Control Engineering Practice 20 (2012) 386–396

Contents lists available at SciVerse ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Recursive identification of Hammerstein systems with application to electrically stimulated muscle

Fengmin Le*, Ivan Markovsky, Christopher T. Freeman, Eric Rogers

Information: Signals, Images, Systems Research Group, School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK

ARTICLE INFO

Article history:
Received 15 June 2010
Accepted 17 August 2011
Available online 11 January 2012

Keywords:
Recursive identification
Hammerstein system
Muscle model
Functional electrical stimulation

ABSTRACT

Modeling of electrically stimulated muscle is considered in this paper where a Hammerstein structure is selected to represent the isometric response. Motivated by the slowly time-varying properties of the muscle system, recursive identification of Hammerstein structures is investigated. A recursive algorithm is then developed to address limitations in the approaches currently available. The linear and nonlinear parameters are separated and estimated recursively in a parallel manner, with each updating algorithm using the most up-to-date estimation produced by the other algorithm at each time instant. Hence the procedure is termed the alternately recursive least square (ARLS) algorithm. When compared with the leading approach in this application area, ARLS exhibits superior performance in both numerical simulations and experimental tests with electrically stimulated muscle.

1. Introduction

Modelling of electrically stimulated muscle has been a widely investigated area and plays an important role in the analysis of motor control and the design of motor system neuroprostheses. Muscle representations are also necessary in the development of increasingly effective rehabilitation systems for patients (de Kroon, Ijzerman, Chae, Lankhorst, & Zilvold, 2005). There exist a large number of models developed for different aspects of muscle contraction under both isometric, e.g. Bernotas, Crago, and Chizeck (1986) and non-isometric conditions, e.g. Durfee and Palmer (1994), considering the modulation of the output force by varying either the number of active muscle fibers, e.g. Chizeck, Crago, and Kofman (1988) or the frequency of the activation, e.g. Bai, Cai, Dudley-Javorosk, and Shields (2009) and Cai, Bai, and Shields (2010). The most widely assumed structure used in model-based control of electrically stimulated muscle is the Hill-type model (Hill, 1938). This describes the output force as the product of three independent experimentally measured factors: the force–length property, the force–velocity property and the nonlinear muscle activation dynamics under isometric conditions respectively, termed simply activation dynamics (AD) of the stimulation input. The first two account for passive elastic and viscous properties of the muscle and comprise static functions of the muscle length and velocity (Freeman et al., 2009a; Jezernik, Wassink, & Keller, 2004; Lan, 2002; Schauer et al., 2005; Riener & Fuhr, 1998). The activation dynamics capture the active properties of the muscle, and are almost uniformly represented by a Hammerstein structure.

This structure is a crucial component of the muscle model since in most applications joint ranges and velocities are small so that the isometric behavior of muscle dominates. The widespread use of a Hammerstein structure to represent the activation dynamics is due to correspondence with biophysics: the static nonlinearity represents the isometric recruitment curve (IRC), which is the static gain relation between stimulus activation level, and steady-state output torque when the muscle is held at a fixed length. The linear dynamics represents the muscle contraction dynamics, which combines with the IRC to give the overall torque generated.

There are many identification methods applicable to Hammerstein models and in general they can be classified into two categories: iterative, for example, Narendra and Gallman (1966), Zhu (2000) and Westwick and Kearney (2001), and Dempsey and Westwick (2004) with application to stretch reflex electromyogram, and non-iterative methods, for example, an equation-error parameter estimation method in Chang and Luus (1971), an optimal two-stage algorithm in Bai (1998), a blind approach in Bai (2002) and decoupling methods in Bai (2004). However, after reviewing the existing techniques, limitations were encountered when identifying an input–output model of electrically stimulated muscles with incomplete paralysis. These drawbacks were associated with both the structure of the linear and nonlinear Hammerstein components, and the form of the excitation inputs employed. Consequently Le, Markovsky, Freeman, and Rogers (2010) developed two iterative algorithms suitable for

* Corresponding author.
E-mail address: fl07r@ecs.soton.ac.uk (F. Le).

0967-0661/$ - see front matter © 2012 Published by Elsevier Ltd.
doi:10.1016/j.conengprac.2011.08.001

the identification of electrically stimulated muscles in subjects with incomplete paralysis, and their efficacy was demonstrated through application to experimentally measured data.

The algorithms developed in Le et al. (2010) represent significant progress in the identification of electrically stimulated muscles, but the models were only verified over a short time interval (20 s duration). However, when applied to stroke rehabilitation, stimulation must be applied during intensive, goal orientated tasks in order to maximize improvement in motor control (Schmidt & Lee, 1998). In clinical trials this translates to sustained application of stimulation during each treatment session between 30 min and 1 h duration (de Kroon et al., 2005). In this case, slowly time-varying properties of the muscle system arise due to fatigue, changing physiological conditions or spasticity (Graham, Thrasher, & Popovic, 2006). Motivated by this, online, also termed recursive, identification will be considered in this paper, in which the model parameters are updated once new data is available. Only a few of the existing identification methods for Hammerstein structures are recursive, and can be divided into three categories.

The first category is the recently developed recursive subspace identification method by Bako, Mercere, Lecoeuche, and Lovera (2009), where the nonlinear function is first recursively estimated by over-parameterizations and component-wise least squares support vector machines (LS-SVM). This is followed by estimation of the Markov parameters by recursive least squares and then a propagator-based method is used to recursively estimate system state-space model matrices from these parameters. This procedure does not have sparsity due to the LS-SVM model, and the resulting computational load makes it unsuitable for real-time implementation.

The second category is stochastic approximation (Chen, 2004; Greblicki, 2002) where a stochastic approximation algorithm with expanding truncations is developed for recursive identification of Hammerstein systems. Two major issues with this method are the rather slow convergence rates, and the lack of information on how to select the optional parameters in the algorithm.

The third category is recursive least squares (RLS) or extended recursive least squares (ERLS). The RLS algorithm is a well known method for recursive identification of linear-in-parameter models and if the data is generated by correlated noise, the parameters describing the model of the correlation can be estimated by ERLS. Here, a typical way to use these two algorithms is to treat each of the cross-product terms in the Hammerstein system equations as an unknown parameter. This procedure, which results in an increased number of unknowns, is usually referred to as the over-parameterization method (Bai, 1998; Chang & Luus, 1971). After this step, the RLS or ERLS method can be applied (Boutayeb & Darouach, 1995; Boutayeb, Aubry, & Darouach, 1996; Zhao & Chen, 2009).

The limitations of current algorithms are stated next and used to justify some of the critical choices necessary for this work to progress.

- The first two categories have only been applied in simulation and the stochastic approximation has not the considered time-varying linear dynamics. This, together with the drawbacks described above, is the reason for not considering them further for the application treated in this paper. The third category is the most promising as it has already been applied to electrically stimulated muscle in Chia, Chow, and Chizeck (1991) and Ponikvar and Munih (2001).
- Most of the test signals used comprise random noise in order to guarantee persistent excitation, even when applied to the human muscle (Ponikvar & Munih, 2001), and use pseudorandom binary sequences. However, this type of signal, which excites the motor units abruptly, will cause patient discomfort and may elicit an involuntary response, as reported in Baker, McNeal, Benton, Bowman, and Waters (1993). In Chia et al. (1991) a test consisting of 25 pulses is used, each of which is of 1 s duration in the form of a noisy triangular wave. This test meets our requirements but is too short to exhibit time-varying properties.
- The most relevant previous work is Chia et al. (1991) where the system considered had linear constraints and RLS was developed for constrained systems. However, the results given do not establish that the constraints are achieved. For example, even when considering the prediction error, the posteriori estimated output without constraints is superior to the one with constraints. Thus, the idea of adding constraints to RLS, leading to increased computational load, still merits consideration.

Overall, RLS has the greatest potential for application to electrically stimulated muscle, but the problem of consistent estimation must be resolved (Chen, 2004; Chia et al., 1991). The deficiency of RLS is illustrated in Section 3, where noise and excitation inputs that correspond with those encountered in the rehabilitation application domain are employed, and confirm its unsatisfactory performance. This motivates development of an alternative recursive algorithm in Section 2.3, as well as the design of a long-period test signal which is persistently exciting and also gradually recruits the motor units, and hence is suitable for application to patients. This problem is addressed in Section 4.

