Computational models of upper-limb motion during functional reaching tasks for application in FES-based stroke rehabilitation

Abstract: Functional electrical stimulation (FES) has been shown to be an effective approach to upper-limb stroke rehabilitation, where it is used to assist arm and shoulder motion. Model-based FES controllers have recently confirmed significant potential to improve accuracy of functional reaching tasks, but they typically require a reference trajectory to track. Few upper-limb FES control schemes embed a computational model of the task; however, this is critical to ensure the controller reinforces the intended movement with high accuracy. This paper derives computational motor control models of functional tasks that can be directly embedded in real-time FES control schemes, removing the need for a predefined reference trajectory. Dynamic models of the electrically stimulated arm are first derived, and constrained optimisation problems are formulated to encapsulate common activities of daily living. These are solved using iterative algorithms, and results are compared with kinematic data from 12 subjects and found to fit closely (mean fitting between 63.2% and 84.0%). The optimisation is performed iteratively using kinematic variables and hence can be transformed into an iterative learning control algorithm by replacing simulation signals with experimental data. The approach is therefore capable of controlling FES in real time to assist tasks in a manner corresponding to unimpaired natural movement. By ensuring that assistance is aligned with voluntary intention, the controller hence maximises the potential effectiveness of future stroke rehabilitation trials.

Keywords: electrical stimulation; iterative learning control; modelling and simulation; rehabilitation engineering.

Introduction

Stroke is a leading cause of adult disability in the EU, and over 6 million stroke patients require care. Of these, two-thirds have impairment of their affected arm 4 years post-stroke, resulting in an annual cost of 38 billion [33]. Approximately 70% of patients will experience altered arm function after a stroke, and about 40% of survivors will be left with a non-functional arm that is weak and often spastic [24]. This motor deficit limits functional arm use in daily life, but also engagement in the community [50]. Residual impairments and functional deficit in a large percentage of stroke patients indicate that current rehabilitation provision is unsatisfactory [45].

In recent years, there has been growing evidence supporting the effectiveness of rehabilitation robots [30] and functional electrical stimulation (FES) [26] to reduce impairment post-stroke. Both technologies enable a person with limited physical ability to practice tasks, and the resulting sensory feedback is associated with cortical changes that can bring about recovery of functional movement. In particular, there is substantial clinical evidence [8–10, 26] indicating that increased functional recovery is closely related to the accuracy with which FES assists the subject’s own voluntary completion of a task. This finding also has theoretical support from neurophysiology [4, 39] and motor learning research [40].

Model-based FES control is critical to reduce the effect of noise/disturbance, increase accuracy, and enable complex functional tasks to be performed. However, inherent environmental constraints mean that few model-based controllers have transferred to clinical practice [54], where typically open-loop or triggered controllers are employed and resulting performance is limited. This is despite a wide variety of FES upper-limb control techniques having been applied in simulation or laboratory conditions.
 Artificial neural networks (ANNs) have been extensively used for upper-limb FES control, especially within neuroprostheses for subjects with spinal cord injury. Here ANNs have been used to represent the inverse dynamics of single-joint, single-muscle systems by creating a mapping from joint kinematics to the required muscle activation. In the case of complex shoulder and arm movements, the inverse dynamics are not unique, and hence sophisticated musculoskeletal models have been developed, such as [5, 11, 47], and combined with optimal cost functions to distribute forces between redundant muscles [3, 18]. ANNs are then trained to approximate the inverse arm dynamics and then used to compute muscle activation levels in real time. This structure is typically combined with an appropriate FES feedback controller to ensure the muscle activation is achieved.

Iterative learning control (ILC) is another model-based approach that has been employed clinically. It has been used in upper-limb stroke rehabilitation [17], to assist movement in the lower limb [1, 2, 32, 41], as well as in the wider biomedical field for drug delivery and heart rate monitoring. When used in FES-based stroke rehabilitation, ILC exploits the repetitive nature of the rehabilitation process where patients attempt the same task multiple times in order to promote re-learning of motor skills. ILC precisely assists the patient’s movement in each attempt of the task and sequentially improves accuracy using data from previous attempts to adjust the FES supplied during the next attempt at the task. By exploiting experience in this way, ILC is able to embed both accuracy and robustness to model uncertainty [17].

ILC update algorithms work by directly using functional mappings that approximate the inverse system dynamics. ILC hence operates in a similar fashion to the aforementioned model-based ANN approaches but implements this mapping directly without employing ANNs. This potentially restricts the available model structure, but has the advantage of not requiring any training procedure, exploiting learning from experience to improve performance, enabling strict control over the amount of FES supplied during each attempt in order to maximise the patient’s volitional effort, and giving rise to transparent analysis of performance, convergence, and robustness to model uncertainty. ILC has yielded a high level of clinical performance in multiple clinical trials [23, 28, 29], most recently assisting functional activities such as closing a drawer, switching on a light switch, stabilising an object, button pressing, and repositioning an object. The setup used in a recent clinical treatment trial is shown in Figure 1.

