

Combined feedforward and feedback control of a redundant, nonlinear, dynamic musculoskeletal system

Dimitra Blana · Robert F. Kirsch ·
Edward K. Chadwick

Received: 6 June 2008 / Accepted: 11 March 2009 / Published online: 3 April 2009
© International Federation for Medical and Biological Engineering 2009

Abstract A functional electrical stimulation controller is presented that uses a combination of feedforward and feedback for arm control in high-level injury. The feedforward controller generates the muscle activations nominally required for desired movements, and the feedback controller corrects for errors caused by muscle fatigue and external disturbances. The feedforward controller is an artificial neural network (ANN) which approximates the inverse dynamics of the arm. The feedback loop includes a PID controller in series with a second ANN representing the nonlinear properties and biomechanical interactions of muscles and joints. The controller was designed and tested using a two-joint musculoskeletal model of the arm that includes four mono-articular and two bi-articular muscles. Its performance during goal-oriented movements of varying amplitudes and durations showed a tracking error of less than 4° in ideal conditions, and less than 10° even in the case of considerable fatigue and external disturbances.

Keywords Feedback control · Musculoskeletal modeling · Functional electrical stimulation · Shoulder · Elbow

1 Introduction

Functional electrical stimulation (FES) systems restore function after spinal cord injury by stimulating paralyzed

muscles to contract in appropriately coordinated patterns. Various FES systems have been developed that benefit different populations, such as standing systems for people with paraplegia [9], and hand grasp systems for people with mid-cervical level spinal cord injuries [15]. We are interested in expanding the FES benefits to individuals with high-level tetraplegia, whose entire upper extremity is essentially paralyzed. This FES system aims to restore shoulder and arm function to this population, and allow them to move their arm throughout a functional workspace.

Current FES systems include highly developed hardware (e.g., implanted stimulators and numerous electrodes), but very basic control algorithms to calculate the stimulation patterns needed for the desired function, namely, predefined patterns, fixed for every task [15]. This limits the functional benefit of the FES system to a set number of tasks, and does not guarantee its performance in the presence of muscle fatigue or unexpected perturbations. A more sophisticated controller is needed for our upper-extremity FES system, because unlike the cyclic movements of the lower extremity, upper-extremity tasks are goal directed [8]. This means that the amplitude, speed and direction of the motions change continuously, so the controller needs to continuously calculate the stimulation patterns for a wide range of motions commanded by the user.

There are different control strategies used in existing FES systems. Feedforward control is commonly used in clinical practice [13, 15]. The output of this type of controller depends only on the user command, and not on the system performance. The advantage of this design is that it does not need sensors to measure the system output, but the disadvantage is that it is unable to make corrections if the actual movement deviates from the desired movement. Feedback control uses sensors to monitor the system

D. Blana · R. F. Kirsch (✉) · E. K. Chadwick
Department of Biomedical Engineering,
Case Western Reserve University, Cleveland, OH, USA
e-mail: rfk3@case.edu

D. Blana
e-mail: dimitra.blana@case.edu

output, so it can correct for errors in the system trajectory [8]. Feedback is necessary in order to maintain good tracking performance in the presence of fatigue and any external disturbances encountered. However, inherent delays in the response of the system can cause problems to the feedback controller [1], especially in the case of fast movements [23]. Therefore, a combination of feedforward and feedback control has the best results, and is the preferred control method in several FES system designs [1, 6, 20], including the one presented in this paper.

The feedforward controller is typically an inverse-dynamic model of the controlled system [19]. Since an actual inverse-dynamic model is usually not available, data collected from the system itself are used to train an artificial neural network (ANN) to behave like the inverse-dynamic model. An ANN that includes time-delayed inputs can approximate any dynamic system based solely on the knowledge of inputs and outputs [19]. In order to train the network to represent the inverse dynamics of the system, inputs and outputs are recorded from the system and used as the outputs and inputs, respectively, of the ANN. This has been used in many FES controllers of single-joint, single-muscle systems [2, 3, 6, 20, 26]. However, in the case of a system with multiple muscles crossing one joint, the inverse of the system is not unique, and the ANN will train on the average of possible solutions, which is not always a correct solution [14].

For this reason, we have used the actual inverse-dynamic model of the system to train the ANN in the feedforward component of our controller. Optimization was used to ensure uniqueness of the solution. In [10], the inverse-dynamic model itself was used as a feedforward controller, but a relatively simple model had to be designed so that it was fast enough to run in real-time. An ANN, however, is a system of simple processing elements, connected into a network by a set of weights [19], so it is computationally light. Off-line training of the ANN according to the inverse-dynamic model allows us to include all the complexities of the musculoskeletal system in a real-time controller.

For the feedback part of the controller, the feasibility of PID controllers for use in FES applications has been examined [1, 6, 25]. However, the performance of these controllers has been limited. As discussed in [20], PID controllers are linear and have been incapable of providing good control over highly nonlinear musculoskeletal systems [4]. Moreover, the PID controllers mentioned above were all tested in single-joint, single-muscle systems, but the complexity of musculoskeletal systems increases substantially when they include more degrees of freedom and more muscles, some of them bi-articular, that cause mechanical coupling between joints [3]. A multivariable feedback controller that uses PI control is described by Lan

[17], but its performance was only evaluated under isometric conditions. These authors suggest that nonlinear control methods may be necessary for control of movements of a multi-joint system.

In order to address these issues, we have created a neuro-PID controller for the feedback loop, which combines the attractive features of PID control (robustness and ease of implementation) with the nonlinear nature of ANN. The goal of the ANN is to model the nonlinear relationships among muscles and joints, allowing the linear PID controller to deal solely with the dynamic response.