2. Problem statement and solution methods

2.1. Problem statement

Consider the discrete-time SISO Hammerstein model shown in Fig. 1. The linear block is represented by ARX model:

$$y(k) = \frac{B(q)}{A(q)} w(k) + \frac{1}{A(q)} v(k) \tag{1}$$

where

$$B(q) = b_0 q^{-d} + b_1 q^{-(d+1)} + \cdots + b_n q^{-(n+d)} \quad \text{and}$$
$$A(q) = 1 + a_1 q^{-1} + \cdots + a_l q^{-l} \tag{2}$$

q^{-1} is the delay operator and n, l and d are the number of zeros, poles and the time delay order, respectively. The parameters n, l and d are assumed to be known. The nonlinearity is represented by a sum of the known nonlinear functions $f_1, f_2, \ldots, f_m$ and a bias:

$$w(k) = f(u(k)) = \beta_0 + \sum_{i=1}^{m} \beta_i f_i(u(k)) \tag{3}$$

The identification problem considered is:

Given N consecutive input–output data measurements $\{u(k), y(k)\}$ estimate recursively the linear parameters $[a_1, \ldots, a_l, b_0, \ldots, b_n]$ in (2) and the nonlinear parameters $[\beta_0, \ldots, \beta_m]$ in (3).

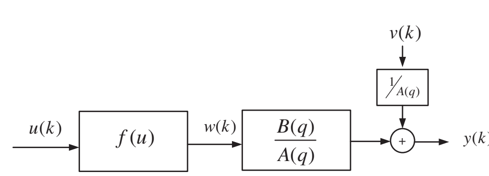

Fig. 1. Hammerstein system.

2.2. RLS algorithm

As the leading approach currently available, the well known RLS algorithm will be applied first, where in order to make the model linear in the parameters, over-parameterization of the Hammerstein structure is required. This results in the extended parameter vector θ

$$\theta(k) = [a_1, \ldots, a_l, \gamma_{01}, \ldots, \gamma_{0m}, \ldots, \gamma_{n1}, \ldots, \gamma_{nm}, \delta]^T \tag{4}$$

where

$$\gamma_{ij} = b_i\beta_j \quad i = 0,1,\ldots,n \;\; j = 1,2,\ldots,m \quad \text{and} \quad \delta = \beta_0 \sum_{i=0}^{n} b_i \tag{5}$$

This is recursively estimated by RLS, using a forgetting factor λ, $0 < \lambda \le 1$ which weights the most recent data at unity, and data that is n time units old at λ^n. In the second step, singular value decomposition (SVD) is used to recover the original parameters.

2.3. Alternately recursive least square (ARLS) algorithm

The use of over-parameterization and subsequent rank-1 approximation often leads to a model which poorly fits the original data (as illustrated in Section 3). A recursive identification method is therefore developed which avoids over-parameterization by splitting the model into nonlinear and linear components, where each is identified independently using a parallel implementation. This method builds on Le et al. (2010) in which two iterative algorithms were developed for Hammerstein systems with differing noise models, and in each case nonlinear and linear parameters were alternately optimized by different projection algorithms. Both algorithms developed in Le et al. (2010) use least squares optimization for off-line identification, and therefore extend naturally to the online case through application of RLS, the one with simpler implementation and faster computation time will be taken as a starting point. By invoking certain approximations, this algorithm can be implemented recursively as follows:

- Recursive identification of linear parameters:
  As described in Le et al. (2010), the parameters of the ARX model can be separated into linear and nonlinear parameter vectors

$$\theta_n = [\beta_0 \;\; \cdots \;\; \beta_m]^T \quad \text{and} \quad \theta_l = [a_1 \;\; \cdots \;\; a_l \;\; b_0 \;\; \cdots \;\; b_n]^T \tag{6}$$

  Assuming that the nonlinear parameter vector θ_n is known at the kth time instant, $y(k)$ can be expressed as a function of linear parameters $a_1(k), \ldots, a_l(k), b_0(k), \ldots, b_n(k)$ only

$$\begin{aligned} y(k) = {} & -a_1(k)y(k-1) - \cdots - a_l(k)y(k-l) \\ & + b_0(k)f(u(k-d),\theta_n) + \cdots + b_n(k)f(u(k-d-n),\theta_n) + v(k) \end{aligned} \tag{7}$$

  or

$$y(k) = \phi_l^T(k,\theta_n)\theta_l(k) + v(k) \tag{8}$$

  where

$$\phi_l^T(k,\theta_n) = [-y(k-1) \;\; \cdots \;\; -y(k-l) \;\; f(u(k-d),\theta_n) \;\; \cdots \;\; f(u(k-d-n),\theta_n)] \tag{9}$$

  A forgetting factor λ_l is used in the RLS algorithm to minimize the criterion

$$V_l(\theta_l,k) = \frac{1}{2}\sum_{i=1}^{k} \lambda_l^{k-i}(y(k) - \phi_l^T(k,\hat{\theta}_n(k-1))\theta_l(k))^2 \tag{10}$$

  where the nonlinear parameter vector is approximated by the estimated value at the previous time instant $k-1$. Thus, the recursive algorithm for the linear parameter vector $\theta_l(k)$ is

$$P_l(k) = \frac{1}{\lambda_l}\left(P_l(k-1) - \frac{P_l(k-1)\phi_l(k,\hat{\theta}_n(k-1))\phi_l^T(k,\hat{\theta}_n(k-1))P_l(k-1)}{\lambda_l I + \phi_l^T(k,\hat{\theta}_n(k-1))P_l(k-1)\phi_l(k,\hat{\theta}_n(k-1))}\right) \tag{11}$$

$$\hat{\theta}_l(k) = \hat{\theta}_l(k-1) + P_l(k)\phi_l^T(k,\hat{\theta}_n(k-1))(y(k) - \phi_l^T(k,\hat{\theta}_n(k-1))\hat{\theta}_l(k-1)) \tag{12}$$

- Recursive identification for the nonlinear parameter vector:
  As in the linear case, it is first assumed that the linear parameter vector θ_l is known. Hence, at the kth time instant,

$$\begin{aligned} \underbrace{y(k) + a_1y(k-1) + \cdots + a_ly(k-l)}_{A(q,\theta_l)y(k)} = {} & \beta_0(k)\sum_{i=0}^{n} b_i \\ & + \beta_1(k)\sum_{i=0}^{n} b_if_1(u(k-d-i)) + \cdots \\ & + \beta_m(k)\sum_{i=0}^{n} b_if_m(u(k-d-i)) + v(k) \end{aligned} \tag{13}$$

  or, in matrix form,

$$A(q,\theta_l)y(k) = \phi_n^T(k,\theta_l)\theta_n(k) + v(k) \tag{14}$$

  where

$$\phi_n^T(k,\theta_l) = \left[\sum_{i=0}^{n} b_i \quad \sum_{i=0}^{n} b_if_1(u(k-d-i)) \cdots \sum_{i=0}^{n} b_if_m(u(k-d-i))\right] \tag{15}$$

  In order to recursively update the nonlinear parameter vector, the linear parameter vector is approximated by the estimated value from previous time instant, resulting in the RLS criterion

$$V_n(\theta_n,k) = \frac{1}{2}\sum_{i=1}^{k} \lambda_n^{k-i}(A(q,\hat{\theta}_l(k-1))y(k) - \phi_n^T(k,\hat{\theta}_l(k-1))\theta_n(k))^2 \tag{16}$$

  The recursive algorithm for the nonlinear parameter vector is

$$P_n(k) = \frac{1}{\lambda_n}\left(P_n(k-1) - \frac{P_n(k-1)\phi_n(k,\hat{\theta}_l(k-1))\phi_n^T(k,\hat{\theta}_l(k-1))P_n(k-1)}{\lambda_n I + \phi_n^T(k,\hat{\theta}_l(k-1))P_n(k-1)\phi_n(k,\hat{\theta}_l(k-1))}\right) \tag{17}$$

$$\begin{aligned} \hat{\theta}_n(k) = {} & \hat{\theta}_n(k-1) + P_n(k)\phi_n^T(k,\hat{\theta}_l(k-1))(A(q,\hat{\theta}_l(k-1))y(k) \\ & - \phi_n^T(k,\hat{\theta}_l(k-1))\hat{\theta}_n(k-1)) \end{aligned} \tag{18}$$

2.4. Initial values for the two algorithms

- RLS:
  In the case of the standard RLS algorithm, initial values are required for the parameter vector θ, and the auxiliary matrix P. These are calculated from several initial samples by the batch least squares algorithm. The number of samples is determined by the dimension of the regressor ϕ in order to obtain the unique solution

$$\theta_{ini} = (\Phi^T\Phi)^{-1}\Phi^TY \quad \text{and} \quad P_{ini} = (\Phi^T\Phi)^{-1} \tag{19}$$

  where $Y = [y(1) \;\cdots\; y(T_{\text{ini}})]^T$ and $\Phi = [\phi(1) \;\cdots\; \phi(N_{\text{ini}})]^T$. The matrix Φ may become singular or poorly conditioned and hence there exist problems with computing its inverse. Consequently, a regularization is applied, in which case (19) becomes

$$\theta_{ini} = (\Phi^T\Phi+\delta I)^{-1}\Phi^T Y \quad \text{and} \quad P_{ini} = (\Phi^T\Phi+\delta I)^{-1} \tag{20}$$

The regularization parameter δ is chosen to be small, say $\delta = 10^{-2}$–10^{-4}, compared to the magnitude of the elements of Φ.