A drawback to previous use of ILC is that it requires joint reference trajectories to follow, with these provided from unimpaired subject data. However, since optimisation-based ILC works by iteratively minimising a cost function using experimental data, there is clear scope for it to automatically generate such trajectories if a suitable model of unimpaired movement is embedded within it. This expansion in the scope of ILC has been facilitated by recent extensions to the ILC framework, such as [14, 15, 38], in which more general classes of constrained optimisation problem are addressed. There is hence potential for ILC to precisely support natural motion if models of human movement can be expressed as suitable optimisation problems.

Many studies have been reported characterising human motion during upper-limb reaching tasks. The majority extract relationships between key variables (e.g., timing and amplitude of kinematic, kinetic or electromyographic data) in order to examine the effect of task conditions and/or participant groups on task execution. These include effect of age [55], task conditions [48], and compensatory strategies post-stroke [52]. For example [7, 42, 44], find significant inter-patient differences in the performance of reaching tasks following stroke compared with unimpaired participants. Human sensorimotor control has also been expressed computationally to more fully capture the underlying dynamics of movement [22, 51]. Approaches can be divided into those that attempt to simulate the internal feedback/feedforward mechanisms present in the central nervous system, and those that try only to model the resulting kinematic motion at the task level. The latter have traditionally posed reaching tasks as optimisation problems, involving, for example, the minimisation of jerk [13], torque change [46], variance [19], interaction torques, or combinations of these [34]. The form of an optimisation holds the possibility of application within FES-based control strategies such as ILC, however, the focus has so far overwhelmingly been on planar point to point tasks. The case of functional tasks necessary to complete activities of daily living has not yet been addressed. The first step in modelling functional tasks is the selection of an appropriate underlying musculoskeletal model.

Musculoskeletal models linking FES and resultant motion have been employed in the design of FES control systems by several authors [12, 18, 21, 25]. However, the majority are purely employed for offline analysis of control performance and behaviour. An exception is the simulation environment of Chadwick et al. [5], which is designed for non-invasive real-time simulation and user feedback. One musculoskeletal model that has been embedded within a real-time upper-limb FES control scheme is that of Freeman et al. [16], which was used within clinical
trials with stroke participants. This form has been shown to accurately capture the dynamics of movement in stroke while also yielding computationally tractable solutions.

This paper extends the upper-limb model of Freeman et al. [16] to incorporate 7-degree-of-freedom (dof) shoulder and elbow motion. It then develops a framework in which functional tasks are represented as optimal objective functions, together with additional constraints acting upon variables within the model. The resulting optimisation can be solved using iterative optimisation approaches to yield computational models of functional movement, together with corresponding FES signals. Moreover, it can also be solved using experimental data within the ILC update structure, thereby enabling high-precision functional task completion to maximise effectiveness of stroke rehabilitation.

In addition to enabling more accurate assistance, the development of computational models of natural movement also provides measures of stroke patients’ movement that correspond closely to functional ability. This paper thereby addresses the significant gap, highlighted in [42], that exists between performance measures based on kinematic and kinetic data (generated by, e.g., robotic therapy systems), and clinical measures based on functional ability (e.g., Fugl-Meyer, action research arm test).

Methods

Participants

Following University of Southampton, Faculty of Health Sciences ethical approval, 12 unimpaired volunteers were recruited to the study. Inclusion criteria were that the participants had to be 30–80 years old, able to comply with study protocol, able to communicate effectively, and able to provide written informed consent. Exclusion criteria were the requirement of an interpreter, uncorrected visual impairment, a skin disease or allergy to sticky tape, severe pain in the arm, shoulder, or hand, and, for the control participants, a neurological condition that affects movement in the arm. Participants were recruited and tested over a 2-month period.

The unimpaired participants (six men and six women) were aged between 49 and 77 years (mean 64, SD 10). All unimpaired
participants, except for one, were right-handed. The side tested for each participant was randomised; six participants completed the tasks using their right hand, and the other six using their left hand. Five participants were tested using their dominant hand. The testing side for unimpaired participants was randomised so as to be more representative of the stroke population, which are almost equally distributed between left- and right-sided incidence [20].

Experimental protocol

All participants attended one testing session lasting 2–3 h in which the kinematic movements of the upper limb and hand during three functional reaching tasks were recorded. The tasks were closing a drawer, turning off a light switch, and picking up a can to drink from. These tasks were selected as they offered ecologically valid reach to grasp tasks of varying demands and were similar to tasks used in many studies investigating stroke patients’ movements [6, 43]. Participants performed tasks both freely and when supported by an exoskeleton unweighing system (ArmeoSpring, Hocoma, Zurich, Switzerland); however, only data collection and results for the former case are presented in this paper.

Position data were recorded using a Vicon MX T-Series motion capture system (Vicon, Oxford, UK) using 12 cameras (6× T40 and 6× T160) sampling at 100 Hz. Reflective anatomical markers were positioned on key landmarks of the torso, shoulder complex, upper limb, wrist, and hand, as shown in Figure 2. Marker clusters were attached to the sternum and acromion on the side that was being tested, with additional markers placed on the radial and ulnar styloids and the second and fifth carpometacarpal (CMC) and metacarpophalangeal (MCP) joints of the hand. A marker wand was used to locate specific anatomical landmarks with respect to the relevant marker cluster (see [69]). These additional bony landmarks consisted of the sternal notch, xyphoid process, C7 vertebra, T8 vertebra, sternoclavicular joint, acromioclavicular joint, acromion angle of the scapula, medial spine of the scapula, inferior angle of the scapula, and medial and lateral epicondyles of the elbow. The elbow joint centre was estimated as the midpoint between the medial and lateral epicondyles. The glenohumeral joint centre was defined according to the regression method of Meskers et al. [31]. Additional markers were positioned on the task objects to aid movement identification. Four markers were positioned in a square on the front edge of the drawer and light switch. Markers were located so that they did not inhibit participants’ movements during the tasks.