The feedforward–feedback FES controller presented in this paper was designed and tested in simulation. Model-based evaluations were used to explore various control strategies before implementing invasive, expensive FES systems in human subjects. The model-based approach is intended to allow initial development of the control system to a point where a human implementation can be justified. The model used includes two joints, and six muscles, two of them bi-articular. This system is sufficiently complex to include both the nonlinear properties of the muscles themselves and the nonlinearities and complicated interactions between multiple muscles and joints. The controller was evaluated for a large set of goal-directed movements that cover the range of both joints. Muscle fatigue and external disturbances were also simulated to evaluate the performance of the controller for realistic conditions.

2 Methods

2.1 The model

The model used to design and test the controller is a two-dimensional model of the upper extremity, in the horizontal plane (no gravity). It includes six muscles (anterior and posterior deltoid, long head of the biceps, brachialis, long and lateral head of the triceps) and two degrees of freedom (shoulder flexion–extension and elbow flexion–extension). The range of the shoulder angle is from -20 to 110° , and the range of the elbow angle is from 0 to 170° . Four of the muscles are mono-articular, and two (long head of biceps and long head of triceps) are bi-articular. A schematic of the model is shown in Fig. 1.

The muscle and joint parameters for the model were obtained from cadaver studies by Klein-Breteler et al. [16]. These parameters include the position of joint centers, inertial parameters for body segments, and the optimal fiber length, origin and insertion, tendon slack length, and physiological cross-sectional area of every muscle. The muscle model is a Hill-type model that includes contraction dynamics, force-length dependence and force-velocity dependence. It was developed by McLean et al. [18].

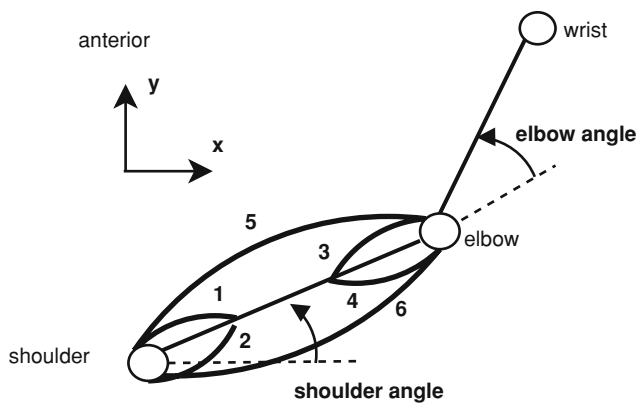


Fig. 1 The two-dimensional model of the human arm used in this study. It has two degrees of freedom (shoulder flexion–extension and elbow flexion–extension), and six muscles numbered in the figure: 1 anterior deltoid, 2 posterior deltoid, 3 brachialis, 4 lateral triceps, 5 long head of biceps, and 6 long head of triceps. 1 and 2 cross the shoulder joint, 3 and 4 cross the elbow joint, and 5 and 6 cross both joints

The model can run both forward and inverse-dynamic simulations. In the forward-dynamic mode, the inputs are the activations of the six muscles, and the outputs are the angles and angular velocities of the two joints. In the inverse-dynamic mode, the inputs are the desired angles and angular velocities of the two joints, and the outputs are the required muscle activations. Both modes are used in the controller design and testing, as described in the next section. In the case of inverse-dynamic simulations, an optimization routine is needed to distribute the muscle forces based on the required joint torques, since there are more muscles than degrees of freedom. The objective function currently used was proposed by Praagman [21] to minimize energy consumption:

$$E_m = E_f + E_a = m \left\{ c_1 \frac{F_m}{PCSA} + c_2 \left(\frac{F_m}{F_{\max}} \right)^2 \right\} \quad (1)$$

where E_f and E_a are the muscle energy consumption due to the detachment of cross bridges and re-uptake of calcium, respectively, m is the muscle mass, F_m is the muscle force, PCSA is the physiological cross-sectional area, F_{\max} is the maximum muscle force, and c_1 and c_2 are two constants chosen such that 50–50 contribution from the linear and nonlinear terms at 50% activation is reached [21].

2.2 The controller

The controller consists of a feedforward and a feedback part (Fig. 2). The feedforward part is an ANN trained to behave like the inverse-dynamic model of the arm, so that it can generate the muscle activations required for a desired movement, based on knowledge of system

dynamics. This ANN is a two-layer network with a sigmoidal transfer function in the hidden layer, and a linear transfer function in the output layer. The inputs are the shoulder and elbow angles, and four past values of the angles, each delayed by 80 ms, used to estimate angular velocity. The outputs are activation levels for the six muscles.

Training of this ANN was done using data from inverse-dynamic simulations using the model. Forty movements with a duration of 60 s each were used for training. The movements were trajectories covering the range of both shoulder and elbow angles, with bell-shaped velocity profiles. The bell-shaped curves had random amplitudes, within the range of the angles, and random maximum velocities, up to 4 rad/s. This is the type of trajectory seen in goal-directed movements performed by able-bodied subjects [12], and it covers a wide range of amplitudes and frequencies, which is a desired characteristic for ANN learning [23]. In order to improve learning, Gaussian white noise with mean zero and standard deviation equal to 5% of the maximum angle was added to the input data, as described in [20]. The ability of the network to approximate the inverse-dynamic model was quantified using the root mean squared error between the original model-based activations (i.e., the outputs of the inverse-dynamic simulations) and ANN-predicted muscle activations (i.e., the output of the ANN that approximates the inverse-dynamic model) as follows:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N (y_i^{\text{ANN}} - y_i^{\text{model}})^2}{N}} \quad (2)$$