- ARLS:
  For ARLS, the initial values are θ_l, θ_n, P_l and P_n. The initial values for θ_l and θ_n are found by applying rank-1 approximation, and then calculating Φ_l and Φ_n, where $\Phi_l = [\phi_l(1,\theta_n) \ \cdots \ \phi_l(T_{\text{ini}},\theta_n)]^T$ and $\Phi_n = [\phi_n(1,\theta_l) \cdots \phi_n (T_{\text{ini}},\theta_l)]^T$. The initial values for P_l and P_n are therefore $P_l = (\Phi_l^T\Phi_l)^{-1}$ and $P_n = (\Phi_n^T\Phi_n)^{-1}$ and again regularization may be applied to avoid ill-conditioning.

3. Simulation study

The two techniques are now compared in simulation using similar noise levels and excitation inputs to those encountered in the rehabilitation application domain (Hughes et al., 2009). These simulations enable comparisons to be drawn across a number of criteria, including error norm and convergence of parameter estimates, which is not possible in the experimental setting considered in Section 4. Comparison is also made with their offline counterparts that, in both cases, involves exchanging the RLS update procedure for offline LS optimization using full test data.

3.1. Numerical example

The numerical example employed in Boutayeb et al. (1996) has been used to provide a fair platform for comparison, in addition to being highly relevant to the work reported in this paper

$$B(q) = q^{-1}+0.6q^{-2}, \quad A(q) = 1-q^{-1}+0.8q^{-2}$$

$$\text{and} \quad f(u) = 2.8u-4.8u^2+5.7u^3 \tag{21}$$

The input signal used in Boutayeb et al. (1996) is a zero mean white noise sequence, which is widely employed in recursive identification to guarantee persistent excitation. However, as previously noted, this is unsuitable for the present application, and will therefore be replaced by a half cosine wave signal which has similar characteristics to signals used in rehabilitation (see Hughes et al., 2009). To ensure this signal is persistently exciting, the diminishing excitation technique (Chen & Guo, 1991) has been applied

$$u(k) = u_d(k)+\frac{\varepsilon(k)}{k^{\tau/2}} \tag{22}$$

where $u_d(k)$ is the designed input and $\varepsilon(k)$ is a bounded random sequence with $\tau > 0$ sufficiently small. The added measurement noise $v(k)$ is zero mean white noise such that the signal-to-noise ratio

$$SNR = \left(\frac{var(y_{sig})}{var(y_{noi})}\right)^{1/2} \tag{23}$$

is equal to 10, 5, or 2. Here $y_{sig} = (B(q)/A(q))w(k)$ is the noise-free output signal, $y_{noi} = (1/A(q))v(k)$ is the correlated noise and $var(\cdot)$ the population variance of a finite-size sequence,

$$var(y) = \frac{1}{N-1}\sum_{t=1}^{N}(y_t-\overline{y})^2 \quad \text{where} \quad \overline{y} = \frac{1}{N}\sum_{t=1}^{N} y_t \tag{24}$$

3.2. Results

The two recursive algorithms, RLS and ARLS, are compared in terms of the following three aspects:

1. Error norm: The error norm is the normalized error between the true values and the estimated values of the linear and nonlinear parameters, which is defined as

   $$Error\ norm = \sqrt{\left(\frac{\|\theta_n-\hat{\theta}_n\|_2}{\|\theta_n\|_2}\right)^2+\left(\frac{\|\theta_l-\hat{\theta}_l\|_2}{\|\theta_l\|_2}\right)^2}$$

   The recursive algorithms, together with their associated offline batch implementations, have been applied to 100 independent trials using different noise levels. The mean error norms of the updated parameter values at each time instant from the two recursive algorithms are traced in Fig. 2 and compared with the reference lines, that is, the mean error norms after 2000 samples from the two batch algorithms, least squares (LS) and the first iterative algorithm (iterative), developed in Le et al. (2010). The mean and standard deviation of the error norms after 2000 samples for 100 independent trials using different noise levels are also listed in Table 1.
2. Convergence of parameter estimates: To show how fast the estimated parameter values converge to the true values, Fig. 3 plots the mean values of the updated nonlinear parameters for 100 independent trials using different noise levels.
3. Effect of an abrupt change in the true model: To determine how well the two recursive algorithms track a time-variant model, an abrupt change in the true model after 2000 samples is introduced, and the nonlinear function becomes

   $$f(u) = 2.8u-5.1u^2+5.7u^3 \tag{25}$$

   where the coefficient of the term of the second degree changes from -4.8 to -5.1, which is such a slight change that it cannot be observed from the output plot. The convergence plots for the nonlinear parameter estimates from the two recursive algorithms are compared in Fig. 4(a) where $\lambda = 0.9993$ is chosen for RLS and $\lambda_l = 1$ and $\lambda_n = 0.9993$ for ARLS. The ARLS plots are magnified in Fig. 4(b) to show more clearly that the ARLS estimates converge to the true values after 5000 samples.

3.3. Discussion

The ARLS is superior to RLS across all the simulation criteria employed in this numerical example, especially in the noisy environment. For a low noise level, such as SNR = 10, RLS performs comparably well with ARLS, see Figs. 2(a) and 3(a) and the first row in Table 1. However, when the measurements are more significantly contaminated by noise, such as SNR = 5 or even 2, RLS takes a long time or even fails to converge to the true value. In this circumstance, ARLS can still maintain the error norm at a lower level, and even approach the performance of the iterative algorithm which involves several iterations to optimize the parameters. These conclusions are illustrated by Figs. 2(b), (c) and 3(b), (c) respectively and the last two rows in Table 1.

The reason for these conclusions is that the estimates from LS, which is the non-recursive algorithm from which RLS arises, are already poor compared with the iterative algorithm, as highlighted by the fourth and fifth columns in Table 1. Hence, after calculation in a recursive fashion, the estimates from RLS are likely to be the same or inferior to those from LS which uses all the data for estimation. The iterative algorithm generates the best error norm for 2000 samples data. By avoiding parameterization through a parallel recursive

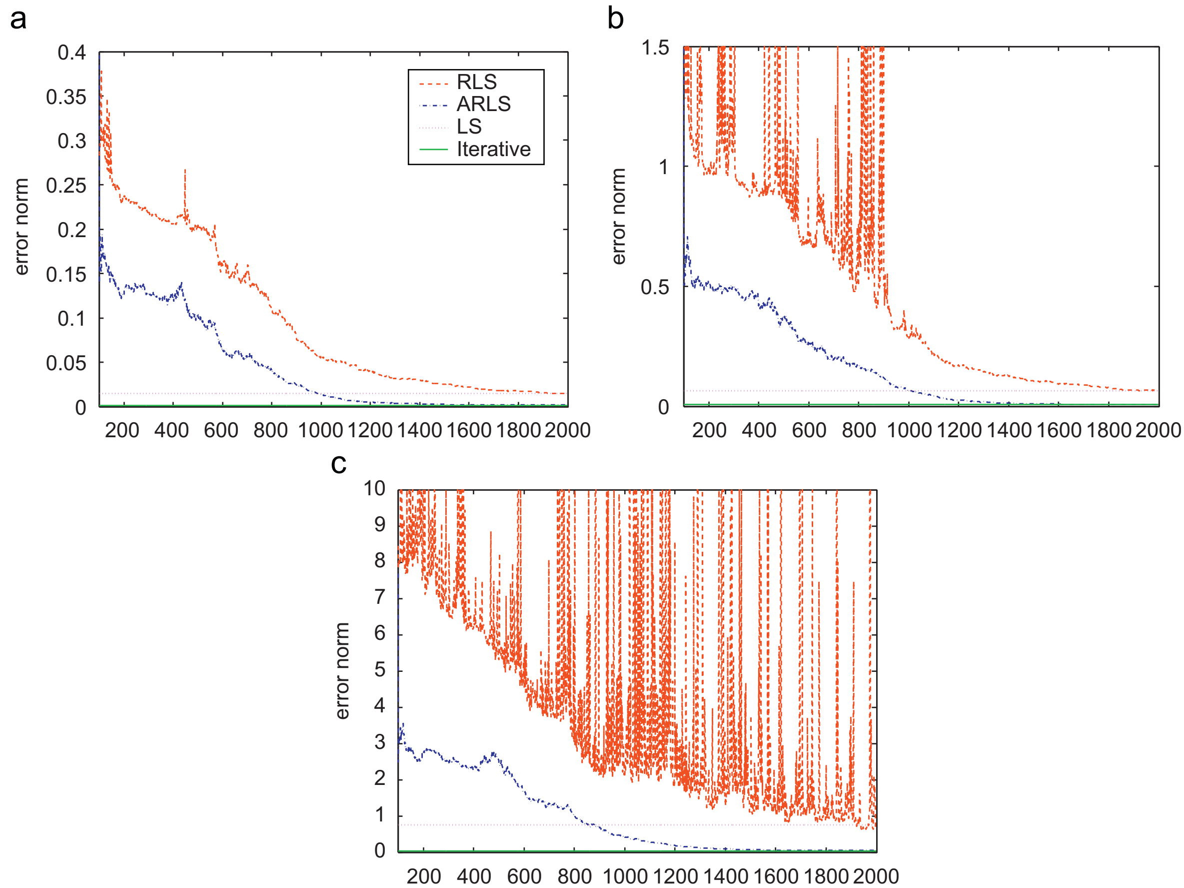

Fig. 2. Numerical example: the mean error norms of the updated parameter values at each time instant for 100 independent trials using different noise levels (SNR = 10 (a), 5 (b) and 2 (c)) from the two recursive algorithms (red dashed line for RLS and blue dash-dot line for ARLS) are compared with the mean error norms after 2000 samples from the two batch algorithms (magenta dotted line for LS and green solid line for iterative). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Numerical example: the mean and standard deviation of the error norms after 2000 samples for 100 independent trials using different noise levels (SNR = 10, 5 and 2) from the two recursive algorithms (RLS and ARLS) and the two batch algorithms (LS and iterative).

