Robotica (2016) volume 34, pp. 1587–1609. © Cambridge University Press 2014
doi:10.1017/S026357471400246X

Virtual decomposition control of an exoskeleton robot arm

Cristóbal Ochoa Luna†* Mohammad Habibur Rahman†‡, Maarouf Saad†, Philippe Archambault‡§ and Wen-Hong Zhu¶

† *Department of Electrical Engineering, École de technologie supérieure, Montreal, Canada.*
Emails: mohammad-habibur.rahman.1@ens.etsmtl.ca, maarouf.saad@etsmtl.ca
‡*School of Physical & Occupational Therapy, McGill University, Montreal, Quebec, Canada.*
Email: philippe.archambault@mcgill.ca
§*Center for Interdisciplinary Research in Rehabilitation (CRIR), Montreal, Quebec, Canada*
¶*Space Exploration, Canadian Space Agency, Longueuil, Quebec, Canada.*
Email: Wen-Hong.Zhu@asc-csa.gc.ca

(Accepted September 16, 2014. First published online: October 15, 2014)

SUMMARY
Exoskeleton robots, which can be worn on human limbs to improve or to rehabilitate their function, are currently of great importance. When these robots are used in rehabilitation, one aspect that must be fulfilled is their capacity to adapt to different patients without significantly varying their performance. This paper describes the application of a relatively new control technique called virtual decomposition control (VDC) to a seven degrees-of-freedom (DOF) exoskeleton robot arm, named ETS-MARSE. The VDC approach mainly involves decomposing complex systems into subsystems, and using the resulting simpler dynamics to conduct control computation, while strictly ensuring global stability and having the subsystem dynamics interactions rigorously managed and maintained by means of virtual power flow. This approach is used to deal with different masses, joint stiffness and biomechanical variations of diverse subjects, allowing the control technique to naturally adapt to the variances involved and to maintain a successful control task. The results obtained in real time on a 7DOF exoskeleton robot arm show the effectiveness of the approach.

KEYWORDS: Exoskeleton robot; Virtual decomposition control; Passive rehabilitation; Active rehabilitation; Upper-limb impairment; Dynamic variations.

1. Introduction

Orthotic robots, or exoskeletons, mimic and adapt to the structure of an injured limb to improve, re-establish or complete some of its functionality.[1] Following occupational and sport injuries, traumas, geriatric disorders[2]and strokes,[3]etc., diverse types of physiological complications can arise, possibly leading to social, labour and personal limitations, and consequently decreasing the quality of life of a large number of individuals. One of the symptoms most often diagnosed as a result of a stroke is the loss of function on one side of the body. The main treatment for this is rehabilitation, which requires hours of highly skilled and extensive procedures. The number of people qualified to provide such support is hardly enough to treat all patients affected. Rehabilitation can be supplemented with the use of robots, especially wearable (orthotic or prosthetic) robots.[4–6] These robots can help people by reducing the impacts of their disabilities and by significantly improving their quality of life.

Passive arm movement therapy is a form of treatment used among patients who are unable to actively move their arms throughout their complete range of motion, following a neurological insult,

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

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

such as a stroke, which can lead to weakness or paralysis. In passive rehabilitation, the affected limb is moved by external forces (therapist or robot) while the patient relaxes. Active rehabilitation for its part focuses on long-term results; it encourages subjects to move their arms by themselves in order to build muscle strength and flexibility. These issues must therefore be properly addressed in rehabilitation robotics. In the area of upper limb rehabilitation, which was initialized in the middle of the twentieth century,[7] it was not until very recently that truly functional experimental implementations were obtained with the full capabilities of the whole arm (shoulder, elbow and wrist).[8,9] Different studies have shown that the use of robotic tools in rehabilitation provides intensive therapy that leads to measurable functional gains.[10–13] To that end, work in our laboratory has led to the development of a seven DOF exoskeleton robot to ultimately assist patients with shoulder (3DOF), elbow (1DOF), and wrist (3DOF) movements.[14] The primary objective of this research is to develop a robotic device that can offer passive and active rehabilitation, as well as daily assistance of the upper limbs, to physically disabled people.

A wide range of approaches have been proposed to control rehabilitation robots, ranging from simple PD controllers to nonlinear techniques such as computed torque control (CTC) and neuro-fuzzy controllers. Some attractive combined approaches, such as an adaptive-fuzzy controller based on sliding mode control (SMC), for example, was presented by Sun *et al*,[15] and seems to be able to deal with the inherent chattering of the SMC, subject to model uncertainties and external disturbances. A combination of adaptive and robust control[16] merges the advantages of both control laws, eliminating the disadvantage of adaptive controllers in transient behaviour and suppressing the steady-state error. Furthermore, a CTC with fuzzy control[17] provides satisfactory results, subject to manipulator-structured and unstructured uncertainties. An adaptive control with an artificial neural network was presented by Ciliz[18] for dealing with frictional uncertainties, which are highly difficult to model. Some other approaches use robust adaptive control that indicates an interesting way of dealing with system uncertainties and external disturbances.[19,20] However, some fundamental problems are yet to be solved.

In CTC, the main concern is that the control only shows moderate performance improvement when the dynamical model is not exactly known, and this is true even for robots with relatively small numbers of DOF, such as 4DOF orthosis[21] for example. Neuro-fuzzy controllers are highly complex, and use different sets of rules and layers involving switching in different regions of operation. This is true even for 3DOF arm exoskeletons.[22] In most of the above-mentioned approaches,[15–20] the results are mainly restricted to simulations (i.e., a 2DOF manipulator,[15] a two-link planar robot[16] and a two-link elbow planar manipulator)[17] or implementation on simple robots (i.e., successfully tested on a two-link SCARA type direct drive arm).[18]

As can be seen, two specific problems need to be considered together in exoskeleton research: the high number of DOF and the system variability (biomechanical and physiological). For these reasons, a novel application of the VDC technique is introduced in this paper for the control of exoskeleton robots, in order to address two important issues: (a) to facilitate the implementation of a high DOF robot (7DOF in this case), and (b) to cope with the changes in the dynamics of the robot-human system, due to the biomechanical and physiological characteristics of different patients. As robots become more complex, the complexity of the control task increases as well. Indeed, the complexity of the computation of robot dynamics is proportional to the fourth power of the number of DOF in Lagrangian dynamics.[23] This research deals with a complex 7DOF robot. The VDC approach allows us to break the system down into subsystems with relatively simpler dynamics and to apply the control directly over those subsystems. The VDC uses the dynamics of subsystems (rigid links and joints) to conduct control design, while rigorously maintaining the L_2 and L_∞ stability of the entire robotic system.[24] This virtual decomposition yields simpler dynamic equations, both in their formulation and implementation. Proceeding as such maintains the complexity of the dynamics of the subsystems constant, meaning that as more and more DOF are added to the robot, the complexity of the total system increases linearly. With the system broken down into simpler subsystems, the VDC handles the dynamic interactions among subsystems mathematically through the definition of virtual power flows. Thus, even with the subsystem decomposition, the highly nonlinear dynamics of the robotic system are not decoupled, thereby maintaining the control and stability of the entire system.

An additional advantage of the VDC is its ability to manage, through its parameter adaptation, the inherent variability that is present in human dynamics. In addition to the imperfect knowledge of the dynamics of the body structures, different individuals will display differences in body mass,

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

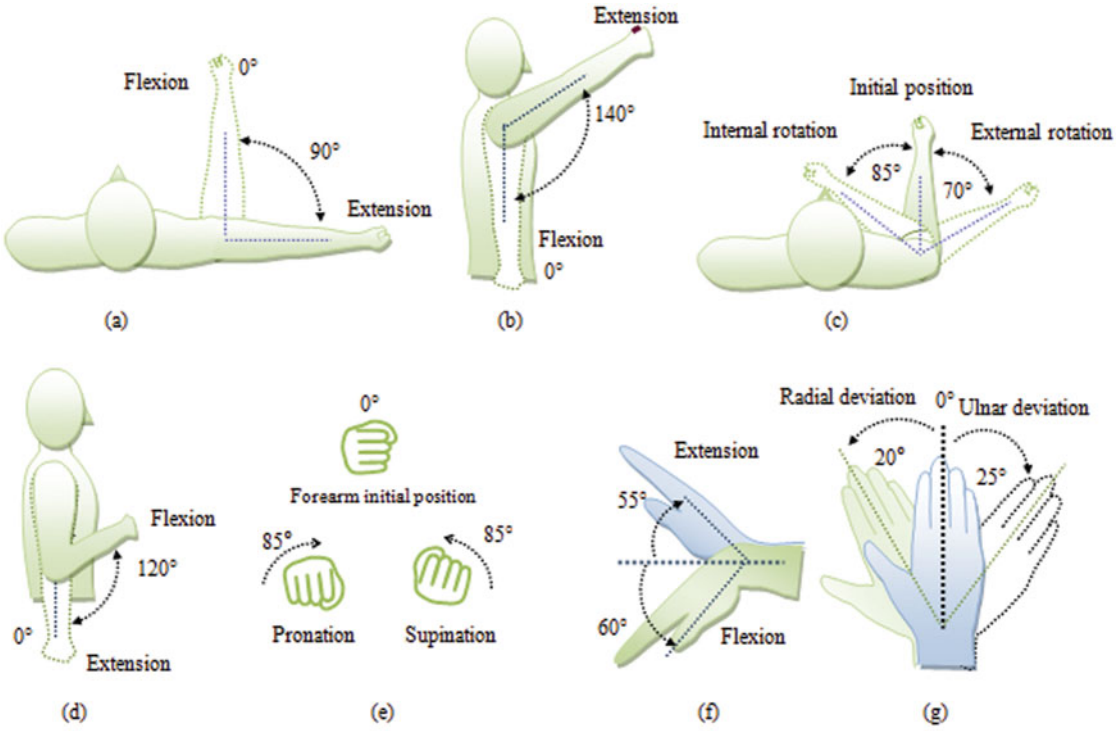


Fig. 1. Joint range of motion of ETS-MARSE. (a) shoulder joint: horizontal flexion/extension; (b) shoulder joint: vertical flexion/extension; (c) shoulder joint: internal/external rotation; (d) elbow flexion/extension; (e) forearm pronation/supination; (f) wrist flexion/extension; (g) radial/ulnar deviation.

weight distribution, limb length, etc. Moreover, muscular and neurological conditions can change the visco-elastic properties of muscles and joints. For controllers that need to know the dynamical model of the system (e.g., CTC), variations in all these parameters could have a negative impact on the performance of the robot. In our previous research,[25] SMC was implemented for dynamic trajectory tracking corresponding to typical passive rehabilitation exercises. Though SMC performed well, it also showed some limitations in dealing with different subjects. The adaptive characteristics of the VDC overcome this limitation by dealing with biomechanical and physiological variations, providing a better and constant performance of the exercises.

The original contribution of this paper is the application of the VDC to address the issue of dynamic variability, due to individual differences, in order to obtain constant performance by a rehabilitation exoskeleton arm. As well, the simplification of the control task of these multi-articulated robots, through the virtual decomposition process and the use of virtual stability and virtual power flows, allows a simpler computation for systems with high number of DOF. The experiments were carried out on different healthy subjects with the ETS-MARSE exoskeleton robot, designed by our research team.