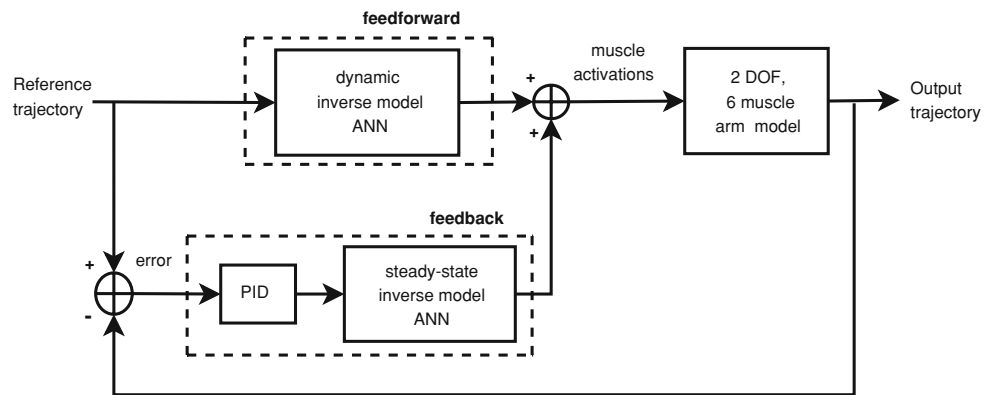
Networks with different numbers of neurons in the hidden layer were trained, and it was found that the network with 15 neurons had the smallest testing RMS error: 0.07 (no units, this is normalized muscle activation ranging from 0 to 1). Consequently, this ANN was chosen as the feedforward part of the controller.

The feedback part of the controller was designed as a traditional PID controller in series with a static ANN. The PID controller is described by:

$$u_i(t) = K_{Pi}e_i(t) + K_{Di}\frac{de_i(t)}{dt} + K_{Ii}\int_0^t e_i(t), \quad i = 1, 2 \quad (3)$$

where K_{Pi} , K_{Di} and K_{Ii} are the proportional, derivative and integral gains for the shoulder ($i = 1$) and the elbow ($i = 2$). The outputs of the PID controller (u_1 and u_2) then serve as the inputs to the static ANN. The PID controller gains were tuned using the Ziegler–Nichols step response method, which correlates the controller parameters to features of the step response [5], with additional manual fine-tuning. The values found after tuning were: $K_P = 2.5$,

Fig. 2 The combined feedforward and feedback (FF + FB) controller scheme, shown controlling the 2 DOF model from Fig. 1. The “reference trajectory” includes the desired joint angles (shoulder and elbow) and desired joint angular velocities (shoulder and elbow). The output trajectories indicate the same quantities as generated by the overall system (controller plus arm model)



$K_I = 3$ and $K_D = 1$ for the shoulder angle, and $K_P = 2.5$, $K_I = 5$ and $K_D = 1$ for the elbow angle.

The feedback ANN has the same architecture as the feedforward ANN (two layers, sigmoidal and linear transfer function in the hidden and output layer, respectively). However, this ANN is static, since it uses only the present angle values (and no past values) as inputs. In order to obtain training data, a 100 by 100 grid of shoulder and elbow angles was created, and inverse-dynamic simulations were run with these angles as constant inputs. From each simulation, we recorded the set of muscle activations calculated by the model in the steady state at each of the 10,000 positions. The angles and the corresponding muscle activations were used to train networks with different numbers of neurons in the hidden layer, and the network with 10 neurons was chosen because its RMS error of 0.09 was the lowest.

The controller was then tested using forty 60-s trajectories, similar to the trials used for training (i.e., goal-directed movements with bell-shaped velocity profiles of different amplitudes and maximum velocities) but not the same. Its performance was quantified using the RMS error between the desired and actual trajectory (Eq. 2).

In order to investigate the benefits of each component of the controller, the feedforward-only (FF) component, feedback-only (FB) component, and combination of feedforward and feedback (FF + FB) components were tested separately.

To test the performance of the controller for more realistic, non-ideal conditions, a simple model of muscle fatigue was incorporated into the model. For this fatigue model, the maximum force (F_j) of each muscle was linearly reduced according to the equation:

$$F_j(t) = F_{0j}(t)(1 - 0.005 \times t), \quad j = 1, \dots, 6 \quad (4)$$

where F_{0j} is the maximum isometric force for muscle j . Using a “fatigue rate” of 0.005 resulted in a 50% force

reduction over 100 s, which has also been used by [1] and [20]. The performance of the controller in the presence of fatigue was tested using forty 160-s goal-oriented movements, similar to the ones described earlier, and ten cyclical 160-s movements, with repeated reaching between two randomly chosen goals. The cyclical movements were included because they can clearly show the effects of increasing muscle fatigue on the model trajectory. The performance of the controller in the presence of fatigue was quantified using the RMS error between the desired and actual trajectory (Eq. 2) for all 50 trials.

To test the ability of the controller to resist external disturbances, forces of different amplitudes were applied to the wrist of the model, in both the x (medial-lateral) and y (anterior-posterior) directions. The forces were applied during goal-oriented movements similar to those described earlier, and tended to displace the arm from the desired trajectory. They had bell-shaped amplitudes, and lasted between 1 s (simulating, for example, a sudden push) and 10 s (e.g., carrying an object along the trajectory). The maximum amplitude of the perturbations was drawn from a normal distribution with standard deviation 1 N and four different means: 1, 5, 10 and 15 N, to simulate smaller or larger disturbances. Ten 60-s goal-oriented movements were used for testing: first they were performed without perturbations, and then they were performed again for each case of force amplitude (about 1, 5, 10 and 15 N). The mean RMSE between the desired and actual trajectory was calculated for the ten trials in each case.