	Recursive		Batch	
	RLS	ARLS	LS	Iterative
SNR = 10	0.0146 ± 0.0102	0.0017 ± 0.0010	0.0146 ± 0.0102	0.0014 ± 0.0008
SNR = 5	0.0650 ± 0.0420	0.0074 ± 0.0041	0.0650 ± 0.0420	0.0065 ± 0.0034
SNR = 2	0.7586 ± 2.2713	0.0404 ± 0.0253	0.7586 ± 2.2713	0.0338 ± 0.0224

structure, ARLS confirms substantial improvement relative to RLS, but cannot quite match the offline iterative algorithm.

When tested with a time-varying system, where the true model parameter is slightly changed from its nominal value, the RLS estimates suffer a larger oscillation after the change and even fail to converge to the true values, as illustrated in Fig. 4(a), but ARLS can still converge to the true model parameters after 9000 samples, see Fig. 4(b). It is expected that ARLS will outperform RLS when applied to electrically stimulated muscle since the response of stimulated muscle is time-varying and experimental results from such a system are typically very noisy.

4. Application to electrically stimulated muscle

In this section, the recursive algorithms developed above are applied to online identification of the response of electrically stimulated muscle. As is almost universal in the literature on muscle modeling, the isometric properties are modeled as a Hammerstein structure. The non-linearity has been parametrized in a number of ways, taking the form of a simple gain with saturation (Ferrarin, Palazzo, Riener, & Quintern, 2001), a piece-wise linear function (Hunt, Munih, Donaldson, & Barr, 1998; Lan, 2002) and a predefined functional form (Riener & Quintern, 1997; Previdi & Carpanzano, 2003). The linear dynamics have been assumed to be first order in Lan (2002), a series of two first order systems in Riener and Fuhr (1998), Happee and der Helm (1995), and Jezernik et al. (2004), critically damped second order in Durfee and MacLean (1989), Baratta and Solomonow (1990), Veltink, Chizeck, Crago, and El-Bialy (1992) or second order with possible transport delay in Chizeck et al. (1988) and Hunt et al. (1998).

In the tests which follow, the linear block is represented by an ARX model described by (1) and (2), with the parameters

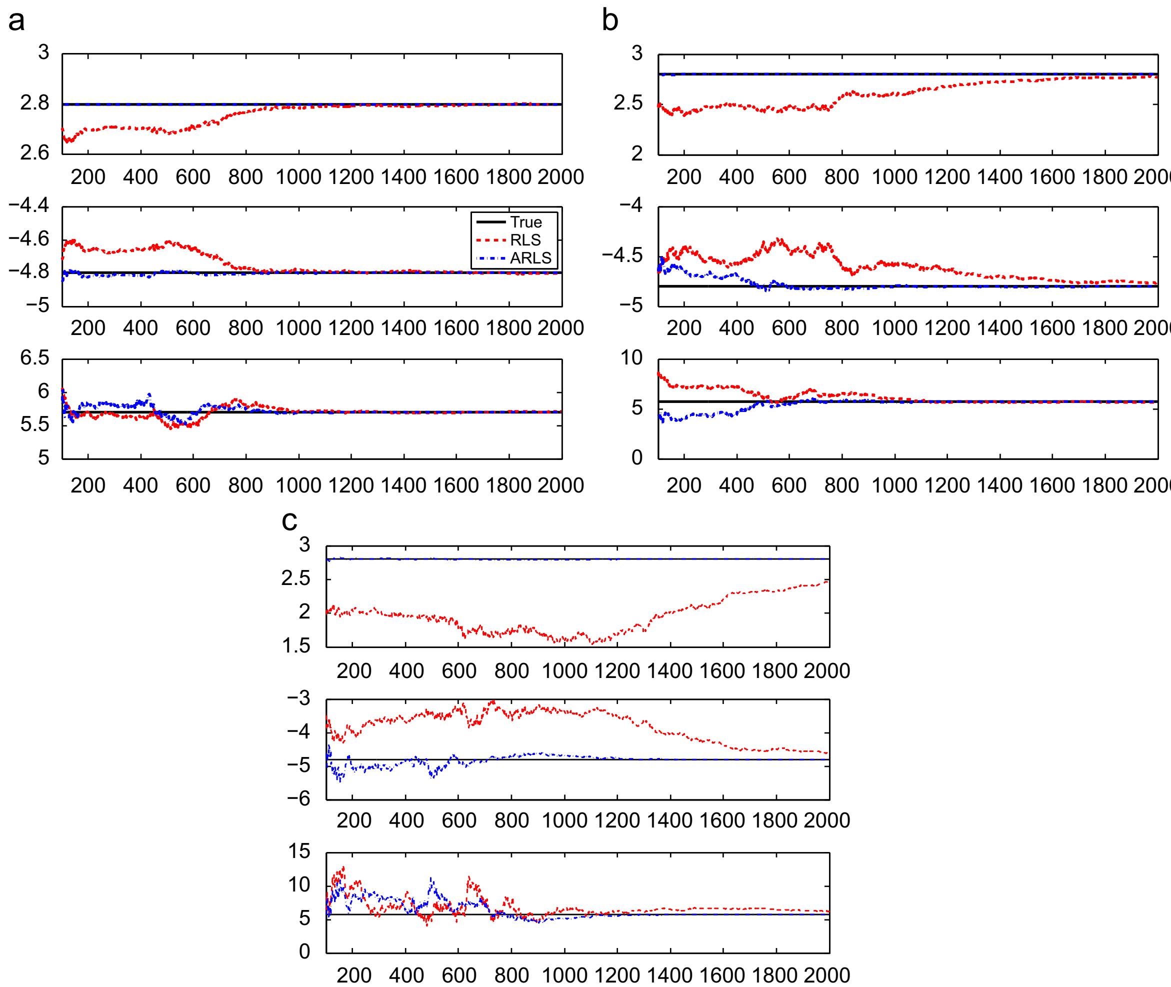

Fig. 3. Numerical example: the mean of the updated values for 100 independent trials using different noise levels (SNR = 10, 5 and 2) from the two recursive algorithms (red dashed line for RLS and blue dash-dot line for ARLS) are compared with the true values (black solid line) of the nonlinear parameters. (a) SNR = 10 (b) SNR = 5 (c) SNR = 2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