Participants were seated at a table that was adjusted so that the underside was 10 cm above their knee. For each task, the participant was asked to start with their hand (palm down) on their knee. On a “start” command, the participant completed the task and then placed their hand back on their knee. Tasks were performed at both self-selected and maximal speeds. Five successful trials were collected for each task and each speed; trials were repeated if any of the reflective markers were occluded for more than 25 epochs during the trial. Participants were given a 15- to 30-s break between each trial. All participants completed the drawer closing task first. Maximum reach for each participant was measured from the anterior edge of the acromion (AA) to the end of the index finger.

Drawer closing task A custom-made cabinet that had a drawer with a large, round, central knob was placed on the table in front of the participant so that the drawer knob was directly in line with the participant’s shoulder for the side being tested. The cabinet was placed at a distance corresponding to 100% of the participant’s maximum reach, with the drawer knob directly in line with the participant’s shoulder for the side being tested. When opened, the drawer knob was at a distance of 75% of the participant’s maximum reach. Participants were asked to move their hand from their knee to push the drawer closed using the knob and to return their hand back to their knee.

Light switch task On the opposite side of the cabinet, a standard light switch had been mounted. The cabinet was positioned so that the light switch was directly in line with the participant’s shoulder, at 75% of their maximum reach. Participants were asked to move their hand from their knee to turn off the light switch and to return their hand back to their knee. Participants were always required to push in the top of the light switch, which required more control and stability of the arm than pressing the bottom of the switch.

Data analysis

Kinematic data, collected using the Vicon system, were reconstructed using Vicon Nexus (Version 1.8) software to provide three-dimensional position data for all the markers. Marker position data were filtered using a fourth-order low-pass Butterworth filter, with a typical cutoff frequency of 10 Hz. Anatomical landmarks, recreated from their known positions with respect to the marker clusters, were used to define local coordinate systems for the thorax, scapular, humerus, and ulna following ISB guidelines [33]. The hand was defined with an origin at the midpoint between the second and fifth CMC markers, the Y axis was defined as being parallel with the line formed by the midpoint of the CMC markers to the midpoint of the MCP markers pointing dorsally, the Z axis parallel to the line from the second CMC marker to the fifth CMC marker pointing laterally (with the hand supinated), with the X axis orthogonal to the Y and Z axes.
Positional data were averaged at each time point across the five repetitions, for each participant, task, and speed. Key timings were then extracted, comprising the start and end of the movement, defined by the initial hand movement from the participant’s knee and the hand returning to the knee, respectively. For the light task, an additional timing was when the light switch was pressed. For the drawer task, two additional timings were at the start and end of the drawer movement.

Computational model development

A simplified biomechanical model of the electrically stimulated human arm is shown in Figure 3 and is an extension of the planar model [16] validated with both unimpaired and stroke subjects. Parameters \( l_i \) and \( a_i \) denote link and centre of mass lengths, respectively, for \( i \in \{1, 2, 3\} \), and \( m_i \) and \( I_i \) are mass and inertias, respectively. Joint angles \( \theta_i \), \( i \in \{1, 2, 3\} \), relate to a 3-dof shoulder joint, \( \theta_s \) to a 1-dof elbow joint, \( \theta_e \) to forearm rotation, and \( \theta_f \), \( i \in \{6, 7\} \), corresponds to wrist flexion/extension, abduction/adduction. This rigid body representation has been shown to adequately capture the dynamic properties of the upper limb, while giving rise to optimisations that are computationally tractable (a key issue when subsequently used in real-time control) [16]. Note that the experimental results reported in this paper omit joint angles here for completeness.

The dynamics are represented by

\[
\mathbf{B}(\dot{\theta}(t), \ddot{\theta}(t)) = \mathbf{M}(\theta(t)) \ddot{\theta}(t) + \mathbf{C}(\theta(t), \dot{\theta}(t)) \dot{\theta}(t) + \mathbf{F}(\dot{\theta}(t)) + \mathbf{G}(\theta(t))
\]

where the joint angle vector \( \dot{\theta} = [\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5, \dot{\theta}_6, \dot{\theta}_7]^T \) and \( \mathbf{B}() \) and \( \mathbf{C}() \) are the 7x7 inertial and Coriolis matrices, respectively. In addition, \( \mathbf{F}() \) is the system Jacobian, \( \mathbf{h}(\cdot) \) is the externally applied force/torque, and \( \mathbf{F}(\cdot) \) and \( \mathbf{G}(\cdot) \) are friction and gravitational vectors. The general form \( \mathbf{F}(\dot{\theta}(t)) = \mathbf{F}(\theta(t), \dot{\theta}(t)) \) is used and identification procedures for the terms in Eq. 1 can be found in [16]. It is assumed that \( n \) muscles are involved in the movement, with elements of vector \( \mathbf{u}(t) \) specifying the innervation input applied to each (either through voluntary action or application of electrical stimulation). The \( i \)th element of the muscle torque vector \( \tau_i \) is the sum of moments generated by each of these muscles, which each may impart a moment about the \( i \)th joint. From [27], it is assumed that each can be represented as a Hill-type model of the form