Finally, the trajectory errors were also expressed in terms of endpoint (ep) position, since the hand location is more functionally meaningful than the individual joint angles. This was calculated as the RMS of the Euclidean distance between the desired and actual endpoint position:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N \left[(\text{desired_ep}_x - \text{actual_ep}_x)^2 + (\text{desired_ep}_y - \text{actual_ep}_y)^2 \right]}{N}}$$

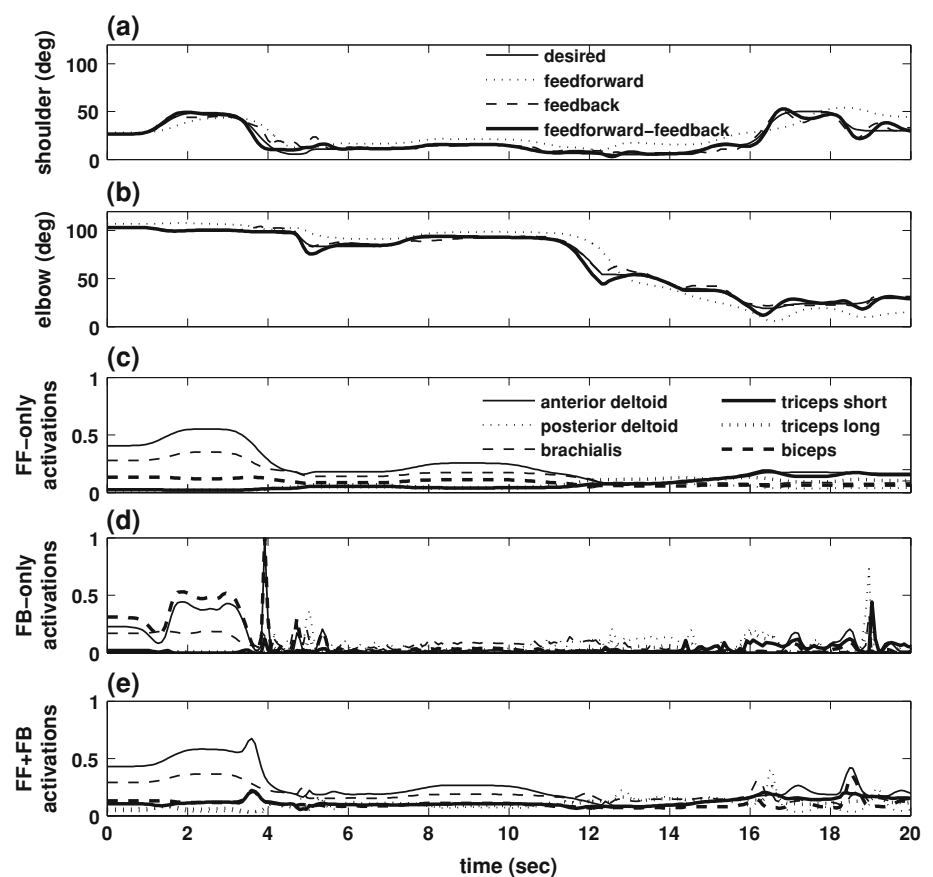
3 Results

Figure 3 shows the performance of the FF, FB, and FF + FB controller for a 20-s segment of one of the testing movements. Panel (a) shows the shoulder angle, and panel (b) shows the elbow angle. The RMS error for the three controllers was 8.3, 4.6, and 3.3°, respectively, for the shoulder angle, and 8.4, 4.0, and 2.7°, respectively, for the elbow angle. The muscle activations corresponding to that movement are shown in panels (c), (d) and (e) for the FF, FB and FF + FB controllers, respectively. Note that the feedforward-only muscle activations were much smoother and lower in maximum magnitude than the feedback-only activations, with the overall feedforward–feedback activations showing a combination of smooth, moderate amplitude activations when the desired angles changed slowly and fast adjustments when the desired angles changed rapidly.

The mean tracking performance of the three controllers across all forty 60-s testing movements, calculated as the mean RMS error (Eq. 2), is summarized in Fig. 4. The combined feedforward–feedback controller had the best performance, with an RMS error of less than 4° for both the shoulder and the elbow. The feedback-only controller had an RMS error of 5–6°, and the feedforward-only controller had the worst tracking performance, with an RMS error of 15–20°. The three controllers were significantly different from each other, for both the shoulder and the elbow (paired *t* test, *P* < 0.001).

Figure 5 shows an example of a cyclical movement that was performed while the maximum forces of all six muscles were continuously decreased according to our fatigue model (Eq. 4). The movement is continued beyond the 160 s used to quantify the controller performance, to illustrate the model operation in the presence of extreme muscle fatigue: since the maximum muscle force was

Fig. 3 An example of the performance of the FF, FB and FF + FB controllers for a 20-s segment of a goal-oriented movement. *Panel a* illustrates the shoulder angles for the various controller configurations and *panel b* illustrates the corresponding elbow angles. *Panel c* shows the activations calculated by the FF-only controller, *panel d* by the FB-only controller, and *panel e* for the FF + FB controller