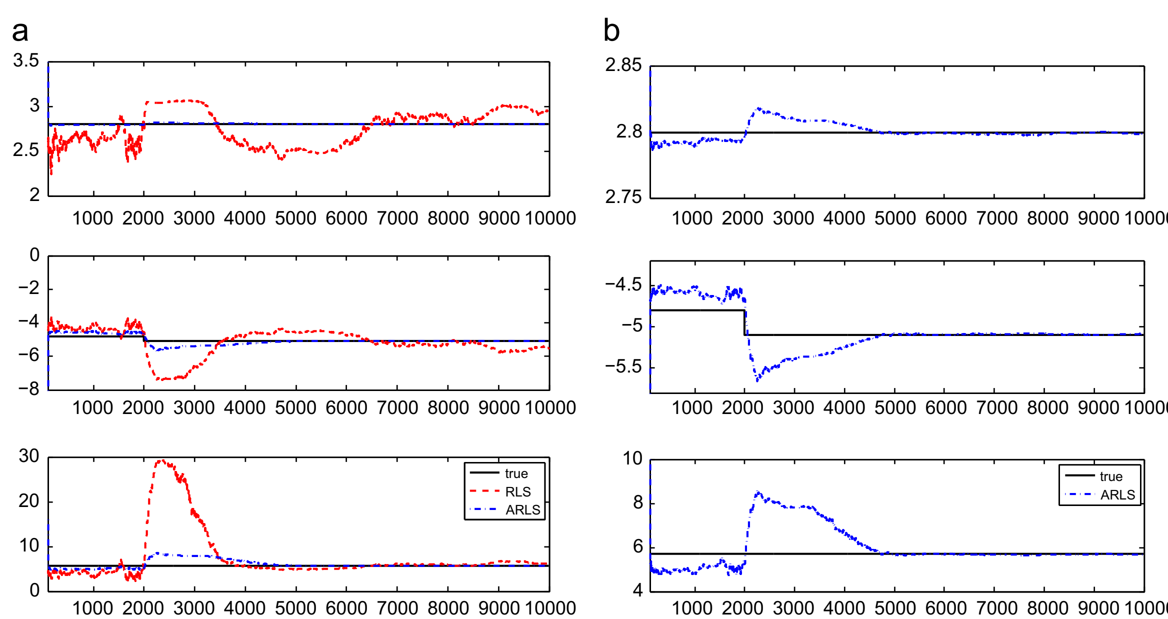

Fig. 4. Numerical example with an abrupt change after 2000 samples: the time trajectory of the estimated nonlinear parameter values from the two recursive algorithms (red dashed line for RLS and blue dash-dot line for ARLS) at SNR = 10. (a) RLS and ARLS (b) close-up of ARLS trajectory. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$l=2, n=2, d=1$. To provide a smooth function with continuous derivatives suitable for subsequent control, the nonlinear function $f(u)$ is represented by the cubic spline

$$f(u)=\beta_0+\beta_1 u+\beta_2 u^2+\beta_3 u^3+\sum_{i=4}^{m+3}\beta_i |u-u_{i-3}|^3 \tag{26}$$

where $u_{\min}=u_1<u_2<u_3<\cdots<u_m=u_{\max}$ are the spline knots. These forms were selected in Le et al. (2010) in the production of a model suitable for application in the stroke rehabilitation domain.

4.1. Experiment setup

Experimental tests have been carried out using a workstation which has been developed as a platform for upper limb stroke rehabilitation. It incorporates a five-link planar robotic arm which includes a six axis force/torque sensor in its extreme link, and an overhead projector used to display trajectories to patients. The system has been used in a clinical trial in which electrical stimulation was applied to the patient's triceps to assist their completion of trajectory tracking tasks. Full details of the system are given in Freeman et al. (2009b), which includes experimental validation of the sensor and stimulation hardware.

Recursive identification tests were performed on a single unimpaired subject, and took place on several independent days.

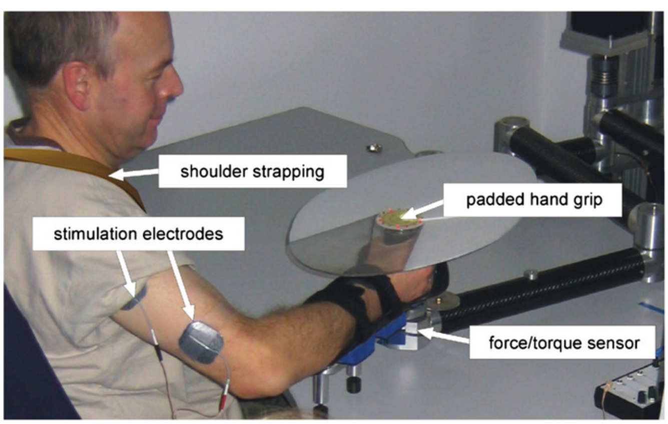

Fig. 5. Robotic workstation.

The participant's upper arm and forearm lengths were first taken, they were then seated in the workstation and their right arm was strapped to the extreme link of the robotic arm. Straps were applied about the upper torso to prevent shoulder and trunk movement (as shown in Fig. 5). The subject's upper limb was then moved over as large an area as possible and a kinematic model of the arm produced using the recorded measurements. This was used to transform the force recorded by the force/torque sensor to torque acting about the elbow (see Freeman et al., 2009b). The electrodes were then positioned on the lateral head of triceps and adjusted so that the applied stimulation generated maximum forearm movement. The stimulation consists of a series of bi-phasic pulses at 40 Hz, whose pulsewidth is variable from 0 to 300 μs with a resolution of 1 μs. The amplitude, which is fixed throughout all subsequent tests, is determined by setting the pulsewidth equal to 300 μs and slowly increasing the applied voltage until a maximum comfortable limit is reached. A sample frequency of 1.6 KHz is used by the real-time hardware.

The position of the robotic arm was then fixed at an elbow extension angle of approximately $\pi/2$ rads using a locking pin. This removes the non-isometric components of the biomechanical model, and hence the resulting system corresponds to a Hammerstein structure (comprising the muscle model with the addition of passive elastic torque from the remaining arm which may also vary in time). The model's input is the stimulation pulsewidth, and its output is the torque about the elbow. The recursive identification tests last for 10 min, comprising 10 repeated waves of either a half-cosine function, or a staircase signal, added to which the diminishing excitation technique has been used to make the input signals persistently exciting. The two kinds of input signal have similar characteristics to those used in rehabilitation (see Hughes et al., 2009) and the corresponding output signals are plotted in Fig. 6.

4.2. Results

Here, two recursive algorithms, RLS and ARLS, are compared in the following aspects:

1. One-step ahead prediction: In order to evaluate the accuracy of the recursive algorithms, the measured torque outputs y are compared with the one-step ahead predicted outputs $\hat{y}$ using

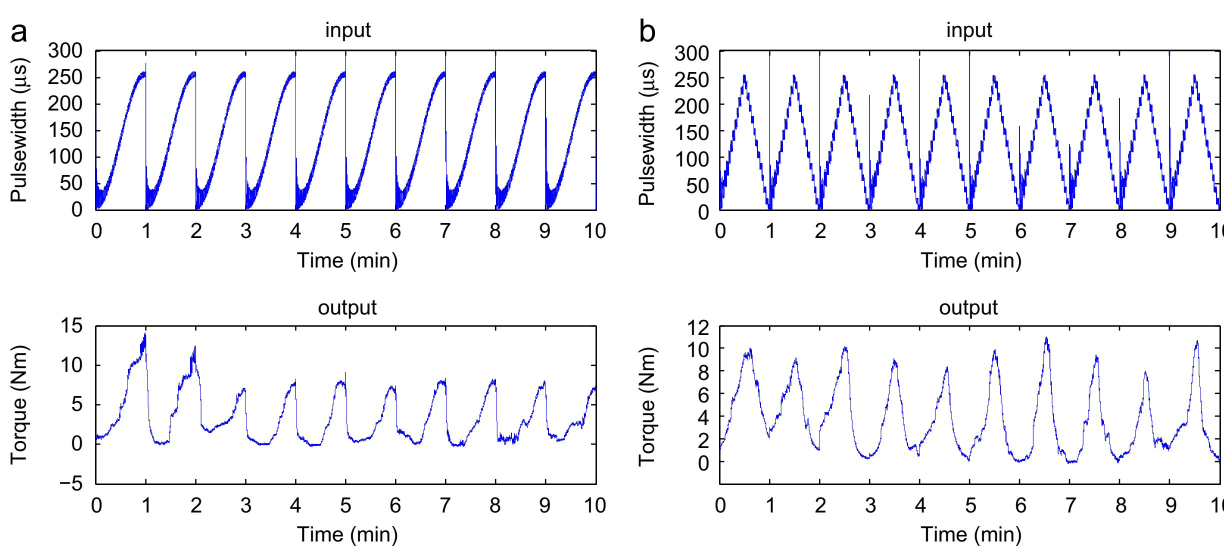

Fig. 6. The input and output signals for recursive identification tests. (a) Half cosine wave. (b) Staircase wave.

a fitting criterion defined as follows:

$$\textit{Fitting criterion} = \left(1 - \frac{\|y-\hat{y}\|_2}{\|y-\bar{y}\|_2}\right) \times 100$$

where $\bar{y}$ is the mean value of y appearing in (24), and $\hat{y}$ is defined as

$$\hat{y}(k+1) = G(q,\hat{\theta}_l(k)) f(u,\hat{\theta}_n(k))$$

which is a one-step ahead prediction, using the updated model at the time instant k to predict the output at the next time instant $k+1$. Table 2 lists the fitting criterion for both half cosine and staircase wave inputs respectively, and considers both the whole 10-min dataset and the first 1-min dataset, the latter of which contains less time-varying information. The corresponding fitting plots are shown in Fig. 7.

Table 2
Muscle tests: fitting criterion for the two recursive algorithms: RLS and ARLS.

Test duration	Half cosine wave input		Staircase wave input	
	RLS	ARLS	RLS	ARLS
1 min	−10.0244	87.9188	−130.4187	80.3162
10 min	−52.3874	61.3267	−408.2148	57.4049

2. Long-period prediction: In order to demonstrate the predictive ability for the longer period, the two recursive algorithms together and their corresponding offline batch implementations have been applied to the first 3, 4, 5, 6 and 7 min of the data respectively, and the resulting models then used to predict the corresponding outputs for the remaining time period. The fitting criterion results for both identification and prediction are listed in Table 3. Fig. 8 shows the fitting plots in the case of the first 5 min for the identification phase as it is a representative of all the results obtained.