This paper is organized as follows: Section 2 presents a brief description of the robot and its characteristics. Section 3 lays out a detailed presentation of the control development. Finally, in Section 4, the implementation of the control and experiments executed are explained and the results are presented and discussed, followed in Section 5 by the conclusions of the work.

2. Characteristics of the Motion Assistive Robotic Exoskeleton for Superior Extremity

The wearable upper-limb exoskeleton robot, named "motion assistive robotic exoskeleton for superior extremity" (ETS-MARSE) has 7DOF, and was designed to correspond to the natural range of movement of the upper extremity of the human body. It is composed of a shoulder motion support part (which assists in horizontal flexion/extension (Fig. 1a), vertical flexion/extension (Fig. 1b), and internal/external rotation (Fig. 1c) of the shoulder), an elbow motion support part (Fig. 1d) for elbow flexion/extension, a forearm motion support part which is responsible for forearm

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

Table I. ETS-MARSE modified Denavit–Hartenberg parameters.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0°	0	d_1	θ_1
2	−90°	0	0	θ_2
3	90°	0	d_2	θ_3
4	−90°	0	0	θ_4
5	90°	0	d_5	θ_5
6	−90°	0	0	θ_6–90°
7	−90°	0	0	θ_7

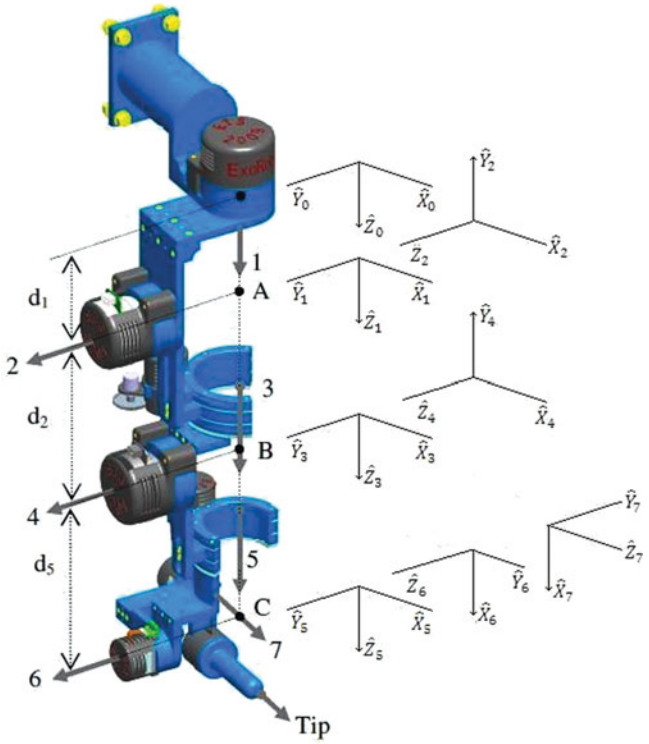


Fig. 2. 7DOF exoskeleton robot arm. Frame {0} represents the base frame; frame {1}: shoulder horizontal flexion/extension; frame {2}: shoulder vertical flexion/extension; frame {3}: shoulder internal/external rotation; frame {4}: elbow flexion/extension; frame {5}: forearm pronation/supination; frame {6}: wrist flexion/extension and frame {7}: wrist radial/ulnar deviation. Axes shown are in positive direction.

pronation/supination (Fig. 1e), and a wrist motion support part (for flexion/extension (Fig. 1f) and radial/ulnar deviation (Fig. 1g) of the wrist).

The exoskeleton is an open kinematic chain, manipulator-like robot, with seven revolute DOF ($n = 7$). The kinematics of any robot can be described with four parameters per joint, two describing the joint, and two for its connection with the adjacent joint. One parameter is the joint variable (angle if revolute, distance if prismatic), while the other three are constants (an angle or a distance, opposite of the joint variable and the angle and distance from joint to joint). This convention is known as the Denavit–Hartenberg notation; it has two variations, the original and the modified. As described in Craig,[26] with the frames attached to the robot as shown in Fig. 2, the modified Denavit–Hartenberg parameters are obtained and summarized in Table I.

3. Analysis of the ETS-MARSE Robot with Virtual Decomposition Control[27]

The first step in VDC analysis is to perform the virtual decomposition of the robot; in this case, the exoskeleton is a single open chain. This decomposition allows us to base control directly on subsystem dynamics, allowing it to remain relatively simple, i.e., proportional to the number of

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

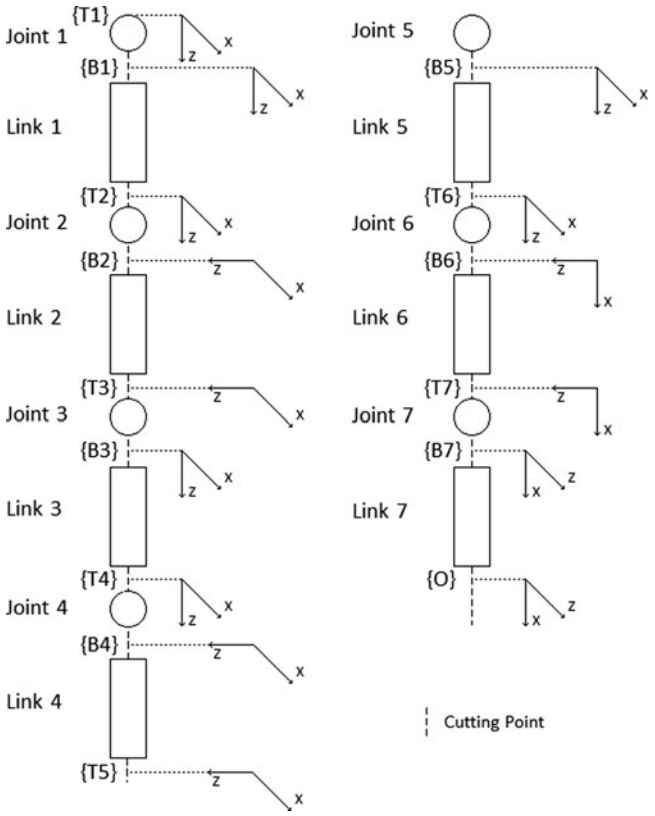


Fig. 3. Virtual decomposition into 14 subsystems.

subsystems. Decomposition is done by placing virtual cutting points separating each joint and link of the system into a subsystem. Although the robot or the human arm does not have seven links (see Fig. 2), this is a virtual decomposition, and is performed as such to express the system in its most basic operative units. This is a basic approach of VDC to maintain the equations of the system at their most simple expressions for computational handling; another advantage is that if a change is made to any actuator, only the dynamics of that joint will change, not the rest of the system. The result of the system decomposition is illustrated in Fig. 3: it yielded a decomposition into fourteen subsystems, with the virtual cutting points represented by vertical dotted lines, linked to their respective frames by horizontal dotted lines.

3.1. Kinematics

From the resulting modified Denavit–Hartenberg parameters for the decomposition shown in Fig. 3, the homogeneous transform matrices of the system are obtained. Using them, it is necessary to calculate the force/moment transformation matrices between any consecutive $\{B\}$ frames ${}^{B_i}U_{B_{i+1}}$ for $i = 1, \ldots, 6$ (see Fig. 3) as follows:

$$ {}^{B_i}\boldsymbol{U}_{B_{i+1}} = \begin{bmatrix} {}^{B_i}\boldsymbol{R}_{B_{i+1}} & 0_{3\times 3} \\ \left({}^{B_i}r_{B_{i+1}}\times\right){}^{B_i}\boldsymbol{R}_{B_{i+1}} & {}^{B_i}\boldsymbol{R}_{B_{i+1}} \end{bmatrix} \in \mathbb{R}^{6\times 6}, \tag{1} $$

where ${}^{B_i}\boldsymbol{R}_{B_{i+1}}$ represents the rotation matrix from frame B_i to frame B_{i+1} and ${}^{B_i}r_{B_{i+1}}$ the vector pointing from B_i to frame B_{i+1}, and

$$ \left({}^{B_i}r_{B_{i+1}}\times\right) = \begin{bmatrix} 0 & -{}^{B_i}r_{B_{i+1}}(3) & {}^{B_i}r_{B_{i+1}}(2) \\ {}^{B_i}r_{B_{i+1}}(3) & 0 & -{}^{B_i}r_{B_{i+1}}(1) \\ -{}^{B_i}r_{B_{i+1}}(2) & {}^{B_i}r_{B_{i+1}}(1) & 0 \end{bmatrix}, \tag{2} $$

where ${}^{B_i}r_{B_{i+1}}(i)$ represents the ith term of the ${}^{B_i}r_{B_{i+1}}$ vector.

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

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

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

where ${}^{B_i}v$ is the linear velocity vector and ${}^{B_i}\omega$ is the angular velocity vector of the corresponding frame. Defining each joint position (angle) as q_1 to q_7 and using the previous linear/angular velocity vectors in (3), the augmented velocity vector is formed as follows:

$$ \mathcal{V}_7 = \left[\dot{q}_1 \cdots \dot{q}_7 \ {}^{B_1}V^T \cdots {}^{B_7}V^T \right]^T. \tag{4} $$

Factorizing the joint velocity vector $\dot{q} = [\, \dot{q}_1 \quad \cdots \quad \dot{q}_7 \,]^T$ in (4), we obtain the following:

$$ \mathcal{V}_7 = \boldsymbol{J}_7 \dot{q}, \tag{5} $$

where $\boldsymbol{J}_7$ is the VDC Jacobian matrix of the system:

$$ \boldsymbol{J}_7 = \begin{bmatrix} & \boldsymbol{I}_7 & & \\ z & 0_6 & \cdots & 0_6 \\ {}^{B_1}\boldsymbol{U}^T_{B_2} z & z & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0_6 \\ {}^{B_1}\boldsymbol{U}^T_{B_7} z & \ldots & {}^{B_6}\boldsymbol{U}^T_{B_7} z & z \end{bmatrix} \tag{6} $$

where $z = [\, 0\ 0\ 0\ 0\ 0\ 1 \,]^T$, $\boldsymbol{I}_7$ represents the 7×7 identity matrix and 0_6 is a six-element zero vector. Note that the dimension of $\boldsymbol{J}_7$ is 49×7; the aforementioned identity matrix and a block of 6×7 for each one of the seven DOF of the robot.