\[
\tau_i(u(t), \theta_i(t), \dot{\theta}_i(t)) = h_i(u(t), t) \times F_{m,i}(\theta_i(t), \dot{\theta}_i(t)),
\]

where \( h_i(u(t), t) \) is a Hammerstein structure incorporating a static non-linearity, \( h_{IRC}(u(t)) \), representing the isometric recruitment curve, cascaded with linear activation dynamics, \( h_{IRC}(\cdot) \). Here \( h_{IRC}(\cdot) = L[h_{IRC}(\cdot)](s) \), where \( L(\cdot) \) denotes the Laplace transform. The term \( F_{m,i}(\theta(t), \dot{\theta}(t)) \) models the multiplicative effect of the joint angle and joint angular velocity on the active torque developed by the muscle. It is assumed for simplicity that \( h_{m,i}(\cdot) = h_{IRC}(\cdot) \), and \( h_{IRC}(\cdot) = h_{IRC}(\cdot) \), so that \( h_i(u(t), s) \). This leads to the structure

\[
\tau_i(u(t), \theta_i(t), \dot{\theta}_i(t)) = \sum_{j=1}^{n} f_{m,i}(u_j(t), \theta_j(t), \dot{\theta}_j(t)) = \sum_{j=1}^{n} \{ h_j(u(t), t) \times F_{m,i}(\theta_j(t), \dot{\theta}_j(t)) \}
\]

Now let \( h_{IRC}(\cdot), \mathbf{h}(\cdot), \mathbf{F}(\cdot), \mathbf{G}(\cdot) \) have continuous time state-space model matrices \( \{A_{IR}, B_{IR}, C_{IR}, D_{IR}\} \) (state, input, and output, respectively), and states \( x_{IRC}(\cdot) \), so that

![Figure 3](https://example.com/figure3.png)