Table 3
Muscle tests: fitting criterion results for identification and validation.

Part of the data	Used for	Recursive		Batch	
		RLS	ARLS	LS	Iterative
First 3 min	Identification	−115.2605	40.0999	−115.2605	46.4750
Next 7 min	Prediction	−119.4234	30.4174	−119.4234	−0.1979
First 4 min	Identification	−90.0323	38.0301	−90.0323	40.5572
Next 6 min	Prediction	−100.0545	60.3873	−100.0545	19.5024
First 5 min	Identification	−60.7013	40.6237	−60.7013	42.2075
Next 5 min	Prediction	−70.6340	66.5179	−70.6340	29.9268
First 6 min	Identification	−10.9030	40.7501	−10.9030	41.7214
Next 4 min	Prediction	−7.5807	69.0567	−7.5807	37.4819
First 7 min	Identification	23.4563	44.5587	23.4563	45.0254
Next 3 min	Prediction	17.2580	63.8627	17.2580	40.9369

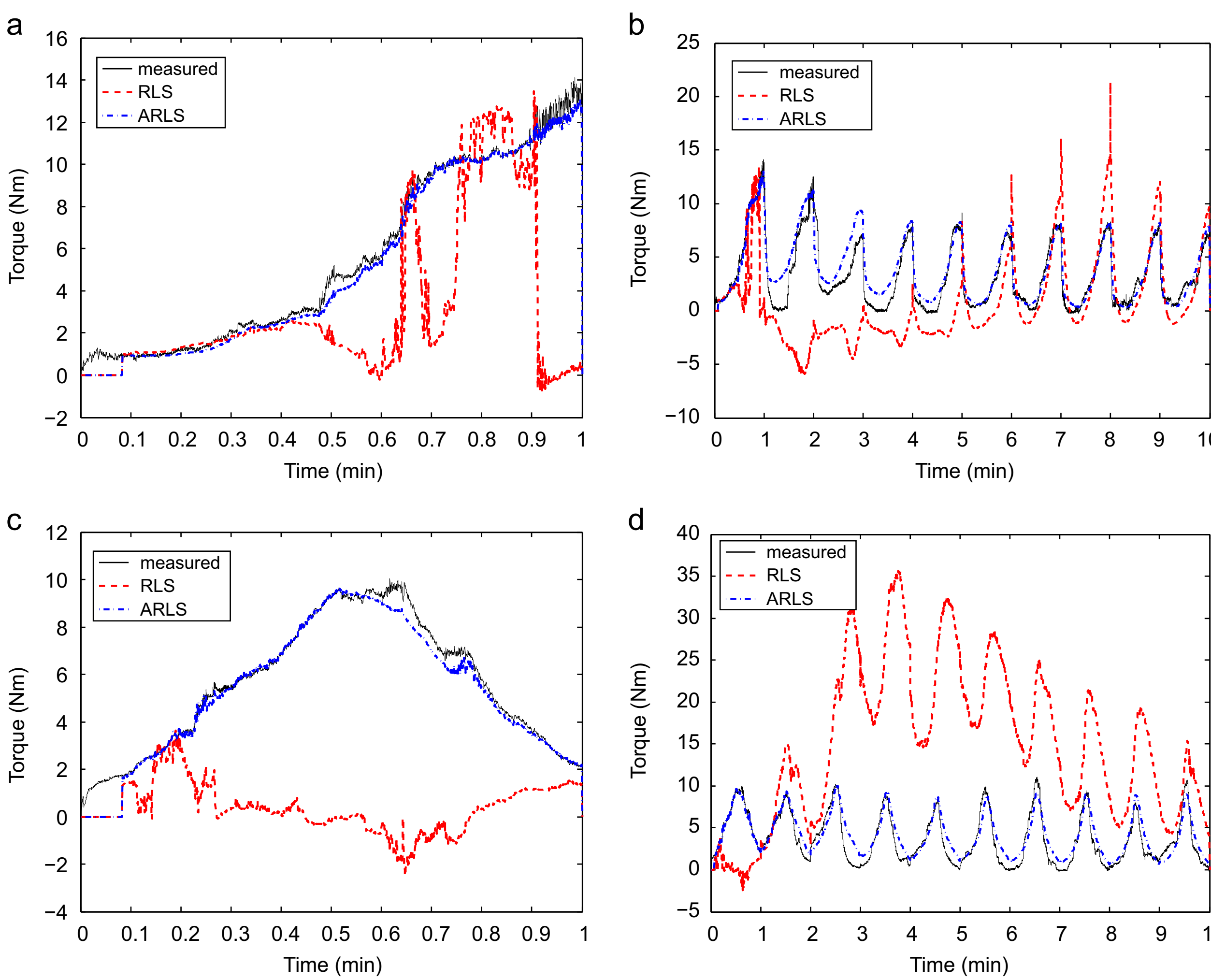

Fig. 7. Muscle tests: the measured outputs and the one-step ahead predicted outputs from the two recursive algorithms: RLS and ARLS. (a) Half cosine wave input: 1 min. (b) Half cosine wave input: 10 min. (c) Staircase wave input: 1 min. (d) Staircase wave input: 10 min.

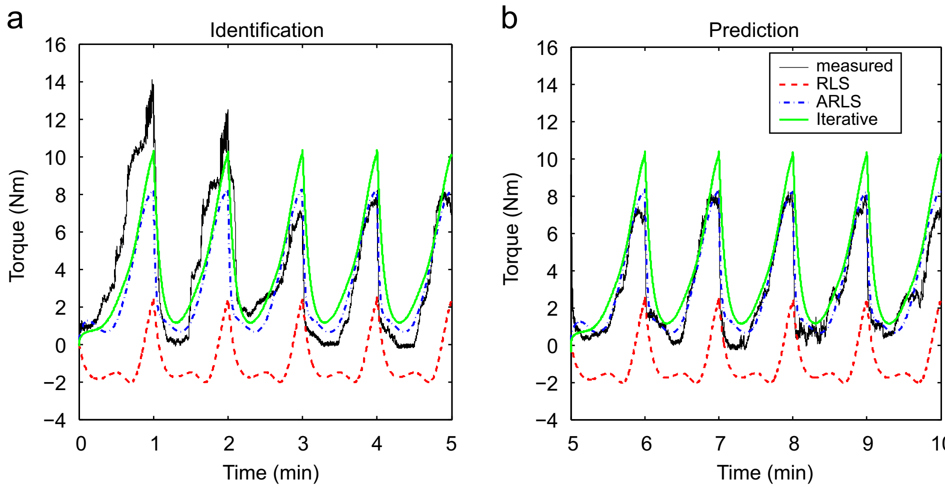

Fig. 8. (a) Identification and (b) prediction fitting plots using a 50% split between identification and prediction.

Table 4
Muscle tests: computational time in seconds for a single updating step for the two recursive algorithms: RLS and ARLS.

Method	RLS	ARLS
Computational time	0.0019	1.0989×10^{-4}

Table 5
Muscle tests: fitting criterion results for the measured outputs and modeled outputs from the two batch algorithms: LS and Iterative.

Test duration	Half cosine wave input		Staircase wave input	
	LS	Iterative	LS	Iterative
1 min	−49.5367	86.9901	−133.4340	85.2571
10 min	23.8095	44.6698	−46.7207	12.2363

3. Computational time: Since the algorithms are intended for online implementation in real-time, their computation time is an important factor. The time taken to perform a single updating step for both recursive algorithms is listed in Table 4.

4.3. Discussion

- Batch algorithms vs recursive algorithms: Batch algorithms are off-line and use all available data to perform identification process in order to find the best model according to the minimization criterion. Table 5 gives the identification results for the two batch algorithms: LS and iterative respectively. It is clear that LS cannot deal with the noisy and time-varying experimental data and the iterative algorithm greatly improves the best fit rates. For 1 min data, the iterative algorithm achieves a fitting criterion of around 85%, which is in agreement with the results reported in Le et al. (2010), where a 20 s test was used. However, for the 10 min data, which contains more time-varying information, even the iterative algorithm cannot find a time-invariant model to fit all the data and only yields a fitting criterion of 12% for the staircase input.

  It follows from Table 3 that the iterative algorithm provides the best identification fitting rates in all cases but performs very poorly for prediction. However, ARLS is very good at prediction and gives even higher fitting criterion values for prediction compared to identification. The results of Fig. 8 reflects the fact that the iterative algorithm uses all the identification data to calculate the best model, which, due to the time-varying properties of the system, produces an identified model which may be interpreted as an 'average' response, see Fig. 8(a). However, since ARLS updates the estimated model such that it is responsive to changes in underlying dynamics, the model produced after 5 min, even when it has not fit past data particularly well, is the best model to predict the future output, as illustrated by Fig. 8(b). Finally, the batch algorithm is not good at identification of long-period data from a time-varying system.