Finally, for the kinematics, it is necessary to calculate the required velocities, which are functions of the control requirements. They incorporate terms that allow the tracking of a control objective to the desired velocities obtained from the trajectory generator. In this case, position control is used such that the position is incorporated into the vector of required velocities.

$$ \dot{q}_r = \dot{q}_d + \lambda \left(q_d - q \right), \tag{7} $$

where $\dot{q}_r = [\, \dot{q}_{1r} \cdots \dot{q}_{7r} \,]^T$ is a 7×1 reference velocity vector, $\dot{q}_d = [\, \dot{q}_{1d} \cdots \dot{q}_{7d} \,]^T$ is the 7×1 desired velocity vector, $\lambda > 0$ is a control parameter, $q_d = [\, q_{1d} \cdots q_{7d} \,]^T$ is the 7×1 desired position vector and $q = [\, q_1 \cdots q_7 \,]^T$ the 7×1 position vector. From (6) and (7), the 49×1 augmented required velocity vector is obtained as follows:

$$ \mathcal{V}_{7r} = \boldsymbol{J}_7 \dot{q}_r \tag{8} $$

From (4), it can be established that:

$$ \mathcal{V}_{7r} = \left[\dot{q}_{1r} \cdots \dot{q}_{7r} \ {}^{B_1}V_r^T \cdots {}^{B_7}V_r^T \right]^T. \tag{9} $$

3.2. Dynamics

For the simulation, the dynamics of the robot was obtained using the Newton–Euler approach.[26] To apply the VDC approach and to obtain the inverse dynamics that takes part in the control of the robot, it was necessary to calculate the required force/moment vectors for the links, ${}^{B_i}F_r$, and then the net torques required by the joints, τ^*_{ir}.

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

From the general formulation of a rigid body dynamics, in which one frame $\{A\}$ is used as the reference to express the dynamics, and another frame $\{B\}$ is assumed to be located at the centre of mass, the rigid body dynamics can be expressed as following:

$$\boldsymbol{M}_A \frac{d}{dt}\left({}^A V\right) + \boldsymbol{C}_A\left({}^A\omega\right){}^A V + G_A = {}^A F^*, \tag{10}$$

where[27]

$$\boldsymbol{M}_A = \begin{bmatrix} m_A \boldsymbol{I}_3 & -m_A\left({}^A r_{AB}\times\right) \\ m_A\left({}^A r_{AB}\times\right) & \boldsymbol{I}_A - m_A\left({}^A r_{AB}\times\right)^2 \end{bmatrix} \tag{11}$$

$$\boldsymbol{C}_A\left({}^A\omega\right) = \begin{bmatrix} m_A\left({}^A\omega\times\right) & -m_A\left({}^A\omega\times\right)\left({}^A r_{AB}\times\right) \\ m_A\left({}^A r_{AB}\times\right)\left({}^A\omega\times\right) & \left({}^A\omega\times\right)\boldsymbol{I}_A + \boldsymbol{I}_A\left({}^A\omega\times\right) - m_A\left({}^A r_{AB}\times\right)\left({}^A\omega\times\right)\left({}^A r_{AB}\times\right) \end{bmatrix} \tag{12}$$

$$G_A = \begin{bmatrix} m_A {}^A\boldsymbol{R}_I g \\ m_A\left({}^A r_{AB}\times\right){}^A\boldsymbol{R}_I g \end{bmatrix} \tag{13}$$

where $m_A \in \mathbb{R}$ is the mass of the rigid body, $\boldsymbol{I}_3$ is a 3×3 identity matrix, $\boldsymbol{I}_A = {}^A\boldsymbol{R}_I \boldsymbol{I}_o(t) {}^I\boldsymbol{R}_A$ is time invariant, $\boldsymbol{I}_o(t) \in \mathbb{R}^{3\times3}$ denotes the moment of inertia matrix around the centre of mass, ${}^A\boldsymbol{R}_I \in \mathbb{R}^{3\times3}$ is the rotation matrix from frame A to the inertial frame, ${}^A\mathrm{r}_{AB} \in \mathbb{R}^3$ is the vector pointing from frame A to frame B, ${}^A\omega \in \mathbb{R}^3$ is the angular velocity vector, $g = [\,0\ 0\ 9.81\,]^T \in \mathbb{R}^3$ and $({}^A\omega\times)$ and $({}^A r_{AB}\times)$ are defined as in relation (2).

Substituting from (10) the dynamic model velocity vector ${}^A V$ by the required velocity vector ${}^A V_r \in \mathbb{R}^6$ defined in (9), the following equation is defined:

$$\boldsymbol{M}_A \frac{d}{dt}\left({}^A V_r\right) + \boldsymbol{C}_A\left({}^A\omega\right){}^A V_r + G_A \stackrel{\text{def}}{=} \boldsymbol{Y}_A \theta_A, \tag{14}$$

where $\boldsymbol{Y}_A \in \mathbb{R}^{6\times13}$ is a regressor matrix composed of elements of $d/dt({}^A V_r) \in \mathbb{R}^6$, the velocity vectors ${}^A V \in \mathbb{R}^6$ and ${}^A V_r \in \mathbb{R}^6$ and terms from the vector of gravity propagation, and $\theta_A \in \mathbb{R}^{13}$ is the 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, and elements of the inertia tensor of the body. The exact representation of each element of $\boldsymbol{Y}_A$ and θ_A is given in Appendix A.

By means of the linear parameterization expressed above in (14), the first step for the link dynamics is to obtain the required net force/moment vectors with the following equation:

$${}^{B_i} F_r^* = \boldsymbol{Y}_i \hat{\theta}_i + \boldsymbol{K}_{si}\left({}^{B_i} V_r - {}^{B_i} V\right) \tag{15}$$

for $i = 1, \ldots, 7$ where ${}^{B_i} F_r^*$ are the required net force/moment vectors, K_{si} is a symmetric positive-definite gain matrix, ${}^{B_i} V$ and ${}^{B_i} V_r$ are the velocities defined in (4) and (9), and $\hat{\theta}_i \in \mathbb{R}^{13}$ is the estimated parameter vector. The estimation of the parameters in $\hat{\theta}_i$ is defined as follows:[28]

$$\dot{\hat{\theta}}_{i\gamma} = \mathcal{P}\left(s_{i\gamma}(t), \rho_{i\gamma}, a_{i\gamma}(t), b_{i\gamma}(t), t\right) \tag{16}$$

for $i = 1, \ldots, 7$ and $\gamma = 1, \ldots, 13$, where $\hat{\theta}_{i\gamma}$ represents the γth parameter of the ith link, $s_{i\gamma}(t) \in \mathbb{R}$ is a scalar variable defined as the γth element in:

$$s_i = \boldsymbol{Y}_i^T\left({}^{B_i} V_r - {}^{B_i} V\right). \tag{17}$$

$\rho_{i\gamma}$ 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 the projection function $\mathcal{P}$ is a differentiable scalar function[29] defined for $t \geq 0$ such that its time

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

derivative is governed by

$$\dot{\mathcal{P}} = \rho s\left(t\right)\mathcal{K} \tag{18}$$

with

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

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

$$\begin{aligned} {}^{B_7}F_r &= {}^{B_7}F_r^* \\ {}^{B_i}F_r &= {}^{B_i}F_r^* + {}^{B_i}U_{B_{i+1}}^{B_{i+1}}F_r \end{aligned} \tag{19}$$

for $i = 6, \ldots, 1$. The next step is the calculation of the dynamics of the joints. For $i = 1, \ldots, 7$ in (20)–(24), the following Y vector and parameter vector are defined:

$$Y_{ai} = \begin{bmatrix} \ddot{q}_{ir} & sign\left(\dot{q}_{ir}\right) & \dot{q}_{ir} & 1 \end{bmatrix} \tag{20}$$

$$\theta_{ai} = \begin{bmatrix} J_{mi} & k_{ci} & k_{vi} & c_i \end{bmatrix}^T \tag{21}$$

where the sub-index a in the vectors is used to differentiate joint variables from the link variables introduced in (15), J_{mi} is the equivalent mass or moment of inertia, $k_{ci} > 0$ denotes the Coulomb friction coefficient, $k_{vi} > 0$ is the viscous friction coefficient and c_i denotes an offset that accommodates asymmetric Coulomb frictions. The adaptation of the parameters defined in (21) is calculated as following:

$$\hat{\theta}_{ai\gamma} = \mathcal{P}\left(s_{ai\gamma}\left(t\right), \rho_{ai\gamma}, a_{ai\gamma}\left(t\right), b_{ai\gamma}\left(t\right), t\right) \tag{22}$$

for $i = 1, \ldots, 7$ and $\gamma = 1, \ldots, 4$, where $\hat{\theta}_{ai\gamma}$ represents the γth parameter of the ith joint, $s_{ai\gamma}(t) \in \mathbb{R}$ is a scalar variable defined as the γth element in

$$s_{ai} = Y_{ai}^T\left(\dot{q}_{ir} - \dot{q}_i\right). \tag{23}$$

$\rho_{ai\gamma}$ is the parameter update gain, $a_{ai\gamma}(t)$ and $b_{ai\gamma}(t)$ are the lower and upper bounds of $\hat{\theta}_{ai\gamma}$ respectively, and the projection function $\mathcal{P}$ is a differentiable scalar function defined in (18).

Finally, the net torque for the joint τ_{ir}^* is obtained as follows:

$$\tau_{ir}^* = Y_{ai}\hat{\theta}_{ai} + k_{ai}\left(\dot{q}_{ir} - \dot{q}_i\right), \tag{24}$$

where τ_{ir}^* is the net torque of the joint, k_{ai} denotes a feedback gain and $\hat{\theta}_{ai} \in \mathbb{R}^4$ defined in (22) denotes the estimate of $\theta_{ai} \in \mathbb{R}^4$ defined in (21).

3.3. Control

To develop the control of the entire system, the torques resulting from the link and joint analysis are combined. First, it is necessary to extract the torque of the link force/moment vector τ_{air} as follows:

$$\tau_{air} = z^{T\,B_i}F_r \tag{25}$$

for $i = 1, \ldots, 7$. With $z = [\,0\;0\;0\;0\;0\;1\,]^T$ that was defined earlier, the control torque τ_i is designed as:

$$\tau_i = \tau_{ir}^* + \tau_{air} \tag{26}$$

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

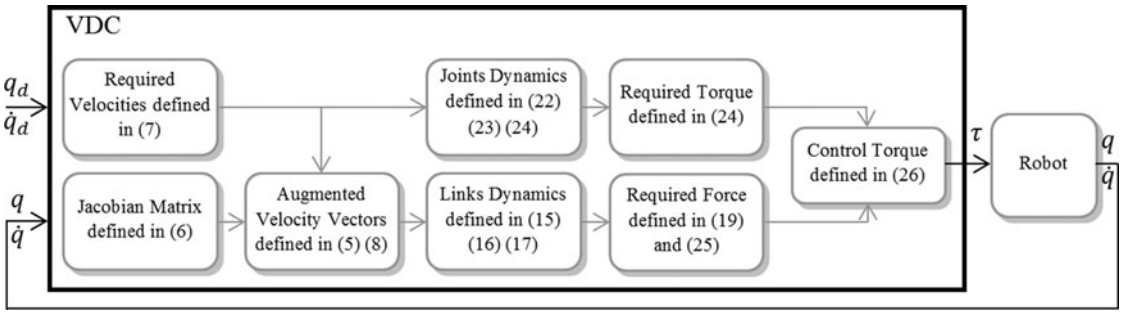


Fig. 4. Block diagram of the system (with the equations used in each step in the parentheses).

for $i = 1, \ldots, 7$. The flow diagram with the calculations needed in the VDC technique is shown in Fig. 4.

