference on Robotics and Automation*; 8-13 May 1994; San Diego, CA. IEEE; 1994. p. 2705-2712.
- [19] Zhu W-H, Yu-Geng X, Zhang Z-J et al. Virtual decomposition based control for generalized high dimensional robotic systems with complicated structure. *IEEE Transactions on Robotics and Automation*. 1997; 13:411-436. DOI: 10.1109/70.585903.
- [20] Zhu W-H, Piedboeuf J-C, Gonthier Y. A dynamics formulation of general constrained robots. *Multi-body System Dynamics*. 2006; 16:37-54. DOI: 10.1007/s11044-006-9011-9.
- [21] Zhu W-H, De Schutter J. Experimental verifications of virtual-decomposition-based motion/force control. *IEEE Transactions on Robotics and Automation*. 2002; 18:379-386. DOI: 10.1109/TRA.2002.1019474.
- [22] Ochoa Luna C, Rahman M H, Saad M et al. Virtual decomposition control of an exoskeleton robot arm. *Robotica*. 2014; CJO:1-23. DOI: 10.1017/S026357471400246X.
- [23] Calanca A, Fiorini P. Human-Adaptive Control of Series Elastic Actuators. *Robotica*. 2014; CJO:1-16. DOI: 10.1017/S0263574714001519.
- [24] Ozkul F, Barkana D E. Upper-Extremity Rehabilitation Robot RehabRoby: Methodology, Design, Usability and Validation. *International Journal of Advanced Robotic Systems*. 2013; 10:1-13. DOI: 10.5772/57261.
- [25] Younggeun C, Gordon J, Duckho K et al. An Adaptive Automated Robotic Task-Practice System for Rehabilitation of Arm Functions After Stroke. *IEEE Transactions on Robotics*. 2009; 25:556-568. DOI: 10.1109/TRO.2009.2019787.
- [26] Colombo R, Pisano F, Micera S et al. Robotic techniques for upper limb evaluation and rehabilitation of stroke patients. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 2005; 13:311 - 324. DOI: 10.1109/TNSRE.2005.848352.
- [27] Zhang M, Davies T C, Xie S. Effectiveness of robot-assisted therapy on ankle rehabilitation - a systematic review. *Journal of NeuroEngineering and Rehabilitation*. 2013; 10:30. DOI: 10.1186/1743-0003-10-30.
- [28] Zhu W-H, Lamarche T, Dupuis E et al. Precision Control of Modular Robot Manipulators: The VDC Approach With Embedded FPGA. *IEEE Transactions on Robotics*. 2013; 29:1162-1179. DOI: 10.1109/TRO.2013.2265631.
- [29] Zhu W-H, De Schutter J. Adaptive control of mixed rigid/flexible joint robot manipulators based on virtual decomposition. *IEEE Transactions on Robotics and Automation*. 1999; 15:310-317. DOI: 10.1109/70.760352.
- [30] Craig J J. *Introduction to robotics: mechanics and control*. 3rd ed. Upper Saddle River, N.J.: Pearson/Prentice Hall; 2005. 408 p.
- [31] Zhu W-H, Zeungnam B, De Schutter J. Adaptive motion/force control of multiple manipulators with joint flexibility based on virtual decomposition. *IEEE Transactions on Automatic Control*. 1998; 43:46-60. DOI: 10.1109/9.654886.
- [32] Asada H, Slotine J-J E. *Robot analysis and control*. New York: J. Wiley; 1986. 277 p.
- [33] Zhang B, Xiong R, Wu J. Kinematics analysis of a novel 7-DOF humanoid manipulator for table

tennis. In: 2011 International Conference on Electronics, Communications and Control (ICECC); 09-11 September 2011; Ningbo, China. IEEE; 2011. p. 1524-1528.

[34] Gorinevsky D M, Formalsky A M, Schneider A Y. Force control of robotics systems. Boca Raton, FL: CRC Press; 1997. 363 p.

## Appendix A

The motors and harmonic drives specifications are listed in Table 3:

Actuators, Maxon brushless DC motors			
Specification	EC-90, flat 90 W (joints 1, 2, 4)	EC-45, flat 30 W (joints 3, 5-7)	
Nominal voltage (V)	24	12	
Nominal speed (rpm)	2650	2860	
Torque constant (mNm/A)	70.5	25.5	
Weight (g)	648	88	
Harmonic drives			
Specification	CSF-2UH17-120 F (joints 1, 2)	CSF-2XH14-100 F (joint 4)	CSF-2XH11-100 F (joints 3, 5-7)
Torque at 2000 rpm (Nm)	24	7.8	5
Momentary peak torque (Nm)	86	54	25
Repeated peak torque (Nm)	54	28	11
Gear ratio	120	100	100

**Table 3.** ETS-MARSE detailed specification for DC motors and harmonic drives

## Appendix B

The 13 elements of parameter vectors  $\theta_A \in \mathbb{R}^{13}$  are listed as:

$$\theta_{A1} = m_A \quad (\text{A.1})$$

$$\theta_{A2} = m_A \, {}^A r_{mx} \quad (\text{A.2})$$

$$\theta_{A3} = m_A \, {}^A r_{my} \quad (\text{A.3})$$

$$\theta_{A4} = m_A \, {}^A r_{mz} \quad (\text{A.4})$$

$$\theta_{A5} = m_A \, {}^A r_{mx}^2 \quad (\text{A.5})$$

$$\theta_{A6} = m_A \, {}^A r_{my}^2 \quad (\text{A.6})$$

$$\theta_{A7} = m_A \, {}^A r_{mz}^2 \quad (\text{A.7})$$

$$\theta_{A8} = m_A \, {}^A r_{mx} \, {}^A r_{my} - I_{Axy} \quad (\text{A.8})$$

$$\theta_{A9} = m_A \, {}^A r_{mx} \, {}^A r_{mz} - I_{Axz} \quad (\text{A.9})$$

$$\theta_{A10} = m_A \, {}^A r_{my} \, {}^A r_{mz} - I_{Ayz} \quad (\text{A.10})$$

$$\theta_{A11} = I_{Axx} \quad (\text{A.11})$$

$$\theta_{A12} = I_{Ayy} \quad (\text{A.12})$$

$$\theta_{A13} = I_{Azz} \quad (\text{A.13})$$

where  $m_A$  is the mass;  ${}^A r_m = [{}^A r_{mx} \, {}^A r_{my} \, {}^A r_{mz}]^T \in \mathbb{R}^3$  denotes a vector pointing from the origin of frame A towards the mass centre, expressed in frame A; and  $I_{Axx}$ ,  $I_{Ayy}$ ,  $I_{Azz}$ ,  $I_{Axy}$ ,  $I_{Axz}$ ,  $I_{Ayz}$  are the elements of  $I_A$ .