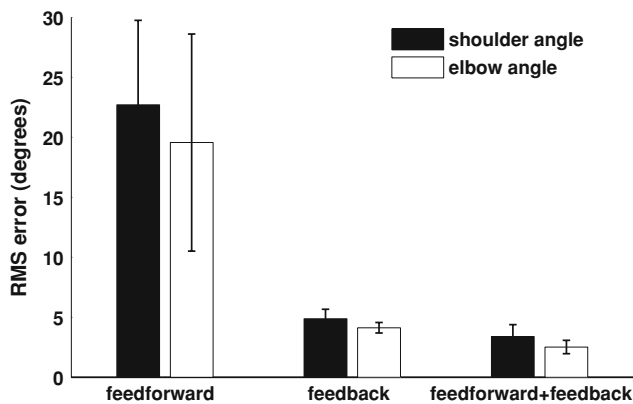


Fig. 4 The mean RMS errors for shoulder angles (*filled bars*) and elbow angles (*open bars*) across all forty 60-s trials tested using the three controllers: FF, FB, and FF + FB. The paired *t* test showed significant differences among the three controllers, for both the shoulder and the elbow

decreased by 0.5% per second, by the end of the 190-s movement the maximum force was 5% of the initial maximum force. The maximum force, as a percentage of the initial maximum force, is shown in panel (a). Panel (b) presents the resulting shoulder motions and panel (c) presents the resulting elbow motions. Panels (d) and (e) are the corresponding joint angle errors (absolute difference between desired and output angles) for the shoulder and elbow, respectively. Panel (f) shows the muscle activations calculated by the FF + FB controller. Note that the error for both the shoulder and elbow does not exceed 5° until after 160-s of movement, which corresponds to a maximum muscle force of just 20% of the “pre-fatigue” levels. For further decreases in maximum muscle forces, most muscle activation levels saturate at their maximum value of 1 (panel f) and the desired arm trajectory can no longer be accurately tracked.

Figure 6 shows the mean RMS errors across all fifty 160-s trials with fatigue as a function of time. The trials were split into 40-s intervals, and the RMS error was calculated for each one. The maximum force as a percentage of the initial maximum force is also shown for each interval. For both the shoulder and the elbow, the RMS error increased as the fatigue level increased. Towards the end of each trial, which corresponds to muscle forces that have decreased to levels of 40–20%, the RMS error was 10–15°. However, the RMS error was less than 8° for the first half of the trials, which corresponds to muscle force as low as 50%.

Figure 7 shows an example of the controller performance during an unperturbed movement (panels a, b, c), and for the same movement in the presence of two sets of random external disturbances at the wrist: small forces (panels d, e, f) and large forces (panels g, h, i). The forces are shown on the top panels: zero force for the unperturbed

movement (panel a), forces around 1 N in panel (d) and forces around 15 N in panel (g). Panels (b), (e) and (h) show the shoulder angles for the three cases, and (c), (f) and (i) show the elbow angles. For the unperturbed movement, the RMS errors were 2.7° (shoulder) and 2.5° (elbow). In the case of small perturbations, the RMS errors were 3.0° (shoulder) and 2.7° (elbow). Finally, in the case of large perturbations, the RMS errors were 5.8° (shoulder) and 3.4° (elbow).

The mean RMS error for the shoulder and elbow angles, for ten 60-s trials performed under five different conditions (unperturbed, external forces of about 1, 5, 10 and 15 N) are shown in Fig. 8. The error increased with the force amplitude, but in all cases it was below 10°.

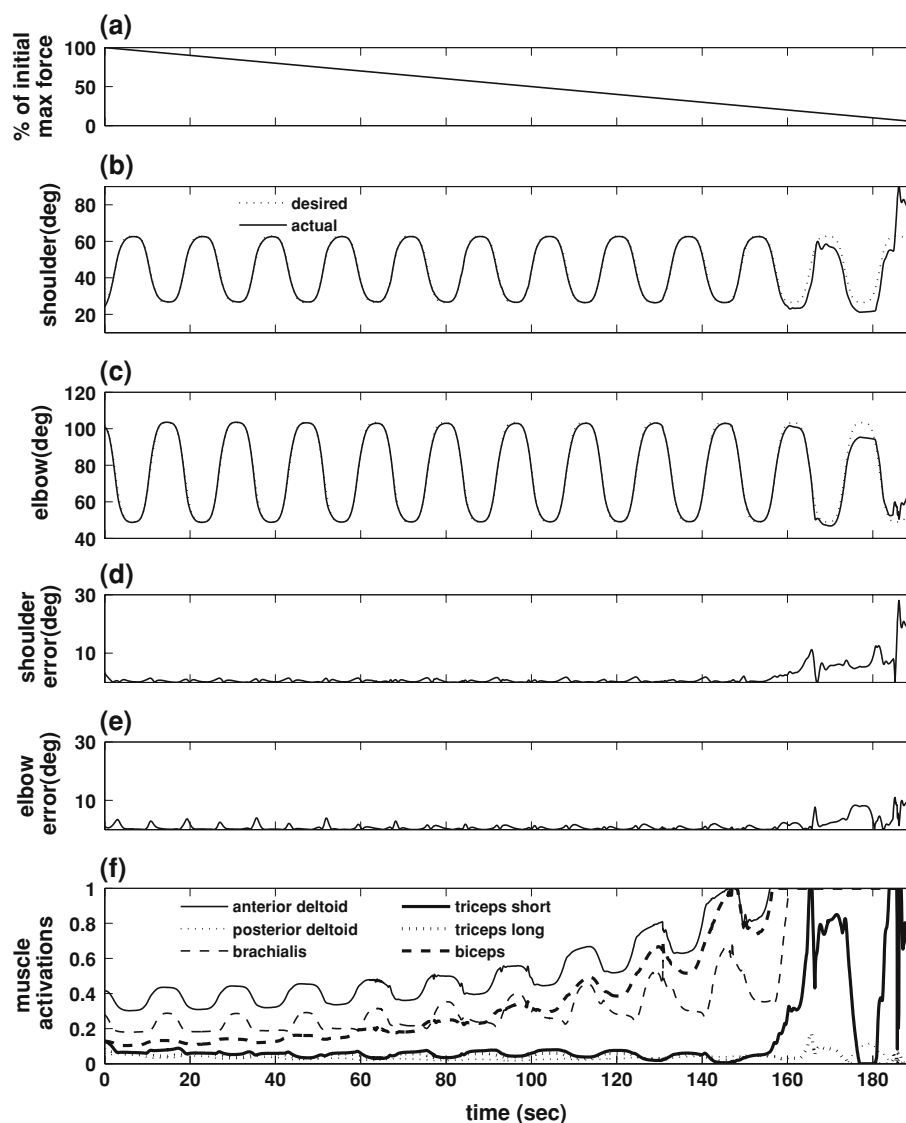
Finally, Fig. 9 shows the mean RMS errors for the endpoint position in four of the cases presented above, using ten trials for each case: ideal conditions (no fatigue or perturbations), large external forces of about 15 N, moderate fatigue that has reduced the muscle forces to 60–80% of their maximum output, and extreme fatigue that has reduced the muscle forces to 20–40%. The mean RMS errors for the four cases were 2.9, 4.8, 3.8 and 17.6 cm, respectively.