Associating (26) with the traditional dynamic equation of a manipulator:

$$\tau = \boldsymbol{M}(\theta)\ddot{\theta} + V\left(\theta, \dot{\theta}\right) + G(\theta) + F\left(\theta, \dot{\theta}\right), \tag{27}$$

where $\boldsymbol{M}(\theta)$ is the manipulator's $n \times n$ mass matrix, $V(\theta, \dot{\theta})$ the $n \times 1$ Coriolis and centrifugal terms vector, $G(\theta)$ the $n \times 1$ gravity terms vector and $F(\theta, \dot{\theta})$ the $n \times 1$ torque friction vector. It is clear that in the case of VDC, the parameter adaptation that is defined in (15) and (24) from Eqs. (10)–(14) for the rigid links and (20)–(21) for the joints, will incorporate not only the mass, inertia (moments and products) and centre of mass parameters of the links, and the moment of inertia and friction for the joints of the robot, but also the uncertainties and the disturbances introduced by the subject, which are not modelled. For example, the mass of the human arm (different from subject to subject) will be combined with the links' mass; changes or displacements in the position of the arm of the subject, combined with changes in position of the centre of mass; and friction of the joints, combined with constraints of the movement caused by the subject's limitations, such as difficulty to move due to spasticity or altered muscle tone.

3.4. Stability analysis

Given the dynamics expressed in relation (27) and the control design given in (19) and (24), combined as indicated in (25) and (26), the asymptotic stability is ensured if the position and velocity errors converge asymptotically to zero:

$$lim_{t\to\infty} q(t)_d - q(t) \to^0 \tag{28}$$

$$lim_{t\to\infty} \dot{q}(t)_d - \dot{q}(t) \to^0 \tag{29}$$

with q and q̇ vectors defined in (4) and (5) and q_d and $\dot{q}_d$ vectors defined in (7).

Using the Lyapunov approach, a non-negative candidate function is defined and then it is shown that the variation is a decreasing function. As explained in the previous section, the system dynamics has two parts. The first represents the link dynamics given by

$$M_{Bi}\frac{d}{dt}\left({}^{Bi}V\right) + C_{Bi}\left({}^{Bi}\omega\right){}^{Bi}V + G_{Bi} = {}^{Bi}F* \qquad i = 1\ldots n. \tag{30}$$

And the second, given in relation (31), represents the joint dynamics, characterized by the torque resultant from the moment of inertia plus the torque from the Coulomb friction:

$$J_{mi}\ddot{q}_i + k_{ci}sign(\dot{q}_i) = \tau_i^* \qquad i = 1\ldots n. \tag{31}$$

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

Therefore, considering the Lyapunov candidate function for the entire robot as following:

$$v_L = \sum_{i=1}^{7} v_i + \sum_{i=1}^{7} v_{ai}, \tag{32}$$

where v_i represents the non-negative Lyapunov candidate function related to the link dynamics and v_{ai} is the non-negative Lyapunov candidate function related to the joint dynamics. Combined with the control Eq. (15) and the parameter adaptation defined in (16) and (17), v_i is defined as follows:

$$v_i = \frac{1}{2}\left({}^{B_i}V_r - {}^{Bi}V\right)^T \boldsymbol{M}\left({}^{B_i}V_r - {}^{Bi}V\right) + \frac{1}{2}\sum_{\gamma=1}^{13}\frac{\left(\theta_{i\gamma} - \hat{\theta}_{i\gamma}\right)^2}{\rho_{i\gamma}}. \tag{33}$$

The non-negative Lyapunov candidate function related to v_{ai} can be defined as follows:

$$v_{ai} = \frac{1}{2}J_{mi}\left(\dot{q}_{ir} - \dot{q}_i\right)^2 + \frac{1}{2}\sum_{\gamma=1}^{4}\frac{\left(\theta_{ai\gamma} - \hat{\theta}_{ai\gamma}\right)^2}{\rho_{i\gamma}}. \tag{34}$$

The derivative of the Lyapunov candidate function (32) is given as:

$$\dot{v}_L = \sum_{i=1}^{7} \dot{v}_i + \sum_{i=1}^{7} \dot{v}_{ai}. \tag{35}$$

Before showing that relation (35) is always decreasing, it is necessary to define the virtual power flows as the inner product of the linear/angular velocity vector error and the force/moment vector error.[27,30] With respect to a frame A:

$$p_A \overset{\text{def}}{=} \left({}^{A}V_r - {}^{A}V\right)^T \left({}^{A}F_r - {}^{A}F\right), \tag{36}$$

virtual power flows characterize the dynamic interaction between subsystems at its cutting points. They are analogue to power flows within a rigid body, but instead of being the inner product of a velocity vector and a force vector, they are the inner product of a velocity vector error and a force vector error. With definition (36), it follows that:

$$\dot{v}_i \leq -\left({}^{B_i}V_r - {}^{Bi}V\right)^T K_{si}\left({}^{B_i}V_r - {}^{Bi}V\right) + \left({}^{B_i}V_r - {}^{Bi}V\right)^T \left({}^{B_i}F_r^* - {}^{Bi}F^*\right). \tag{37}$$

Considering (36) and the force/moment transformation matrices defined in (1) to transform the velocities expressed in (3) and the required velocities expressed in (9), and with the net force/moment and required net force/moment vectors (19), this yields

$$\left({}^{B_i}V_r - {}^{Bi}V\right)^T \left({}^{B_i}F_r^* - {}^{Bi}F^*\right) = p_{B_i} - p_{T_i}, \tag{38}$$

where the terms on the right side of (38) represent the virtual power flows at the two cutting points of each link.

$\dot{v}_{ai}$ can be written[27] as

$$\dot{v}_{ai} \leq -k_{ai}\left(\dot{q}_{air} - \dot{q}_{ai}\right)^2 - p_{B_{ai}} + p_{T_{ai}}. \tag{39}$$

Considering an open chain structure, as is the case in this paper, and with $p_{B_1} = 0$ and $p_{T_n} = 0$, the total virtual power flows give

$$\sum_{i=1}^{n}\left(p_{B_i} - p_{T_i}\right) = 0. \tag{40}$$

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

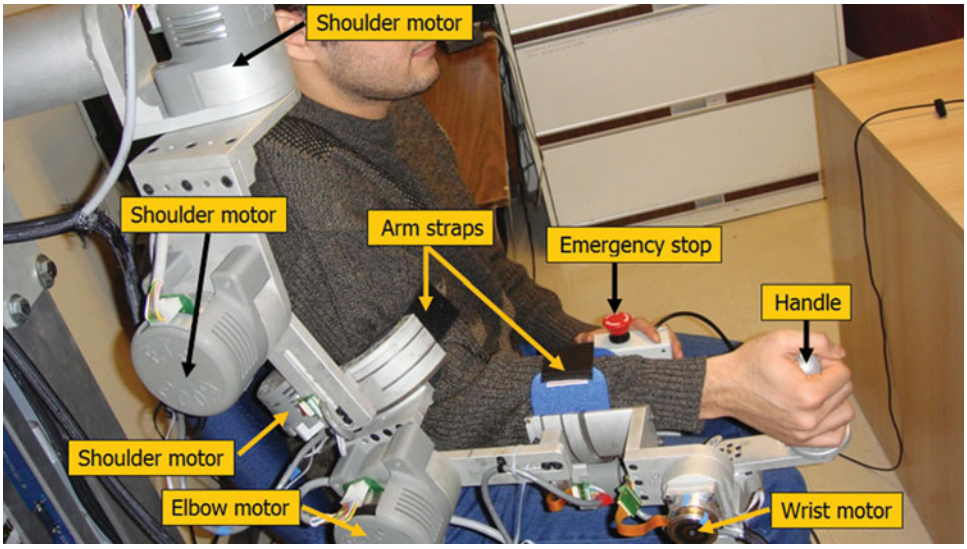


Fig. 5. ETS-MARSE with human subject.

Therefore, relationship (37) becomes

$$\dot{v}_i \leq -\left({}^{B_i}V_r - {}^{Bi}V\right)^T K_{si} \left({}^{B_i}V_r - {}^{Bi}V\right) \tag{41}$$

and the function $\dot{v}_{ai}$ becomes

$$\dot{v}_{ai} \leq -k_{ai} \left(\dot{q}_{air} - \dot{q}_{ai}\right)^2 . \tag{42}$$

Finally, from relationships (41) and (42), and given positive gains K_{si} and k_{ai}, the derivative of the Lyapunov function $\dot{v}_L$, given in (43), is always decreasing.

$$\dot{v}_L \leq -\sum_{i=1}^{n} \left(\left({}^{B_i}V_r - {}^{Bi}V\right)^T K_{si} \left({}^{B_i}V_r - {}^{Bi}V\right) + k_{ai} \left(\dot{q}_{air} - \dot{q}_{ai}\right)^2 \right) . \tag{43}$$

The asymptotic stability in the sense of Lyapunov follows immediately from the bounded reference signals, the bounded control and the bounded joint acceleration.

4. Experimental Results

A general overview of the ETS-MARSE robot is shown in Fig. 5. Brushless DC motors (Maxon EC-45, EC-90) incorporated with harmonic drives (having gear ratio 120:1 for motor 1 and motor 2, and gear ratio 100:1 for motors 3–7) were used to actuate the ETS-MARSE.

A block diagram of the experimental setup, including the control architecture, for the ETS-MARSE system is depicted in Fig. 6. For the experimental phase, three control techniques were used: VDC as the focus of this research and CTC and PID as reference control strategies. The outputs of the controllers were the joints' torque commands. However, the torque commands were converted to motor currents, and finally, to reference voltages as voltage value is the drive command for the motor drivers. The implementation of the system was carried out in a National Instruments PXI system. A PXI-8108 controller card performed the joint-based control (VDC, CTC and PID) and served as the robot operating system. A FPGA (field programmable gate array) mounted in the NI PXI-7813R remote input–output card, performed the low-level (PI) control, that is, the current loop for the robot actuators and the position feedback control via Hall sensors. The sampling rate for the VDC controller was 1ms and 500 μ s for the CTC and PID (Fig. 6, left dotted box). Note that the current loop controller has a sampling time of 50 μ s (up to 20 times faster) than the torque control loop (Fig. 6, right dotted box). Also, a host computer was used (not shown in the schematic of Fig. 6), which displayed the

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

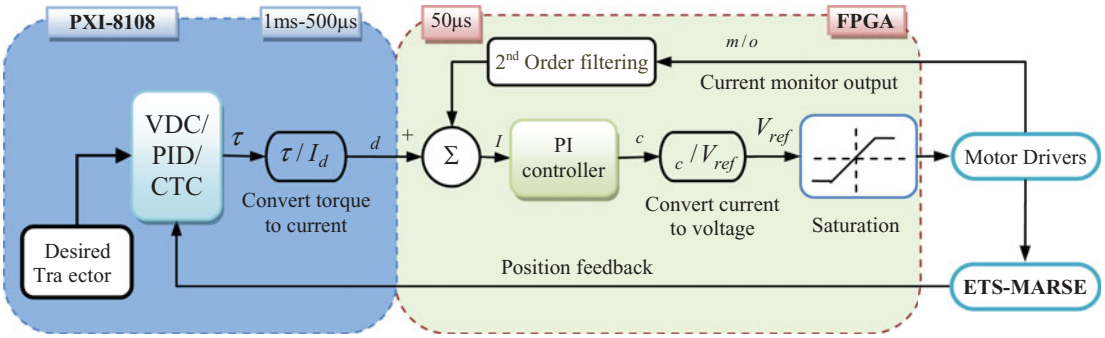


Fig. 6. Experimental setup with control architecture.