Figure 3 (A) Unimpaired participant performing task (with exoskeletal support) and (B) 7 dof model of the upper limb.
\[
I(\mathbf{u}) = \| \mathbf{u} \|^2, \quad \text{such that } g_\mathbf{u}(\mathbf{x}, \mathbf{u}) = \mathbf{p}, \quad g_\mathbf{v}(\mathbf{x}, \mathbf{u}) = 0, \quad u,. + g_\mathbf{u}(\mathbf{x}, \mathbf{u}) \lambda, + g_\mathbf{v}(\mathbf{x}, \mathbf{u}) \lambda, = 0,
\]
subject to the dynamics (5) over \( t \in [0, T] \). Here \( y = \mathbf{k}(t) \) is the direct kinematics equation of the system, with \( y \) the hand position in Cartesian space and \( \mathbf{p} \) is the light switch position in Cartesian space. Problem 7 can be solved through iterative optimisation methods, which are based on linear approximations of the system dynamics [35]. Given an operating point \((\mathbf{x}, \mathbf{u})\), define
\[
A(t) = \left( \frac{d}{dx} f(x(t)) \right)_{x(t)}, \quad B(t) = \left( \frac{d}{du} g(u(t)) \right)_{x(t)},
\]
\[
C(t) = \left( \frac{d}{dx} k(h(x)) \right)_{x(t)} = [I_1(\mathbf{u}) \quad 0 \quad \ldots \quad 0], \quad t \in [0, T],
\]
then the relationship between position, \( y(t) \), and stimulation, \( u(t) \), signals over \( t \in [0, T] \) can be approximated by the linear time-varying system
\[
y(t) = g_\mathbf{u}(\mathbf{x}, \mathbf{u}) : z(t) = A(t)z(t) + B(t)u(t), \quad y(t) = C(t)z(t), \quad t \in [0, T].
\]
Likewise, the relationship between velocity, \( \dot{y}(t) \), and stimulation, \( u(t) \), can be approximated by
\[
\dot{y}(t) = g_\mathbf{u}(\mathbf{x}, \mathbf{u}) : \dot{z}(t) = A(t)z(t) + B(t)u(t), \quad \dot{y}(t) = C(t)A(t)z(t), \quad t \in [0, T].
\]
The linear operators \( g_\mathbf{u}(\mathbf{x}, \mathbf{u}) : L^2([0, T]) \rightarrow \mathbb{R}^n : u \rightarrow y(T) \) and \( g_\mathbf{v}(\mathbf{x}, \mathbf{u}) : L^2([0, T]) \rightarrow \mathbb{R}^n : u \rightarrow y(T) \) in Eqs. 10 and 11 are defined as
\[
g_\mathbf{u}(\mathbf{x}, \mathbf{u}) : \int_0^T C(T)\Phi(T, r)B(r)u(r)dr,
g_\mathbf{v}(\mathbf{x}, \mathbf{u}) : \int_0^T C(T)A(t)\Phi(T, r)B(r)u(r)dr,
\]
where \( \Phi(t, r) \) is the state transition matrix for systems 10 and 11. Around operating point \((\overline{\mathbf{u}}, \overline{\mathbf{x}})\), the optimisation problem (7) can therefore be replaced by the linear counterpart
\[
\min_u \| \mathbf{u} \|^2, \quad \text{such that } g_\mathbf{u}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) = \mathbf{p}, \quad g_\mathbf{v}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) = 0,
\]
with multipliers \( \lambda, \lambda, \). Here the inner product is defined by
\[
\langle \mathbf{u}, \mathbf{u} \rangle = \int_0^T u_i^T(t)u_j(t)dt, \quad \text{with } u_i, u_j \rightarrow \| u_i \|^2.
\]
It follows that the relevant conditions for a stationary point \((\overline{\mathbf{u}}, \overline{\mathbf{v}}, \lambda, \lambda, )\) are
\[
g_\mathbf{u}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) = 0, \quad g_\mathbf{v}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) = 0,
\]
where the adjoint operators \( g_\mathbf{u}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) : \mathbb{R}^n \rightarrow L^2([0, T]) \) and \( g_\mathbf{v}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) : \mathbb{R}^n \rightarrow L^2([0, T]) \) are computed as
\[
w = g_\mathbf{u}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) : v(t) = A^\dagger(t)z(t)
\]
\[
w(t) = B^\dagger(t)z(t), \quad z(t) = C(t)v, \quad t \in [0, T]
\]
and
\[
w = g_\mathbf{v}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) : v(t) = A^\dagger(t)x(t)
\]
\[
w(t) = B^\dagger(t)x(t), \quad x(t) = A(t)C(t)v, \quad t \in [0, T],
\]
respectively. From Eq. 16, the solution to Eq. 13 is given by
\[
\mathbf{u}_n = g^{\mathbf{u}^*}_\mathbf{u}(\overline{\mathbf{x}}, \overline{\mathbf{u}})g^{\mathbf{v}^*}_\mathbf{u}(\overline{\mathbf{x}}, \overline{\mathbf{u}})\mathbf{v},
\]
where the compound operators are defined by
\[
g^{\mathbf{u}^*}_\mathbf{v}(\overline{\mathbf{x}}, \overline{\mathbf{u}}) : \mathbf{v} = g^{(\mathbf{u}^*)\mathbf{v}^*_\mathbf{u}}(\overline{\mathbf{x}}, \overline{\mathbf{u}})\mathbf{v},
\]
To solve the original problem (7), the solution (19) is employed in an iterative update procedure that corresponds to an implementation of the Newton method approach. This necessitates modifying Eq. 19 to take the form
\[
\mathbf{u}_{n+1} = \mathbf{u}_n + \alpha g^{\mathbf{u}^*}_\mathbf{u}(\mathbf{x}_n, \mathbf{u}_n)g^{\mathbf{v}^*}_\mathbf{u}(\mathbf{x}_n, \mathbf{u}_n)\mathbf{v},
\]
where \( \alpha \in [0, 1] \) is a scalar gain chosen to affect a compromise between robustness and convergence speed. The final update procedure is given in Table 1. In this paper, the model (5) is used in simulation within steps (c) and (d) to model voluntary task completion. However, the procedure in Table 1 can equally be employed experimentally to control FES applied to the real human arm by redefining \( \mathbf{u}_n \) to be...
the FES input vector on trial $k+1$, with element $u_{k+1,i}(t)$ the stimulation signal applied to the $i^{th}$ muscle at time $t$. This requires replacing the associated components of the muscle model (3) by identified parameters that capture the dynamics of applied FES [27] and replacing the simulation outputs $y_k$, $\dot{y}_k$ by their experimentally recorded alternatives within step (d). Note that the procedure of Table 1 is similar to ILC approaches in [14, 15, 36–38] and inherits many of their favourable robustness and convergence properties. The problem definition for the drawer closing task is given in Appendix A.1, and extensions to include the return component of both tasks are given in Appendix A.2 and A.3.

**Results**

To examine how accurately the motor control models developed represent the experimentally collected unipaired movement, it is necessary to perform the modelling procedure set out in Table 1. The first step is to extract parameters appearing within the arm dynamic model (5) and task descriptions (7) and (22). The former uses estimated arm lengths, masses, and inertias, with non-conservative forces taking the simple decoupled form

$$F_i(t) = b_i \dot{\theta}_i(t) + k_i (\dot{\theta}_i(t) - \dot{\theta}^s_i),$$

where $b_i$, $k_i$, and set-point $\dot{\theta}^s_i$ are scalars for the joint angles $i=1, \ldots, 4$ employed in the model.