  Conversely, batch algorithms are computationally heavy and not suitable for real-time implementation, as illustrated

Table 6
Muscle tests: computational time for 1 min and 10 min data from the two batch algorithms, LS and Iterative, in seconds.

Test duration	LS	Iterative
1-min	0.1155	1.9881
10-min	28	70

Table 7
Fitting criterion for RLS and ARLS with difference choices of parameter λ using 10 min half cosine wave.

λ		Best fit rate (%)
(a) RLS		
1		−52.3874
0.9999		−109.1885
0.9998		−141.6934
0.9997		−103.2831
0.9990		−64.0752
λ_l	λ_n	Best fit rate (%)
(b) ARLS		
1	1	61.3267
0.9999	0.9999	63.6053
0.9998	0.9999	65.8187
0.9997	0.9999	67.6394
0.9996	0.9998	68.7207
0.9995	0.9997	70.8437
0.9994	0.9996	43.0805

by Table 6, where the computational times for 1 min and 10 min data from the two batch algorithms are listed. The computational time grows considerably with the increase in samples, so that there comes a point when calculations cannot be completed before the arrival of new data.

- RLS vs ARLS: From the above analysis, it is necessary to perform recursive rather than batch identification for the experimental data. Here the two recursive identification algorithms are applied, RLS and ARLS. Both cases first use several samples to generate an initial estimate, less than 0.1 min of data, and then update the linear and nonlinear parameters at each time instant. For the noisy experimental data and slowly time-varying muscle system, it is clear that ARLS is far superior to RLS. For 1 min data, one-step ahead prediction can track the output well, shown in Fig. 7(a) and (c) and for 10 min data, it also can capture long term variation in the muscle properties, as illustrated by Fig. 7(b) and (d). This is confirmed by the plots in Fig. 8(a) and (b) which show a high level of accuracy for ARLS during identification and prediction respectively, whilst also demonstrating the inconsistency of the RLS estimates in these noisy conditions. Moreover, ARLS is even faster than RLS, because ARLS splits the algorithm into two parallel ones, each of which entails low-dimensional matrix multiplication.

  Another advantage of ARLS over RLS is that ARLS has two separate weighting parameters for linear and nonlinear parameters, λ_l and λ_n. In the real muscle system, the linear and nonlinear parameters represent two different mechanisms (muscle activation and recruitment respectively) which change over time at different rates. The ability to choose individual weighting parameters for each mechanisms provides clear selection and performance advantages over a single λ parameter.

  In the previous recursive process, the weighting parameters λ, λ_l and λ_n are fixed at 1, and the implications of this choice

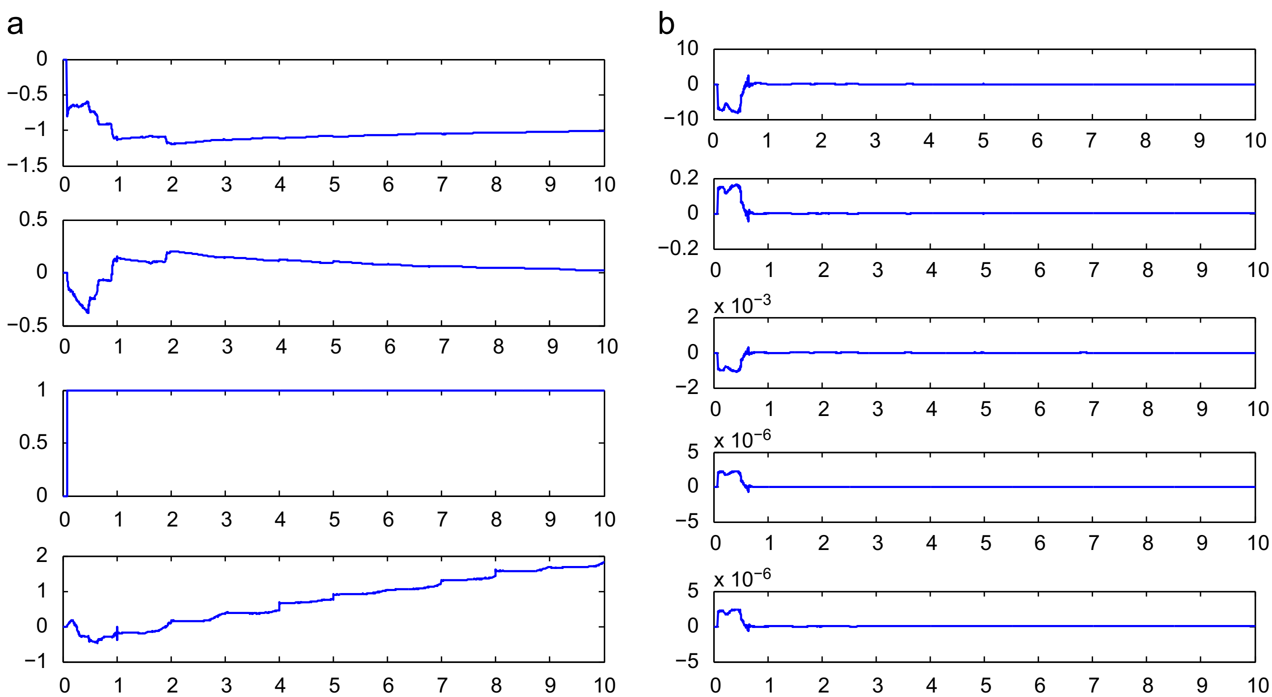

Fig. 9. Muscle tests: the time trajectory of the estimated values of the (a) linear and (b) nonlinear parameters using ARLS.

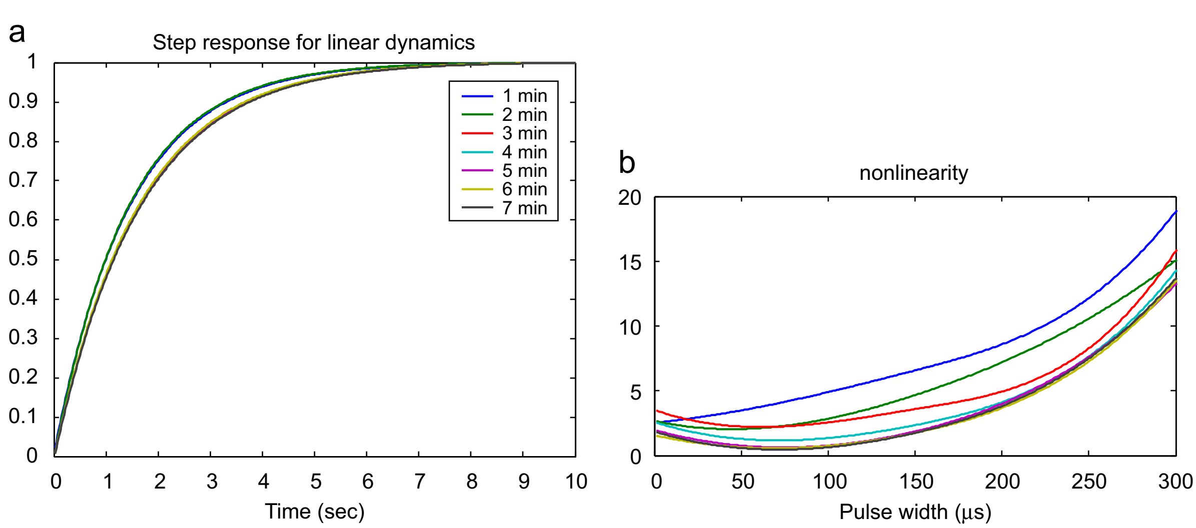

Fig. 10. Muscle tests: (a) identified linear block step responses and (b) identified nonlinearities at 1 min time intervals using ARLS.

are examined using Table 7(a) and (b). For RLS, there is no improvement when tuning the λ parameter, while for ARLS, the fitting criterion reaches 70% for $\lambda_l = 0.9995$ and $\lambda_n = 0.9997$.

- Time-variance of the muscle model: Fig. 9 shows the time trajectory of the estimated values for the linear and nonlinear parameters from ARLS, and the step response for the identified linear block and nonlinearity are plotted against time in Fig. 10. These illustrate the underlying physiological changes in the muscle system over time.

5. Conclusions

A recursive identification algorithm has been developed for Hammerstein structures, in which the linear and nonlinear parameters are recursively identified in an alternate manner. The algorithm has been shown to outperform the leading RLS alternative in both numerical simulation, and when applied to the experimental identification of electrically stimulated muscle. In order to reproduce the results reported in the paper the software and experimental data are available from the following website: http://users.ecs.soton.ac.uk/fl07r/research/research.html The identification procedure will be shortly employed in clinical trials with stroke patients for the purpose of rehabilitation.