4 Discussion

A feedforward–feedback controller has been developed for a two-joint, six-muscle arm model. An inverse-dynamic model of the arm was used to train the ANN-based feedforward component so that it accounted for the complex interactions between muscles and joints, while solving the muscle redundancy problem by distributing the muscle forces according to a minimum energy consumption criterion. The inverse-dynamic model in the steady state was also used to train a separate static ANN that was used in conjunction with a linear PID controller in the feedback loop to create a neuro-PID feedback controller capable of handling the highly nonlinear and dynamic nature of the musculo-skeletal system. The controller showed excellent tracking performance during goal-oriented movements, with less than 4° joint error (3-cm endpoint error) in ideal conditions and less than 10° (6 cm) even in the case of considerable fatigue and large external disturbances.

Using the actual inverse-dynamic model for ANN training solved the muscle redundancy problem caused by the greater number of muscles (six) than joints (two) in the model. In addition, the muscle activations calculated by the controller were almost always at low levels (except for the case of extreme fatigue), since the criterion used to select the unique solution of the muscle force distribution minimized energy consumption. As shown in Figs. 3 and 5,

Fig. 5 Example of controller performance during progressive simulated fatigue. The maximum muscle force was decreased by 0.5% per second, so by the end of the 190-s movement the maximum force was only 5% of the initial maximum force. This is shown in *panel a*. *Panel b* illustrates the desired and actual shoulder angles and *panel c* illustrates the corresponding elbow angles. *Panels d* and *e* show the angular errors (absolute difference between desired and output angles) for the shoulder and elbow, respectively. *Panel f* shows the muscle activations calculated by the FF + FB controller



all muscle activations remained below 0.5 for almost all conditions until fatigue increased above 40–50%.

In the comparison among FF, FB and FF + FB controllers, FF had the poorest performance, even for conditions where there was no fatigue and no external perturbations. This occurred for two reasons. First, the inverse-dynamic model used to train the ANN was not the exact inverse of the forward model used in the controller evaluations because of numerical considerations. Specifically, the presence of numerical approximations and the complexity of the model used for the individual muscles prevent it from being perfectly inverted [11]. Also, practical solution of the redundancy problem by optimization requires the use of discrete time steps in the inverse-dynamic model, meaning the time-history dependence of muscle force was not captured as it was in the forward-dynamic model [26]. Second, even after training, the ANN was still only an approximation of the inverse-dynamic

model. This single, rather simple ANN was nonetheless capable of predicting the needed muscle activations with decent accuracy for a wide range of movements. In practice, we anticipate that the feedforward control will not be used without feedback for controlling arm movements, since it is very difficult to measure all the parameters needed for a perfect inverse-dynamic model of the real human arm [24]. However, as shown in Figs. 3 and 4, even this imperfect feedforward component substantially improved the performance of the overall FF + FB controller.

The inclusion of the FF component was beneficial in one more important way. The FB component had a tracking performance almost as good as FF + FB, but it produced muscle activations that are much larger and rapidly varying than the FF controller. For example, around the 4th second of the movement shown in Fig. 3, the activation of the anterior deltoid calculated by the FB-only controller

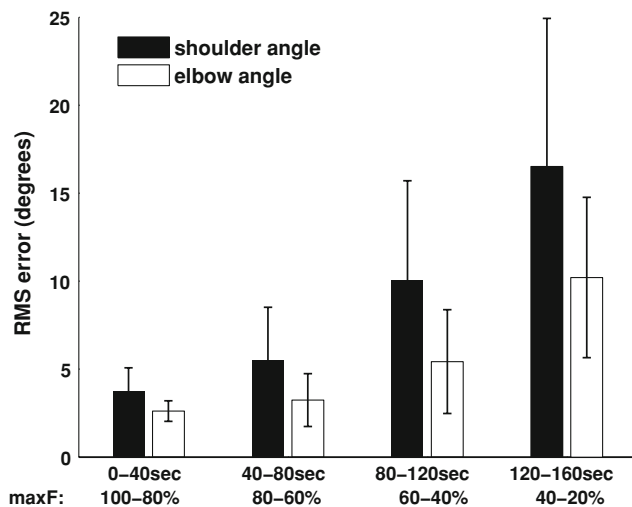


Fig. 6 Mean controller performance during progressive fatigue. The mean RMS errors across all fifty 160-s trials with fatigue are plotted for different intervals

rapidly approached the maximum. Addition of the FF component smoothed the activation and its maximum did not increase above 0.7.