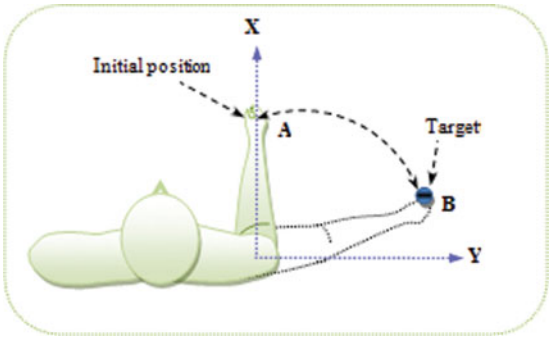


Fig. 7. Schematic diagram, reaching movement exercise.

user interface through which it connected to the ETS-MARSE system in order to issue commands (such as selection of exercise or control techniques) to the PXI, and to provide feedback and retrieve experimental data from the system for analysis, as well as the virtual environment used in active rehabilitation.

4.1. Varying subjects experiment

In these experiments, the variation of the trajectory tracking performance of the ETS-MARSE was evaluated. The experiment was divided into three scenarios: one with the VDC, another with the PID, and the last with the CTC. Although the CTC controller can include a simple PID controller for the decoupling and cancelation of nonlinear terms, in this case CTC was used with a PD controller. The PID controller described in these tests is different; as a traditional PID controller, it attempts to minimize the position error through the adjustment of the torque without any need for dynamic modelling of the robot, and only some tuning of the gains is performed. For the CTC controller (with its internal PD), a complete modelling of the robot's dynamics should be done. The trajectory used in this research represents a typical passive rehabilitation exercise.[31] It is a joint space trajectory, an outside reaching movement, illustrated in Fig. 7, and involving the shoulder, elbow and forearm motion: in each exercise, the robot followed the path from A to B (in four seconds) and then reversed direction from B to A (also in four seconds) three times.

The range of motion of each joint is summarized in Table II; since the wrist is not moved in this experiment (joint 6 and joint 7), only joints 1–5 are presented. To introduce the system variability, this exercise was executed with four different situations: three healthy male human subjects (ages: 24, 33 and 31 years; height: 181, 165 and 182 cm; weight: 75, 63 and 95 kg) and one scenario without a subject. Experiments with subjects were conducted with them in the seated position, analogous to Fig. 5. A height-adjustable chair was used to adjust the seating height, i.e., to align the centre of rotation of the shoulder joint of the subject to that of the ETS-MARSE.

It is important to recall that this rehabilitation movement is a passive one; that is, the robot follows a predefined trajectory with no action from the user. Each condition (i.e., exercises with subjects A, B and C, and without subject) was repeated seven consecutive times, giving a total of 28 trials for

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

Table II. Varying subjects experiment range of motion.

Joint	***Type of motion***	**ETS-MARSE *workspace***
Shoulder joint		
Joint 1	Horizontal flexion	5°
	Horizontal extension	60°
Joint 2	Vertical flexion	60°
	Vertical extension	0°
Joint 3	Internal rotation	20°
	External rotation	0°
Elbow		
Joint 4	Flexion	90°
	Extension	40°
Forearm		
Joint 5	Pronation	40°
	Supination	0°

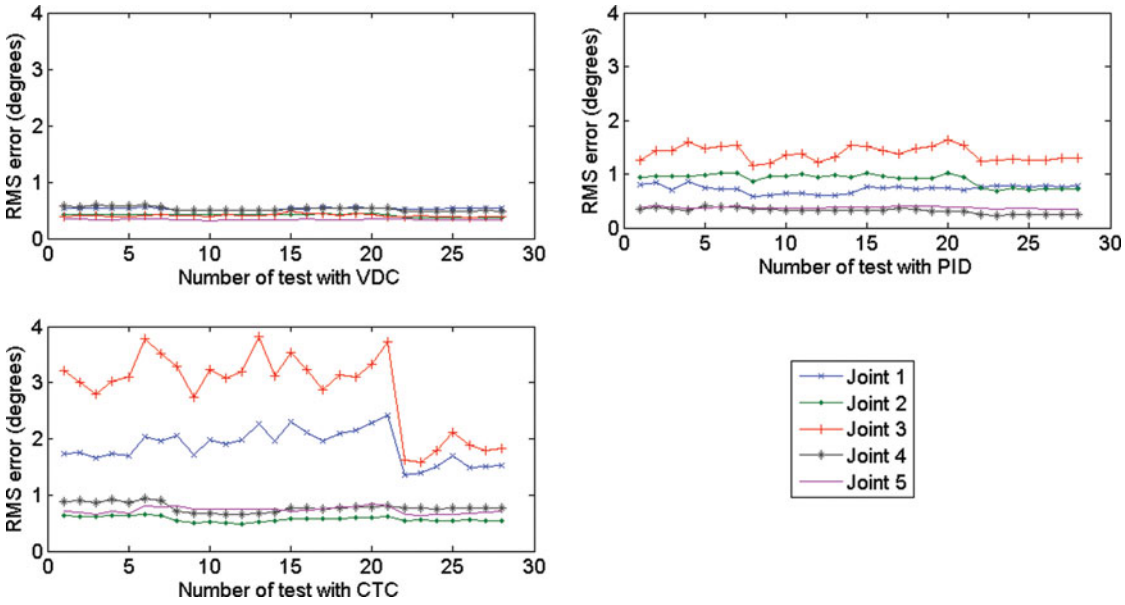


Fig. 8. RMS joint tracking error along the 28 tests for each controller.

each controller. It should be mentioned that no changes were made between the tests, e.g., in terms of gains or parameters in the control strategy.

The experimental results are presented in Fig. 8. Each plot corresponds to one control strategy, and each one shows the RMS value of the trajectory tracking error for each joint moved, along the 28 tests. It should be recalled that even when all joints are involved in the control, joint 6 and joint 7 of the ETS-MARSE remained at zero degrees, and so were not analysed. It is clear that, in addition to providing a lower tracking error, the VDC controller remained barely affected by variations. Meanwhile, the PID controller presented a more variable behaviour, and the CTC was the most affected by changes in the subject's characteristics.

The three controllers performed well, but the analysis that is pertinent for this paper is the variation in the performance from one experiment to another due to the dynamic variations. In Fig. 9, the average RMS error (joints 1–5) of each test and for each controller is shown. It can easily be seen how the VDC results are more compact, confirming that the controller behaves better over distinct dynamic conditions caused by changes of test subjects. It should be noted that the changes from one condition to the other are also present in the VDC, but the variation is small compared to the other techniques used in this paper. It is also clearly seen that the other controllers changed their behaviour

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

Table III. Statistical analysis of performance variation.

Controller	*Joint 1*	*Joint 2*	*Joint 3*	*Joint 4*	*Joint 5*	*Average RMS Error* $\sigma(^\circ)$	*Normalize to VDC RMS Error* σ
	RMS Error $\sigma(^\circ)$						
VDC	0.0180	0.0251	0.0243	0.0330	0.0086	0.0218	1.00
PID	0.0703	0.1116	0.1314	0.0483	0.0187	0.0761	3.49
CTC	0.2887	0.0455	0.6731	0.0789	0.0552	0.2283	10.47

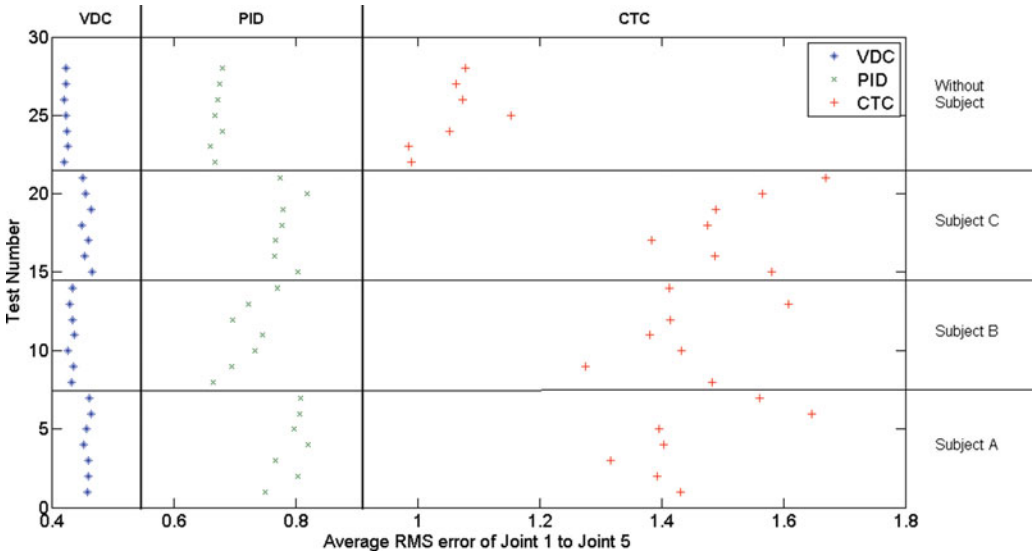


Fig. 9. Average (joints 1–5) RMS error of each test.

notably even under the same testing condition, that is, with the same subject (or without subject), possibly due to natural, unintended changes in the arm position, relaxation, grip on the handle, etc.

Finally, in order to provide a quantitative analysis, Table III summarizes the standard deviation (σ) of the data plotted in Figs. 8 and 9. The second to the sixth columns represent the standard deviation of the RMS error for each joint (Fig. 8). The seventh column is the standard deviation of the average RMS error across joints one to five (Fig. 9). For a better comparison, the average RMS error was normalized to that of the VDC in the last column. The PID showed an average RMS error standard deviation more than 3 times larger than the VDC; and for the CTC, this was more than 10 times. It can clearly be seen that a controller that depends heavily on the dynamic model of the system (CTC) is significantly affected by changes and uncertainties caused by the variations in the robot's user.

4.2. Varying velocity experiment

Another important concern in rehabilitation is movement velocity. To show the efficiency of the VDC approach in dealing with different velocities, an elbow flexion/extension experiment is shown. The trajectory of joint 4 (elbow joint) passes through 90°, 5°, 115°and 0°. The speed of the movement was increased for each of the three tests performed. In the first one, the maximum speed was 20.63°/s, it was 45.20°/s for the second one, and −85.71°/s for the third test (the minus sign indicates the extension movement). The tests were conducted with no subject wearing the robot. To compare performances, PID and VDC were used in each test. Figure 10 shows the results for the VDC controller, where each column displays the position, velocity and error for each movement velocity. Figure 11 shows the same for the PID controller. To summarize the performance, Table IV shows the RMS error in each test. It can be seen that the tracking error for the VDC increased by 83% from the slowest to the fastest movement, while for PID the increase was by 204%.

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

Table IV. RMS error of varying velocity experiments.