**Table 1** Update procedure to solve light switch task.

| a. Determine task parameters $T$, $p$, and parameters within model (5). |
| b. Set $k=0$ and choose an initial input $u_0=0$. |
| c. Apply input $u_k$ to the system. |
| d. Record the outputs $y_k(T)$, $\dot{y}_k(T)$, and states $x_k$. |
| e. Stop if, for sufficiently small positive scalars $\delta$, $\epsilon$, |
| $||u_k - u_{k-1}|| < \delta$, and $||y_k(T) - y_{k-1}(T)|| < \epsilon$. |
| f. Linearise the system about $(u_k, x_k) = (u_0, x_0)$ using Eqs. 9–11, 17, and 18. |
| g. Calculate update $u_{k+1}$ using Eq. 21. |
| h. Increment $k$ and go to step (c). |

**Table 2** Light switch pressing task fitting error.

<table>
<thead>
<tr>
<th>Maximal speed</th>
<th>Reach only</th>
<th>Reach</th>
<th>Return</th>
<th>Reach and return</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td></td>
<td>\hat{\theta} - \theta</td>
<td></td>
<td>$</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>\hat{\theta} - \theta</td>
<td></td>
<td>$</td>
</tr>
<tr>
<td>P1</td>
<td>80.443</td>
<td>76.167</td>
<td>77.880</td>
<td>76.557</td>
</tr>
<tr>
<td>P2</td>
<td>74.847</td>
<td>69.629</td>
<td>70.820</td>
<td>69.164</td>
</tr>
<tr>
<td>P3</td>
<td>66.837</td>
<td>59.148</td>
<td>66.722</td>
<td>61.833</td>
</tr>
<tr>
<td>P4</td>
<td>74.538</td>
<td>68.472</td>
<td>86.136</td>
<td>74.953</td>
</tr>
<tr>
<td>P5</td>
<td>70.221</td>
<td>61.606</td>
<td>73.449</td>
<td>66.826</td>
</tr>
<tr>
<td>P6</td>
<td>69.278</td>
<td>60.013</td>
<td>60.456</td>
<td>76.493</td>
</tr>
<tr>
<td>P7</td>
<td>75.090</td>
<td>72.054</td>
<td>77.303</td>
<td>76.054</td>
</tr>
<tr>
<td>P8</td>
<td>78.587</td>
<td>72.880</td>
<td>77.303</td>
<td>76.054</td>
</tr>
<tr>
<td>P9</td>
<td>73.831</td>
<td>63.517</td>
<td>60.228</td>
<td>60.975</td>
</tr>
<tr>
<td>P10</td>
<td>78.050</td>
<td>75.057</td>
<td>68.387</td>
<td>71.068</td>
</tr>
<tr>
<td>P11</td>
<td>83.692</td>
<td>81.552</td>
<td>69.366</td>
<td>73.323</td>
</tr>
<tr>
<td>P12</td>
<td>85.286</td>
<td>70.227</td>
<td>67.530</td>
<td>68.537</td>
</tr>
<tr>
<td>Mean</td>
<td>75.892</td>
<td>69.194</td>
<td>71.580</td>
<td>69.225</td>
</tr>
<tr>
<td>(SD)</td>
<td>5.620</td>
<td>6.969</td>
<td>7.857</td>
<td>6.407</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Self-selected speed</th>
<th>Reach only</th>
<th>Reach</th>
<th>Return</th>
<th>Reach and return</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td></td>
<td>\hat{\theta} - \theta</td>
<td></td>
<td>$</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>\hat{\theta} - \theta</td>
<td></td>
<td>$</td>
</tr>
<tr>
<td>P1</td>
<td>66.837</td>
<td>59.148</td>
<td>66.722</td>
<td>61.833</td>
</tr>
<tr>
<td>P2</td>
<td>74.538</td>
<td>68.472</td>
<td>86.136</td>
<td>74.953</td>
</tr>
<tr>
<td>P3</td>
<td>70.221</td>
<td>61.606</td>
<td>73.449</td>
<td>66.826</td>
</tr>
<tr>
<td>P4</td>
<td>69.278</td>
<td>60.013</td>
<td>60.456</td>
<td>76.493</td>
</tr>
<tr>
<td>P5</td>
<td>75.090</td>
<td>72.054</td>
<td>77.303</td>
<td>76.054</td>
</tr>
<tr>
<td>P6</td>
<td>78.587</td>
<td>72.880</td>
<td>77.303</td>
<td>76.054</td>
</tr>
<tr>
<td>P7</td>
<td>73.831</td>
<td>63.517</td>
<td>60.228</td>
<td>60.975</td>
</tr>
<tr>
<td>P8</td>
<td>78.050</td>
<td>75.057</td>
<td>68.387</td>
<td>71.068</td>
</tr>
<tr>
<td>P9</td>
<td>83.692</td>
<td>81.552</td>
<td>69.366</td>
<td>73.323</td>
</tr>
<tr>
<td>P10</td>
<td>85.286</td>
<td>70.227</td>
<td>67.530</td>
<td>68.537</td>
</tr>
<tr>
<td>P11</td>
<td>80.443</td>
<td>76.167</td>
<td>77.880</td>
<td>76.557</td>
</tr>
<tr>
<td>P12</td>
<td>74.847</td>
<td>69.629</td>
<td>70.820</td>
<td>69.164</td>
</tr>
<tr>
<td>Mean</td>
<td>75.892</td>
<td>69.194</td>
<td>71.580</td>
<td>69.225</td>
</tr>
<tr>
<td>(SD)</td>
<td>5.620</td>
<td>6.969</td>
<td>7.857</td>
<td>6.407</td>
</tr>
</tbody>
</table>
Figure 4  Simulated and experimental joint angles for light switch task (A) maximal speed reach component, (B) self-selected speed reach component, (C) maximal speed reach and return components, (D) self-selected speed reach and return components.