The fatigue model used here was fairly simplistic, since it did not include activation level and stimulation frequency dependence, or a recovery with rest component [22]. However, our goal was to test the controller in the case of reduced muscle force output, and the fatigue model we chose created the right conditions for that. As shown in Figs. 5 and 6, under moderate fatigue conditions the tracking performance of the controller was as good as the non-fatigued case. During normal functional use (i.e., occasional goal-directed movements), it is very unlikely that all the muscles will fatigue below 50% without the chance for recovery.

The perturbations added to the wrist simulated obstacles or objects that are picked up and held during the movement. They had a much smaller effect on the controller performance than fatigue, with an average error of less than 10° for forces up to 15 N.

Figure 9 shows that the generally small joint angle errors translate into small endpoint errors: in ideal conditions the endpoint error is less than 3 cm, and even in the presence of the largest perturbations tested, the error does not exceed 6 cm. An error at that level is functionally trivial, since it would certainly allow the FES user to eat, or

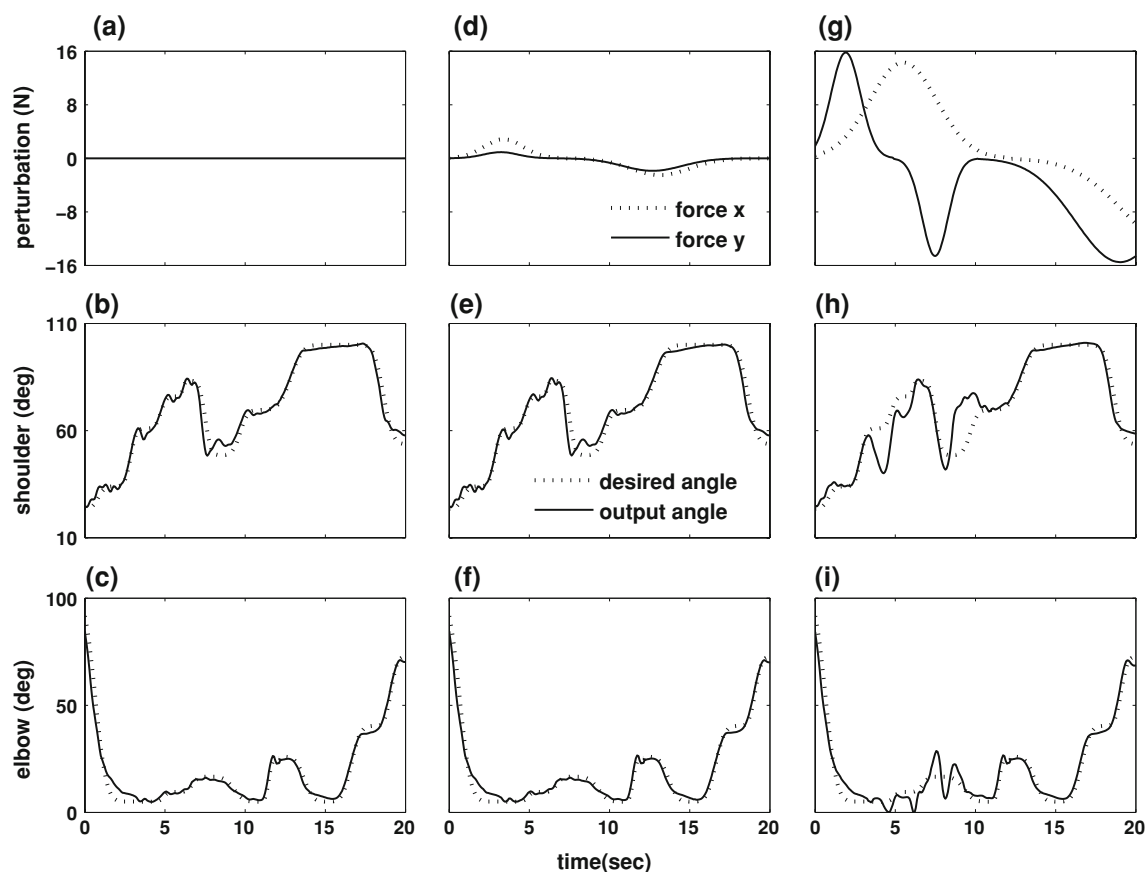


Fig. 7 Example of the FF + FB controller performance during a goal-oriented movement without perturbations (left panels), and in the presence of two sets of random external forces at the wrist: small

forces (middle panels) and large forces (right panels). The forces are shown in panels a, d and g. Panels b, e and h illustrate the shoulder angles, and panels c, f and i illustrate the elbow angles

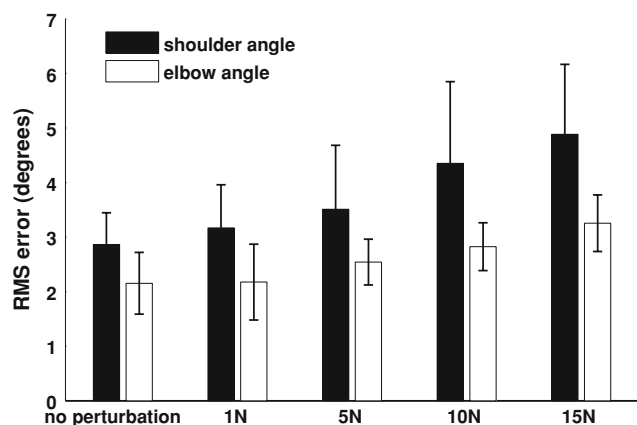


Fig. 8 The mean RMS error for the shoulder and elbow angle, from ten 60-s trials performed under five different conditions (unperturbed, external forces of about 1, 5, 10 and 15 N applied at the wrist)