Maximum speed (degrees per second)	***RMS Error J4 (°) VDC***	***RMS Error J4 (°) PID***
20.63	0.3914	0.3695
45.20	0.4719	0.6209
85.71	0.7169	1.1228

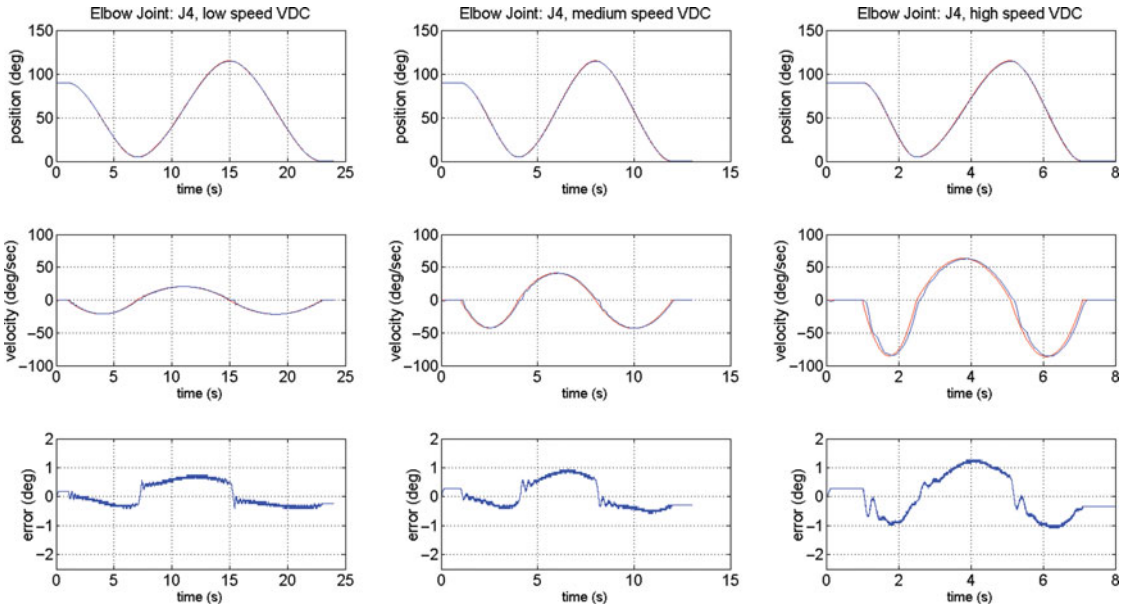


Fig. 10. VDC results for different speeds test.

4.3. Active rehabilitation experiment

Next in the experiment, an active rehabilitation test is presented. Through a virtual environment as the human machine interface (HMI),[32] the user's task was to follow a predefined trajectory, in this case, a straight line to the front and back, that is, movement on the X-axis (see Fig. 7). With the use of a six-axis force sensor mounted on the tip of the robot (refer to Fig. 2 or Fig. 5) the trajectory was actively modified by an admittance function to make the robot follow the user's movement intentions. The tip of the robot, the end effector, is in the handle that the user grabs with his hand while wearing the robot. The test was completed with the VDC controller. Figure 12 shows the user performing the test with a zoom of the virtual interface in the upper right corner. The red line in the centre represents the path the user should follow; the white sphere represents the user end effector, and the green line, the actual trajectory followed by the subject. Finally, Fig. 13 shows the movement in the Cartesian space in the first column, the force readings of the force sensor in each Cartesian axis in the second column, and the torque readings in the last column. Thanks to the robustness of the VDC controller described in Subsections 4.1 and 4.2, the variations in trajectory tracking observed in this active rehabilitation exercise were only due to the user's conditions. This will give a more accurate measurement of the user performance from test to test.

4.4. Parameter adaptation

Finally, to illustrate the parameter adaptation process, a simple experiment was performed. The adaptation process is shown in the parameters of joint 4. The robot was set at an initial position with all joints as shown in Fig. 2, except for joint 4, which was set at 90°. All the parameters were then reset to zero and the trajectory commanded to the robot only affected joint 4. The test was performed without a subject. It stayed for 1 s at the initial position, and then moved from 90°to 5°in 2 s, and stayed there for another 2 s. The trajectory is shown in Fig. 14. Next, Fig. 15 shows the joints parameters adaptation, defined in (21), J_{mi}, the equivalent moment of inertia, k_{ci}, the Coulomb

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

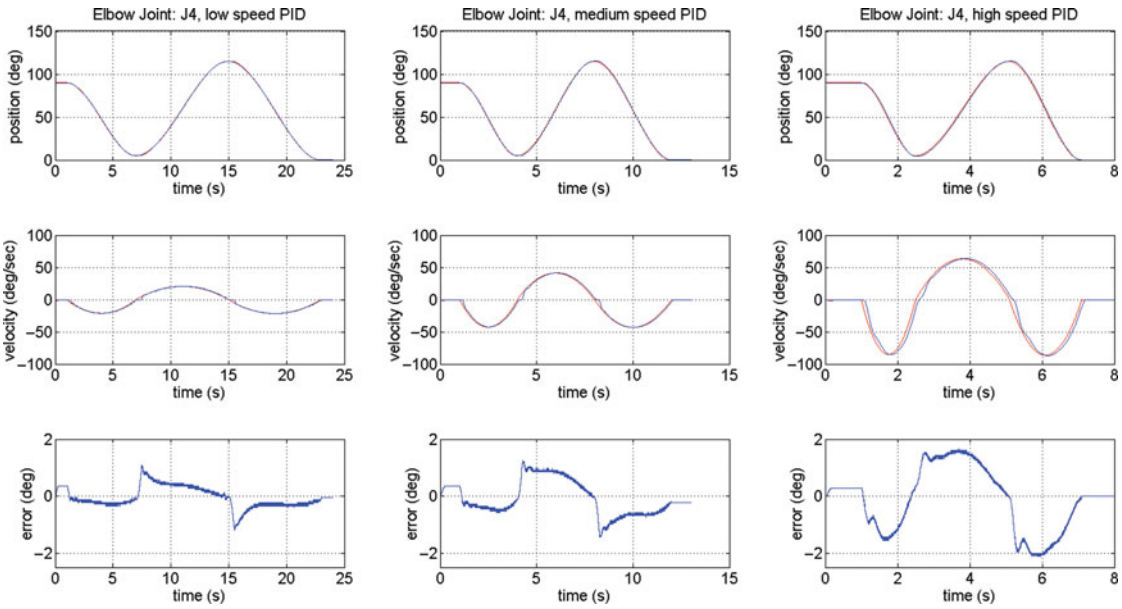


Fig. 11. PID results for different speeds test.

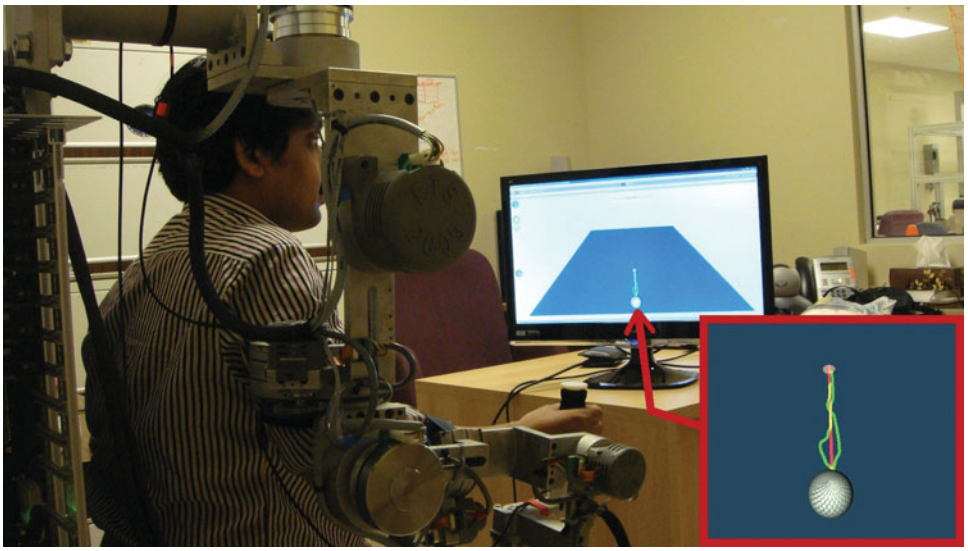


Fig. 12. Experimental setup for active rehabilitation.

friction coefficient, k_{vi}, the viscous friction coefficient and c_i, an offset that accommodates asymmetric Coulomb frictions. It should be recalled that in the VDC, the algorithm guarantees convergence to a certain value that will ensure the fulfilment of the control premise as well as the stability demonstrated in section 3.5, but these values are not necessarily the real values of the model of the robot. Moreover, it is important to remember that the gain for the parameter adaptation was individually adjusted for each parameter, as defined in (22). The plots show how the first adaptation was done from a zero value to a certain value corresponding to the initial position. From time = 1 s to time = 3 s, the parameters adapted according to the dynamics variations of the system in motion. Finally, arriving at another steady position from time = 3 s onwards, the parameters converge to another value from there to 5 s.

4.5. Discussion of experimental results

The controllers can be tuned in an attempt to achieve a better performance (especially in PID and CTC cases); nevertheless, the tuning process of the gains or changes in the dynamic model would

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

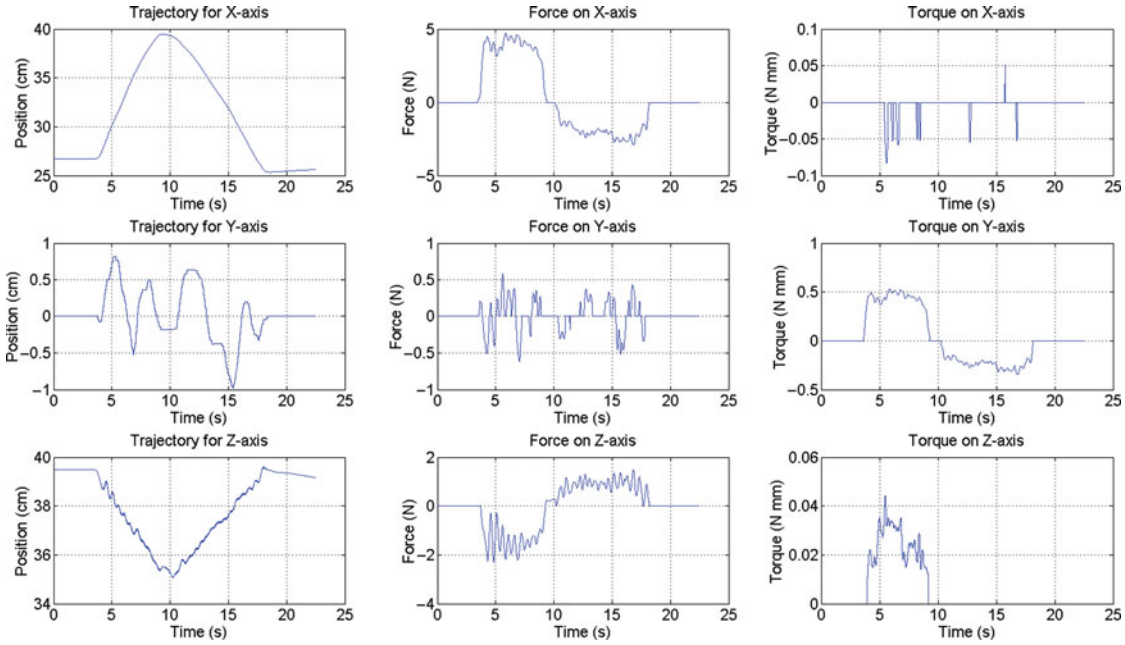


Fig. 13. Cartesian trajectory, forces and torques for active rehabilitation.