Figure 5  Simulated and experimental paths in Cartesian space for maximal speed reach component of the (A) light switch reach task and (B) drawer closing task.

Task parameters are defined by the placement of the participant and manipulated objects. The initial joint variables are taken from the experimentally recorded data set, which is averaged at each time point over the five repetitions of each task to yield $\hat{\theta}$. In the case of the light switch task, the time taken to press the switch is calculated and
Table 3  Drawer closing task fitting error.

<table>
<thead>
<tr>
<th></th>
<th>Maximal speed</th>
<th></th>
<th></th>
<th></th>
<th>Self-selected speed</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Reach only</td>
<td>Reach</td>
<td>Return</td>
<td>Reach and return</td>
<td>Reach only</td>
<td>Reach</td>
<td>Return</td>
<td>Reach and return</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>P1</td>
<td>1.328</td>
<td>2.082</td>
<td>2.273</td>
<td>3.139</td>
<td>2.552</td>
<td>2.969</td>
<td>2.954</td>
<td>4.429</td>
</tr>
<tr>
<td>P2</td>
<td>2.531</td>
<td>3.147</td>
<td>2.367</td>
<td>3.986</td>
<td>3.421</td>
<td>2.187</td>
<td>2.350</td>
<td>3.247</td>
</tr>
<tr>
<td>P3</td>
<td>1.508</td>
<td>1.982</td>
<td>1.672</td>
<td>2.621</td>
<td>5.129</td>
<td>7.294</td>
<td>6.760</td>
<td>9.930</td>
</tr>
<tr>
<td>P4</td>
<td>1.636</td>
<td>2.043</td>
<td>1.511</td>
<td>2.561</td>
<td>4.049</td>
<td>4.937</td>
<td>3.371</td>
<td>6.038</td>
</tr>
<tr>
<td>P6</td>
<td>1.738</td>
<td>2.070</td>
<td>1.753</td>
<td>2.827</td>
<td>3.976</td>
<td>3.851</td>
<td>2.883</td>
<td>5.098</td>
</tr>
<tr>
<td>P12</td>
<td>0.987</td>
<td>1.733</td>
<td>1.460</td>
<td>2.367</td>
<td>1.684</td>
<td>2.098</td>
<td>2.985</td>
<td>3.690</td>
</tr>
<tr>
<td>Mean</td>
<td>83.961</td>
<td>76.145</td>
<td>71.437</td>
<td>71.706</td>
<td>83.961</td>
<td>76.868</td>
<td>71.706</td>
<td>77.194</td>
</tr>
<tr>
<td>(SD)</td>
<td>(5.634)</td>
<td>(6.245)</td>
<td>(7.346)</td>
<td>(5.634)</td>
<td>(5.674)</td>
<td>(6.245)</td>
<td>(7.346)</td>
<td>(5.674)</td>
</tr>
</tbody>
</table>

Fitting results for the full reach and return light switch task are shown in Table 2 at both speeds, with values also given for the reach and return sub-components of the task. The mean fitting for the reach and return light switch task across the 12 participants is now 69.166% and 64.626% for maximal and self-selected speeds, respectively. Figure 4C and D shows the signals $\theta$ and $\hat{\theta}$ (solid and dotted lines, respectively) at both speeds for participant P1.

For the drawer closing task, the time when the participant first makes contact with the drawer knob and the time when the drawer is fully closed are directly extracted from the experimental data. In addition, $p$ is set equal to the position of the drawer knob when fully closed, and $\hat{D}$ and $d$ define the constraint imposed by the drawer’s runners. These are used in Eq. 22 to provide a model of the reach component (which includes the drawer closing action). Fitting results are shown in Table 3 and confirm a mean accuracy of 77.194% and 83.961% for the self-selected and maximal speeds, respectively. Figure 6A and B shows the experimental and simulated joint angles at...
both speeds for participant P1. Figure 5B shows the corresponding paths in Cartesian space for the reach component of the maximal speed drawer closing switch task. These results again confirm that the model can accurately fit the experimental data.

Mean fitting results for the reach and return drawer closing task are shown in Table 2 and confirm a total accuracy of 69.417% and 73.698% for the self-selected and maximal speeds, respectively. Figure 6C and D shows the signals $\theta$ and $\hat{\theta}$ (solid and dotted lines, respectively) at both speeds for participant P1.

**Conclusions**

Fitting results confirm that unimpaired human motion can be accurately predicted using a simple underlying model of the upper limb, together with a suitable optimisation procedure. This means that functional motion can be posed as optimisation problems and embedded in FES control schemes to enable clinically relevant tasks to be assisted. Furthermore, since a reference trajectory is no longer required, this can be achieved without the need to collect data for each new movement (as would be necessary to provide joint reference trajectories). Moreover, the patient’s own estimated dynamic parameters in the model mean the task is tailored to each patient. The close link with volitional control this facilitates is anticipated to lead to more effective FES-based rehabilitation.