Acknowledgments

The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013)/ERC Grant agreement number 258581 "Structured low-rank approximation: Theory, algorithms, and applications".

References

Bai, E. W. (1998). An optimal two-stage identification algorithm for Hammerstein–Wiener nonlinear systems. *Automatica*, *34*(3), 333–338.

Bai, E. W. (2002). A blind approach to the Hammerstein–Wiener model identification. *Automatica*, *38*(6), 967–979.

Bai, E. W. (2004). Decoupling the linear and nonlinear parts in Hammerstein model identification. *Automatica*, *40*(4), 671–676.

Bai, E. W., Cai, Z., Dudley-Javorosk, S., & Shields, R. K. (2009). Identification of a modified Wiener–Hammerstein system and its application in electrically stimulated paralyzed skeletal muscle modeling. *Automatica*, *45*(3), 736–743.

Baker, L. L., McNeal, D. R., Benton, L. A., Bowman, B. R., Waters, R. L. (1993). Neuromuscular electrical stimulation: A practical guide, 3rd ed.

Bako, L., Mercere, G., Lecoeuche, S., & Lovera, M. (2009). Recursive subspace identification of Hammerstein models based on least squares support vector machines. *IET Control Theory and Applications*, *3*(9), 1209–1216.

Baratta, R., & Solomonow, M. (1990). The dynamic response model of nine different skeletal muscles. *IEEE Transactions on Biomedical Engineering*, *37*(3), 243–251.

Bernotas, L., Crago, P. E., & Chizeck, H. J. (1986). A discrete-time model of electrically stimulated muscle. *IEEE Transactions on Biomedical Engineering*, *33*(9), 829–838.

Boutayeb, M., Aubry, D., & Darouach, M. (1996). A robust and recursive identification method for MISO Hammerstein model. In *UKACC international conference on control*, 2–5 September, (Vol. 1, pp. 234–239).

Boutayeb, M., & Darouach, M. (1995). Recursive identification method for MISO Wiener–Hammerstein model. *IEEE Transactions on Automatic Control*, *40*(2), 287–291.

Cai, Z., Bai, E. W., & Shields, R. K. (2010). Fatigue and non-fatigue mathematical muscle models during functional electrical stimulation of paralyzed muscle. *Biomedical Signal Processing and Control*, *5*(2), 87–93.

Chang, F. H. I., & Luus, R. (1971). A non-iterative method for identification using Hammerstein model. *IEEE Transactions on Automatic Control*, *16*(5), 464–468.

Chen, H. F. (2004). Pathwise convergence of recursive identification algorithms for Hammerstein systems. *IEEE Transactions on Automatic Control*, *49*(4), 1641–1649.

Chen, H. F., & Guo, L. (1991). *Identification and stochastic adaptive control*. Boston, MA: Birkhäuser.

Chia, T. L., Chow, P. C., & Chizeck, H. J. (1991). Recursive parameter identification of constrained systems: An application to electrically stimulated muscle. *IEEE Transactions on Biomedical Engineering*, *38*(5), 429–442.

Chizeck, H. J., Crago, P. E., & Kofman, L. S. (1988). Robust closed-loop control of isometric muscle force using pulsewidth modulation. *IEEE Transactions on Biomedical Engineering*, *35*(7), 510–517.

de Kroon, J. R., Ijzerman, M. J., Chae, J., Lankhorst, G. J., & Zilvold, G. (2005). Relation between stimulation characteristics and clinical outcome in studies using electrical stimulation to improve motor control of the upper extremity in stroke. *Journal of Rehabilitation Medicine*, *37*, 65–74.

Dempsey, E. J., & Westwick, D. T. (2004). Identification of Hammerstein models with cubic spline nonlinearities. *IEEE Transactions on Biomedical Engineering*, *51*(2), 237–245.

Durfee, W. K., & MacLean, K. E. (1989). Methods for estimating isometric recruitment curves of electrically stimulated muscle. *IEEE Transactions on Biomedical Engineering*, *36*(7), 654–667.

Durfee, W. K., & Palmer, K. I. (1994). Estimation of force–activation, force–length, and force–velocity properties in isolated, electrically stimulated muscle. *IEEE Transactions on Biomedical Engineering*, *41*(3), 205–216.

Ferrarin, M., Palazzo, F., Riener, R., & Quintern, J. (2001). Model-based control of FES-induced single joint movements. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *9*(3), 245–257.

Freeman, C. T., Hughes, A. M., Burridge, J. H., Chappell, P. H., Lewin, P. L., & Rogers, E. (2009a). A model of the upper extremity using surface FES for stroke rehabilitation. *ASME Journal of Biomechanical Engineering*, *131*(1).

Freeman, C. T., Hughes, A.-M., Burridge, J. H., Chappell, P. H., Lewin, P. L., & Rogers, E. (2009b). A robotic workstation for stroke rehabilitation of the upper extremity using FES. *Medical Engineering and Physics*, *31*(3), 364–373.

Graham, G. M., Thrasher, T. A., & Popovic, M. R. (2006). The effect of random modulation of functional electrical stimulation parameters on muscle fatigue. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *14*(1), 38–45.

Greblicki, W. (2002). Stochastic approximation in nonparametric identification of Hammerstein systems. *IEEE Transactions on Automatic Control*, *47*(11), 1800–1810.

Happee, R., & der Helm, F. C. T. V. (1995). The control of shoulder muscles during goal directed movements, an inverse dynamic analysis. *Journal of Biomechanics*, *28*(10), 1179–1191.

Hill, A. V. (1938). Then heat of shortening and the dynamic constants of a muscle. *Proceedings of the Royal Society*, *126*, 136–195.

Hughes, A. M., Freeman, C. T., Burridge, J. H., Chappell, P. H., Lewin, P., & Rogers, E. (2009). Feasibility of iterative learning control mediated by functional electrical stimulation for reaching after stroke. *Journal of Neurorehabilitation and Neural Repair*, *23*(6), 559–568.

Hunt, K. J., Munih, M., Donaldson, N. N., & Barr, F. M. D. (1998). Investigation of the Hammerstein hypothesis in the modeling of electrically stimulated muscle. *IEEE Transactions on Biomedical Engineering*, *45*(8), 998–1009.

Jezernik, S., Wassink, R. G. V., & Keller, T. (2004). Sliding mode closed-loop control of FES: Controlling the shank movement. *IEEE Transactions on Biomedical Engineering*, *51*(2), 263–272.

Lan, N. (2002). Stability analysis for postural control in a two-joint limb system. *IEEE Transactions on Neural systems and Rehabilitation Engineering*, *10*(4), 249–259.

Le, F., Markovsky, I., Freeman, C. T., & Rogers, E. (2010). Identification of electrically stimulated muscle models of stroke patients. *Control Engineering Practice*, *18*(4), 396–407.

Narendra, K. S., & Gallman, P. G. (1966). An iterative method for the identification of nonlinear systems using a Hammerstein model. *IEEE Transactions on Automatic Control*, *11*(3), 546–550.

Ponikvar, M., & Munih, M. (2001). Setup and procedure for online identification of electrically stimulated muscle with matlab simulink. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *9*(3), 295–301.

Previdi, F., & Carpanzano, E. (2003). Design of a gain scheduling controller for knee-joint angle control by using functional electrical stimulation. *IEEE Transactions on Control Systems Technology*, *11*(3), 310–324.

Riener, R., & Fuhr, T. (1998). Patient-driven control of FES-supported standing up: A simulation study. *IEEE Transactions on Rehabilitation Engineering*, *6*(2), 113–124.

Riener, R., & Quintern, J. (1997). A physiologically based model of muscle activation verified by electrical stimulation. *Bioelectrochemistry and Bioenergetics*, *43*, 257–264.

Schauer, T., Negard, N. O., Previdi, F., Hunt, K. J., Fraser, M. H., & Ferchland, E., et al. (2005). Online identification and nonlinear control of the electrically stimulated quadriceps muscle. *Control Engineering Practice*, *13*, 1207–1219.

Schmidt, R. A., & Lee, T. D. (1998). *Motor control and learning: A behavioural emphasis* (3rd ed.). Human Kinetics Europe Ltd.

Veltink, P. H., Chizeck, H. J., Crago, P. E., & El-Bialy, A. (1992). Nonlinear joint angle control for artificially stimulated muscle. *IEEE Transactions on Biomedical Engineering*, *39*(4), 368–380.

Westwick, D. T., & Kearney, R. E. (2001). Separable least squares identification of nonlinear Hammerstein models: Application to stretch reflex dynamics. *Annals of Biomedical Engineering*, *29*(8), 707–718.

Zhao, W. X., & Chen, H. F. (2009). Adaptive tracking and recursive identification for Hammerstein systems. *Automatica*, *45*(12), 2773–2783.

Zhu, Y. (2000). Identification of Hammerstein models for control using ASYM. *International Journal of Control*, *73*(18), 1692–1702.