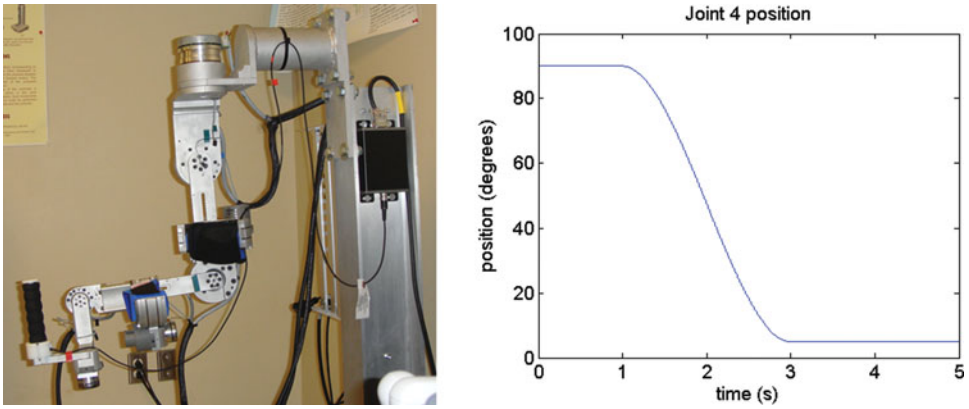


Fig. 14. Left: robot initial position. Right: joint 4 trajectory.

need to be repeated for each user. In rehabilitation, if the PID and the CTC were adjusted for one specific user, this procedure should be repeated for new users with different dynamics (weight, height, neurological condition, etc.). However, this procedure is not needed with the VDC because it is based on an adaptive algorithm; the adaptation process is done online without any manual modification of parameters, which constitutes a big advantage. Making changes to the control system dynamics in controllers like the CTC is an undesirable and time-consuming task for the therapist. Besides, physiological changes due to an improvement in the subjects' condition (such as stiffness) can be another source of periodical variances (reduced resistance with time). These problems are clearly avoided by the use of the VDC. Further, in the experiment presented in Section 4.2, it can be seen that at low velocity, the PID and VDC controllers perform almost equally well. It is crucial to understand that this reflects proper tuning of the gains. Thus, the variations in the system dynamics (in this case, movement velocity) are what lead to a worsening of performance for the PID controller. It should be mentioned that the velocity test was performed without a subject in a bid to specifically isolate the effects of the variations in movement velocity. The results from Sections 4.1 and 4.2 should be considered together, thus supporting the need, in rehabilitation, to have a controller that is robust to changes in biomechanical and physiological conditions. Finally, the active rehabilitation scenario shows that in an active exercise, the tracking of the patient's improvement will be better known, since

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

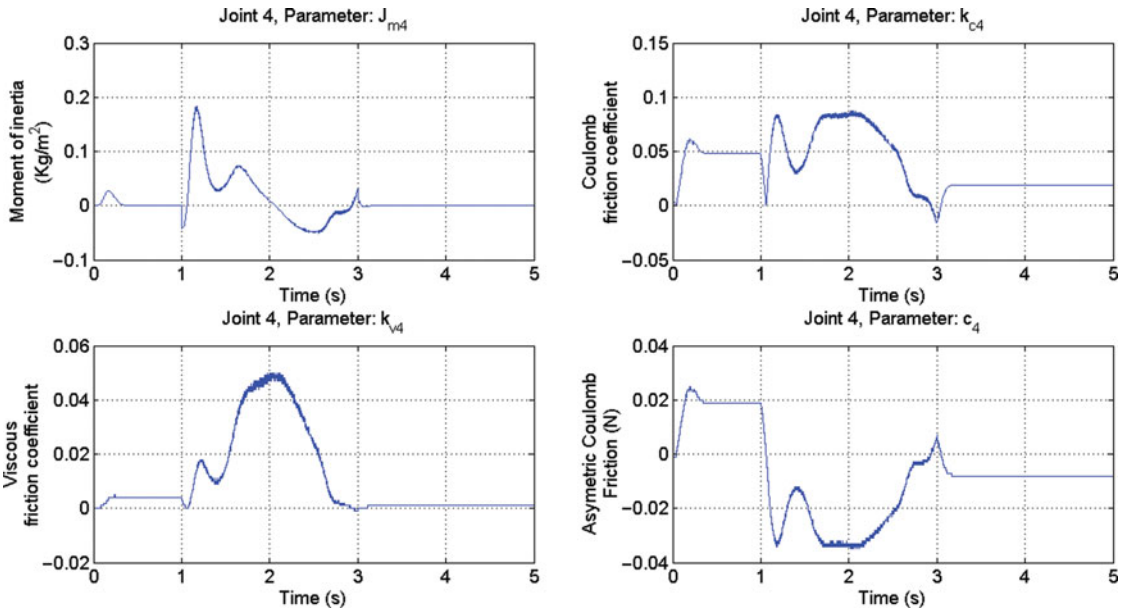


Fig. 15. Parameter adaptation of joint 4.

the uncertainty caused by the system variation will be considerably reduced by the VDC controller approach.

4.6. Safety

It is very important to mention that the capacity to deal with the characteristics of different subjects does not mean that regardless of the subjects' capabilities, the robot will force them to complete the trajectory. The control interface has the provision to modify the position and the velocity limits of the joints, in order to adapt to the different conditions and rehabilitation needs of the subjects. The robot also has mechanical stoppers to maintain the maximum allowable range of motion inside the normal range of human activities of daily living, with these limits illustrated in Fig. 1.

5. Conclusions and Future Work

Trajectory tracking exercises, representing passive and active rehabilitation therapy, were implemented using the VDC approach with trajectories given in joint and in Cartesian space with excellent results. As to their application to rehabilitation, the experimental results show that VDC can respond to subject variations and can effectively accomplish rehabilitation therapy. A performance evaluation against CTC and PID, with all controllers using fixed gains, successfully demonstrated the advantage of the VDC. In experiments, the tracking performance of the robot was compared for different controllers, while varying the subjects' physical characteristics or the movement velocity. The results indicate that the robot under VDC control can better adapt to these variations, as compared to PID and CTC. Thus, we conclude that with this VDC's robustness, a more systematic rehabilitation can be accomplished. The VDC controller will reduce the variation of the trajectory tracking and will provide a better measure of the patient's improvement, as the most important source of variation in movement performance will be the changes in their condition.

A future step in the development of this project is to continue the development of active rehabilitation, where the robot helps the user in completing a given movement. This includes active-assistive rehabilitation, in which the robot can help the subject to different degrees, according to specific rehabilitation needs (time, movement boundaries, etc.). More active rehabilitation work has been developed in our lab, such as changing the effort required of the user to move the robot. Still additional active rehabilitation tasks are to be incorporated, using electromyographic (EMG) signals, to provide the system with another source of information regarding the user's intention of movement. These EMG signals can be used as a full rehabilitation scheme or combined with force sensor data

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

in order to offer the adequate rehabilitation therapy according to patients' needs. Haptic capabilities are already under development for the ETS-MARSE.

Acknowledgements
The first author would like to thank the National Council on Science and Technology of Mexico (CONACYT) for the support under grant No. 211951.

References

1. J. L. Pons, *Wearable Robots: Biomechatronic Exoskeletons* (Wiley, Hoboken, N. J., 2008).
2. United Nations, Department of Economic and Social Affairs, Population Division, *World Population Ageing 2013* (United Nations, New York, 2013).
3. J. Mackay and G. A. Mensah, *The Atlas of Heart Disease and Stroke*, (World Health Organization, Geneva, Switzerland, 2004) pp. 50–53.
4. P. Garrec, J. P. Friconneau, Y. Measson and Y. Perrot, "ABLE, an Innovative Transparent Exoskeleton for the Upper-Limb," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2008) pp. 1483–1488.
5. T. Nef, M. Guidali, V. Klamroth-Marganska and R. Riener, "ARMin - Exoskeleton Robot for Stroke Rehabilitation," *World Congress on Medical Physics and Biomedical Engineering, Munich, Germany (September 7–12, 2009),* (O. Dössel and W. Schlegel, eds.) (Springer Berlin Heidelberg, 2009), pp. 127–130.
6. M. H. Rahman, M. Saad, J. P. Kenné, P. S. Archambault and T. K. Ouimet, "Development of a 4DoFs exoskeleton robot for passive arm movement assistance," *Int. J. Mechatronics Autom.* **2**(1), 34–50 (2012).
7. H. M. Van der Loos and D. J. Reinkensmeyer, "Rehabilitation and Health Care Robotics" **In**: *Springer Handbook of Robotics* (B. Siciliano and O. Khatib, eds.), (Springer, Berlin Heidelberg, 2008) pp. 1225–1227.
8. J. C. Perry and J. Rosen, "Design of a 7 Degree-of-Freedom Upper-Limb Powered Exoskeleton," *The First IEEE/RAS-EMBS International Conference on*, *Biomedical Robotics and Biomechatronics, 2006. BioRob 2006.* (2006) pp. 805–810.
9. N. Tsagarakis, D. G. Caldwell and G. A. Medrano-Cerda, "A 7 DOF Pneumatic Muscle Actuator (pMA) Powered Exoskeleton," *8th IEEE International Workshop on*, *Robot and Human Interaction, 1999.* (1999) pp. 327–333.
10. N. Norouzi-Gheidari, P. S. Archambault and J. Fung, "Effects of robot-assisted therapy on stroke rehabilitation in upper limbs: Systematic review and meta-analysis of the literature," *J. Rehabil. Res. Dev.* **49**(4), 479–496 (2012).
11. I. Sarakoglou, S. Kousidou, N. G. Tsagarakis and D. G. Caldwell, "Exoskeleton-based exercisers for the disabilities of the upper arm and hand" **In**: *Rehabilitation Robotics* (S. S. Kommu, ed.) (I-Tech Education and Publishing, 2007) pp. 499–522.
12. H. S. Lo and S. Q. Xie, "Exoskeleton robots for upper-limb rehabilitation: State of the art and future prospects," *Med. Eng. Phys.* **34**(3), 261–268 (2012).
13. J. Eschweiler, K. Gerlach-Hahn, A. Jansen-Toy and S. Leonhardt, "A survey on robotic devices for upper limb rehabilitation," *J. Neuroengineering Rehabil.* **11**(1), 3 (2014).
14. M. H. Rahman, M. J. Rahman, O. L. Cristobal, M. Saad, J. P. Kenné and P. S. Archambault, "Development of a whole arm wearable robotic exoskeleton for rehabilitation and to assist upper limb movements," *Robotica*, Available on CJO 2014 doi:10.1017/S0263574714000034.
15. F. C. Sun, Z. Q. Sun and G. Feng, "An adaptive fuzzy controller based on sliding mode for robot manipulators," *IEEE Trans. Syst. Man Cybern. B***29**(5), 661–667 (1999).
16. I. Uzmay and R. Burkan, "Parameter estimation and upper bounding adaptation in adaptive-robust control approaches for trajectory control of robots," *Robotica* **20**(06), 653–660 (2002).
17. Z. Song, J. Yi, D. Zhao and X. Li, "A computed torque controller for uncertain robotic manipulator systems: Fuzzy approach," *Fuzzy Sets Syst.* **154**(2), 208–226 (2005).
18. M. K. Ciliz, "Adaptive control of robot manipulators with neural network based compensation of frictional uncertainties," *Robotica* **23**(02), 159–167 (2005).
19. Z. Li, S. S. Ge, M. Adams and W. S. Wijesoma, "Robust adaptive control of uncertain force/motion constrained nonholonomic mobile manipulators," *Automatica* **44**(3), 776–784 (2008).
20. L. Zhijun, S. S. Ge, M. Adams and W. S. Wijesoma, "Adaptive robust output-feedback motion/force control of electrically driven nonholonomic mobile manipulators," *IEEE Trans. Control Syst. Technol.* **16**(6), 1308–1315 (2008).
21. E. A. Brackbill, M. Ying, S. K. Agrawal, M. Annapragada and V. N. Dubey, "Dynamics and Control of a 4-Dof Wearable Cable-Driven upper Arm Exoskeleton," *IEEE International Conference on*, *Rob. and Autom., 2009.* (2009) pp. 2300–2305.
22. M. H. Rahman, K. Kiguchi, M. M. Rahman and M. Sasaki, "Robotic Exoskeleton for Rehabilitation and Motion Assist," *First International Conference on Industrial and Information Systems,* (2006) pp. 241–246.
23. W.-H. Zhu, J.-C. Piedboeuf and Y. Gonthier, "A dynamics formulation of general constrained robots," *Multibody Syst. Dyn.* **16**(1), 37–54 (2006).