**Acknowledgments:** This work is supported by the Engineering and Physical Sciences Research Council grant number EP/I01909X/1.
Appendix A. Additional operator derivations

A.1 Drawer closing task

In the case of the drawer closing task, Eq. 7 is replaced by
\[
\min_{\mathbf{u}} \| \mathbf{u} \|^2, \quad \text{such that} \quad \begin{align*}
\mathbf{Dy} &= d \\
y(T) &= k(\theta(T)) = \mathbf{p} \\
\dot{y}(T) &= \frac{d}{dt}k(\theta(T)) = J(\theta(T))\dot{\theta}(T) = 0
\end{align*}
\]

The operator \( D: [0, T] \rightarrow [T, T_1] \) is defined as \( Dy(t) = (\mathbf{Dy})_t = \dot{\mathbf{Dy}}(t) = \partial k(\theta(t)) \), where \( D \) is a 1x3 matrix with full row rank. The tracking requirement \( \dot{\mathbf{y}}(t) = d \) on \([T, T_1]\), is used to ensure that the hand follows a straight line in Cartesian space (representing the draw knob trajectory). To solve Eq. 22, introduce the linear operator
\[
\begin{align*}
\mathbf{Dg}_1(\bar{x}, \bar{u})\mathbf{u} := & \int_0^T C(T)\Phi(T, r)B(r)u(r)\,dr, \quad t \in [T, T_1]
\end{align*}
\]

so that Eq. 13 is replaced by
\[
\begin{align*}
\min_{\mathbf{u}} & \| \mathbf{u} \|^2, \quad \text{such that} \quad \\
\mathbf{g}_1(\bar{x}, \bar{u})\mathbf{u} &= \mathbf{p} \\
\mathbf{g}_2(\bar{x}, \bar{u})\mathbf{u} &= \mathbf{0} \\
\mathbf{Dg}_1(\bar{x}, \bar{u})\mathbf{u} &= \mathbf{d}
\end{align*}
\]

and the input solution (19) is replaced by
\[
\begin{align*}
\mathbf{u}_- = & \mathbf{g}_{123}(\bar{x}, \bar{u}) (\mathbf{g}_{123}(\bar{x}, \bar{u})\mathbf{g}_{123}(\bar{x}, \bar{u}))^{-1} \begin{bmatrix} \mathbf{p} \\ \mathbf{0} \\ \mathbf{d} \end{bmatrix}
\end{align*}
\]

with the operators
\[
\begin{align*}
\mathbf{g}_{123}(\bar{x}, \bar{u})\mathbf{u} &:= \begin{bmatrix} \mathbf{g}_{12}(\bar{x}, \bar{u})\mathbf{u} \\ \mathbf{Dg}_1(\bar{x}, \bar{u})\mathbf{u} \end{bmatrix}, \\
\mathbf{g}^{'}_{123}(\bar{x}, \bar{u}) \left[ \begin{array}{c} \mathbf{v}_1 \\ \mathbf{v}_2 \end{array} \right] &:= \mathbf{g}_{12}(\bar{x}, \bar{u})\mathbf{v}_1 + \mathbf{g}^{'}_{12}(\bar{x}, \bar{u})D^2\mathbf{v}_2.
\end{align*}
\]

The iterative update (21) used in step (g) of Table 1 is replaced by
\[
\begin{align*}
\mathbf{u}_{k+1} &= \mathbf{u}_k + c_1\mathbf{g}_{123}(\mathbf{x}_k, \mathbf{u}_k) (\mathbf{g}_{123}(\mathbf{x}_k, \mathbf{u}_k) \\
&= \begin{bmatrix} \mathbf{p} \\ \mathbf{0} \\ \mathbf{d} \end{bmatrix} \dot{\mathbf{y}}_k
\end{align*}
\]

A.2 Light switch task return component

To model the return component of the light switch task, Eq. 7 must be extended by redefining the linear operators as
\[
\begin{align*}
\mathbf{g}_1(\bar{x}, \bar{u})\mathbf{u} := & \int_0^T C(T)\Phi(T, r)B(r)u(r)\,dr, \\
\mathbf{g}_2(\bar{x}, \bar{u})\mathbf{u} := & \int_0^T C(T)\Phi(T, r)B(r)u(r)\,dr
\end{align*}
\]

and replacing \( \mathbf{p} \) by \( \begin{bmatrix} \mathbf{p} \\ \frac{\partial k(\theta(T))}{\partial \theta(T)} \end{bmatrix} \). Here \( T \) is the time taken to press the light switch, and \( T_1 \) is the time taken for the hand to return to the starting position.

A.3 Drawer closing task return component

To model the return component of the drawer closing task, Eq. 22 must again be extended by redefining the linear operators as
\[
\begin{align*}
\mathbf{g}_1(\bar{x}, \bar{u})\mathbf{u} := & \int_0^T C(T)\Phi(T, r)B(r)u(r)\,dr, \\
\mathbf{g}_2(\bar{x}, \bar{u})\mathbf{u} := & \int_0^T C(T)\Phi(T, r)B(r)u(r)\,dr
\end{align*}
\]

References