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

24. W. H. Zhu and J. De Schutter, "Experimental verifications of virtual-decomposition-based motion/force control," *IEEE Trans. Robot. Autom.* **18**(3), 379–386 (2002).
25. M. H. Rahman, M. Saad, J. Kenné and P. S. Archambault, "Nonlinear Sliding Mode Control Implementation of an Upper Limb Exoskeleton Robot to Provide Passive Rehabilitation Therapy," In: *Intelligent Robotics and Applications* (C.-Y. Su, S. Rakheja and H. Liu, eds.) (Springer Berlin Heidelberg, 2012) pp. 52–62.
26. J. J. Craig, *Introduction to Robotics : Mechanics and Control* (Pearson/Prentice Hall, Upper Saddle River, N.J., 2005).
27. W. H. Zhu, X. Yu-Geng, Z.-J. Zhang, B. Zeungnam and J. De Schutter, "Virtual decomposition based control for generalized high dimensional robotic systems with complicated structure," *IEEE Trans. Robot. Autom.* **13**(3), 411–436 (1997).
28. W. H. Zhu, T. Lamarche, E. Dupuis, D. Jameux, P. Barnard and L. Guangjun, "Precision control of modular robot manipulators: The VDC approach with embedded FPGA," *IEEE Trans. Robot.* **29**(5), 1162–1179 (2013).
29. W. H. Zhu and J. De Schutter, "Adaptive control of mixed rigid/flexible joint robot manipulators based on virtual decomposition," *IEEE Trans. Robot. Autom.* **15**(2), 310–317 (1999).
30. W. H. Zhu and T. Lamarche, "Modular Robot Manipulators Based on Virtual Decomposition Control," *IEEE International Conference on*, *Robot. and Autom.*, (2007) pp. 2235–2240.
31. Brigham and Women's Hospital Rehabilitation Services, (2014). *Physical Therapy Standards of Care and Protocols.* Available: http://www.brighamandwomens.org/Patients_Visitors/pcs/rehabilitationservices/StandardsofCare.aspx [Accessed 2 Sep. 2014].
32. S. B. Ferrer, C. Ochoa-Luna, M. H. Rahman, M. Saad and P. S. Archambault, "HELIOS: The human machine interface for MARSE robot," *The 6th International Conference on*, *Human System Interaction (HSI)*, (2013) pp. 117–122.
33. W. H. Zhu, "Regressor Matrix and Parameter Vector of an Object" **In**: *Virtual Decomposition Control: Toward Hyper Degrees of Freedom Robots.* (Springer, Berlin Heidelberg, 2010) pp. 387–389.

Appendix A[33]

Non-zero elements in the regressor matrix $Y_A \in \mathbb{R}^{6\times 13}$ for the link linear parametrization in (5):

$$y_A(1,1) = \frac{d}{dt}\left({}^A v_r\right)(1) + {}^A v(5)\,{}^A v_r(3) - {}^A v(6)\,{}^A v_r(2) + {}^A g(1) \tag{A.1}$$

$$y_A(1,2) = -{}^A v(5)\,{}^A v_r(5) - {}^A v(6)\,{}^A v_r(6) \tag{A.2}$$

$$y_A(1,3) = -\frac{d}{dt}\left({}^A v_r\right)(6) + {}^A v(5)\,{}^A v_r(4) \tag{A.3}$$

$$y_A(1,4) = \frac{d}{dt}\left({}^A v_r\right)(5) + {}^A v(6)\,{}^A v_r(4) \tag{A.4}$$

$$y_A(2,1) = \frac{d}{dt}\left({}^A v_r\right)(2) + {}^A v(6)\,{}^A v_r(1) - {}^A v(4)\,{}^A v_r(3) + {}^A g(2) \tag{A.5}$$

$$y_A(2,2) = \frac{d}{dt}\left({}^A v_r\right)(6) + {}^A v(4)\,{}^A v_r(5) \tag{A.6}$$

$$y_A(2,3) = -{}^A v(4)\,{}^A v_r(4) + {}^A v(6)\,{}^A v_r(5) \tag{A.7}$$

$$y_A(2,4) = -\frac{d}{dt}\left({}^A v_r\right)(4) + {}^A v(6)\,{}^A v_r(5) \tag{A.8}$$

$$y_A(3,1) = \frac{d}{dt}\left({}^A v_r\right)(3) + {}^A v(4)\,{}^A v_r(2) - {}^A v(5)\,{}^A v_r(1) + {}^A g(3) \tag{A.9}$$

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

$$y_A(3,2) = -\frac{d}{dt}\left({}^A v_r\right)(5) + {}^A v(4)\,{}^A v_r(6) \tag{A.10}$$

$$y_A(3,3) = \frac{d}{dt}\left({}^A v_r\right)(4) + {}^A v(5)\,{}^A v_r(6) \tag{A.11}$$

$$y_A(3,4) = -{}^A v(4)\,{}^A v_r(4) - {}^A v(5)\,{}^A v_r(5) \tag{A.12}$$

$$y_A(4,3) = y_A(3,1) \tag{A.13}$$

$$y_A(4,4) = -y_A(2,1) \tag{A.14}$$

$$y_A(4,6) = y_A(3,3) \tag{A.15}$$

$$y_A(4,7) = -y_A(2,4) \tag{A.16}$$

$$y_A(4,8) = y_A(3,2) \tag{A.17}$$

$$y_A(4,9) = -y_A(2,2) \tag{A.18}$$

$$y_A(4,10) = {}^A v(6)\,{}^A v_r(6) - {}^A v(5)\,{}^A v_r(5) \tag{A.19}$$

$$y_A(4,11) = \frac{d}{dt}\left({}^A v_r\right)(4) + {}^A v(5)\,{}^A v_r(6) - {}^A v(6)\,{}^A v_r(5) \tag{A.20}$$

$$y_A(4,12) = -{}^A v(6)\,{}^A v_r(5) \tag{A.21}$$

$$y_A(4,13) = {}^A v(5)\,{}^A v_r(6) \tag{A.22}$$

$$y_A(5,2) = -y_A(3,1) \tag{A.23}$$

$$y_A(5,4) = y_A(1,1) \tag{A.24}$$

$$y_A(5,5) = -y_A(3,2) \tag{A.25}$$

$$y_A(5,7) = y_A(1,4) \tag{A.26}$$

$$y_A(5,8) = -y_A(3,3) \tag{A.27}$$

$$y_A(5,9) = {}^A v(4)\,{}^A v_r(4) - {}^A v(6)\,{}^A v_r(6) \tag{A.28}$$

$$y_A(5,10) = y_A(1,3) \tag{A.29}$$

$$y_A(5,11) = {}^A v(6)\,{}^A v_r(4) \tag{A.30}$$

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

$$y_A(5,12) = \frac{d}{dt}\left({}^A v_r\right)(5) + {}^A v(6)\,{}^A v_r(4) - {}^A v(4)\,{}^A v_r(6) \tag{A.31}$$

$$y_A(5,13) = -{}^A v(4)\,{}^A v_r(6) \tag{A.32}$$

$$y_A(6,2) = y_A(2,1) \tag{A.33}$$

$$y_A(6,3) = -y_A(1,1) \tag{A.34}$$

$$y_A(6,5) = y_A(2,2) \tag{A.35}$$

$$y_A(6,6) = -y_A(1,3) \tag{A.36}$$

$$y_A(6,8) = {}^A v(5)\,{}^A v_r(5) - {}^A v(4)\,{}^A v_r(4) \tag{A.37}$$

$$y_A(6,9) = y_A(2,4) \tag{A.38}$$

$$y_A(6,10) = -y_A(1,4) \tag{A.39}$$

$$y_A(6,11) = -{}^A v(5)\,{}^A v_r(4) \tag{A.40}$$

$$y_A(6,12) = {}^A v(4)\,{}^A v_r(5) \tag{A.41}$$

$$y_A(6,13) = \frac{d}{dt}\left({}^A v_r\right)(6) + {}^A v(4)\,{}^A v_r(5) - {}^A v(5)\,{}^A v_r(4) \tag{A.42}$$

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

$$\theta_{A1} = m_A \tag{A.43}$$

$$\theta_{A2} = m_A {}^A r_{mx} \tag{A.44}$$

$$\theta_{A3} = m_A {}^A r_{my} \tag{A.45}$$

$$\theta_{A4} = m_A {}^A r_{mz} \tag{A.46}$$

$$\theta_{A5} = m_A {}^A r_{mx}^2 \tag{A.47}$$

$$\theta_{A6} = m_A {}^A r_{my}^2 \tag{A.48}$$

$$\theta_{A7} = m_A {}^A r_{mz}^2 \tag{A.49}$$

$$\theta_{A8} = m_A {}^A r_{mx} {}^A r_{my} - I_{Axy} \tag{A.50}$$

$$\theta_{A9} = m_A {}^A r_{mx} {}^A r_{mz} - I_{Axz} \tag{A.51}$$

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press

$$\theta_{A10} = m_A {}^A r_{my} {}^A r_{mz} - I_{Ayz} \tag{A.52}$$

$$\theta_{A11} = I_{Axx} \tag{A.53}$$

$$\theta_{A12} = I_{Ayy} \tag{A.54}$$

$$\theta_{A13} = I_{Azz} \tag{A.55}$$

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.

https://doi.org/10.1017/S026357471400246X Published online by Cambridge University Press