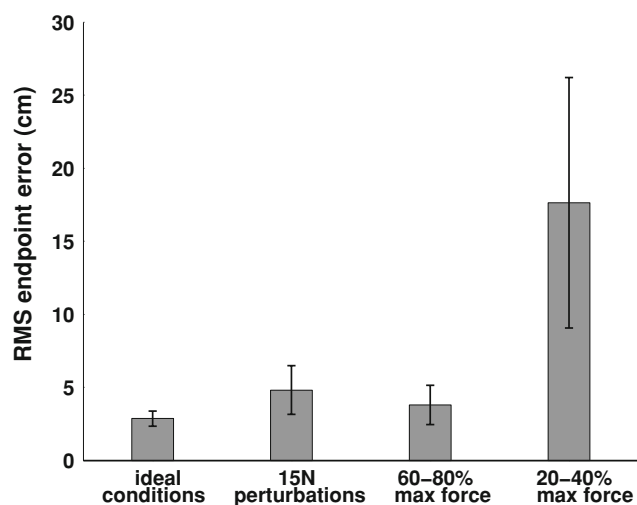


Fig. 9 The mean RMS error for the endpoint, for four cases presented previously, using ten trials for each case: ideal conditions, perturbations of about 15 N, fatigued muscles that operate at 60–80% of their “pre-fatigue” levels, and fatigued muscles that operate at 20–40% of their “pre-fatigue” levels

reach for various objects. The endpoint error approaches 20 cm only during extreme fatigue, when there is insufficient muscle force for trajectory tracking.

The controller evaluated here was tested using a two-joint, six-muscle system, but it could easily be extended for use with a full arm model. If the inverse-dynamic model is available, the ANN could be trained with more inputs and outputs, although a larger number of neurons in the hidden layer may be needed. One important practical benefit of using PID in series with an ANN instead of a purely PID feedback loop is that the number of gains that need to be tuned is reduced to 6 in the case of our model (proportional, integral and derivative for each angle) instead of 36 (the previous 6 PID gains for each of the 6 muscles). This

will become a major advantage when the number of muscles increases.

For the controller to be implemented in the real FES system, the first step will be customizing the arm model to reflect the FES user’s arm as closely as possible, by including information about possible voluntary muscle forces, denervation, as well as which muscles are the targets for FES stimulation. A controller for this customized arm model will be built, but because of the differences between the model and the FES user’s arm, it is unlikely that the model-based ANN parameters will result in optimal performance. This controller will instead be used as a starting point for adaptation, using data collected from the FES-driven arm, as shown previously in [1]. While the FES system is in use, the stimulation levels and resulting movements will be recorded for either online or off-line adaptation of the ANN parameters. As both the feedforward and feedback ANN are adapted to the arm dynamics of the specific user, the controller performance will continue to improve, produce more accurate movements, and provide more functional benefits to the user.

Acknowledgments The authors would like to thank Dr. Antonie van den Bogert for his help with the muscle model implementation. This study was funded by NIH/NINDS contract N01-NS-5-2365.

References

1. Abbas JJ, Chizeck HJ (1995) Neural network control of functional neuromuscular stimulation systems: computer simulation studies. *IEEE Trans Biomed Eng* 42(11):1117–1127. doi:[10.1109/10.469379](https://doi.org/10.1109/10.469379)
2. Abbas JJ, Triolo RJ (1997) Experimental evaluation of an adaptive feedforward controller for use in functional neuromuscular stimulation systems. *IEEE Trans Rehabil Eng* 5(1):12–22. doi:[10.1109/86.559345](https://doi.org/10.1109/86.559345)
3. Adamczyk MM, Crago PE (2000) Simulated feedforward neural network coordination of hand grasp and wrist angle in a neuro-prosthesis. *IEEE Trans Rehabil Eng* 8(3):297–304. doi:[10.1109/86.867871](https://doi.org/10.1109/86.867871)
4. Åström KJ, Hägglund T (2001) The future of PID control. *Control Eng Pract* 9:1163–1175. doi:[10.1016/S0967-0661\(01\)00062-4](https://doi.org/10.1016/S0967-0661(01)00062-4)
5. Åström KJ, Hägglund T (2004) Revisiting the Ziegler–Nichols step response method for PID control. *J Process Contr* 14:635–650. doi:[10.1016/j.jprocont.2004.01.002](https://doi.org/10.1016/j.jprocont.2004.01.002)
6. Chang GC, Luh JJ, Liao GD et al (1997) A neuro-control system for the knee joint position control with quadriceps stimulation. *IEEE Trans Rehabil Eng* 5(1):2–11. doi:[10.1109/86.559344](https://doi.org/10.1109/86.559344)
7. Chizeck HJ, Kobetic R (1988) Control of functional neuromuscular stimulation systems for standing and locomotion in paraplegics. *Proc IEEE* 76(9):1155–1165. doi:[10.1109/5.9661](https://doi.org/10.1109/5.9661)
8. Crago PE, Lan N, Veltink PH et al (1996) New control strategies for neuroprosthetic systems. *J Rehabil Res Dev* 33(2):158–172
9. Davis JA, Triolo RJ, Uhler JP et al (2001) Surgical technique for installing an eight-channel neuroprosthesis for standing. *Clin Orthop Relat Res* 237–252. doi:[10.1097/00003086-200104000-00035](https://doi.org/10.1097/00003086-200104000-00035)

10. Ferrarin M, Palazzo F, Riener R, Quintern J (2001) Model-based control of FES-induced single joint movements. *IEEE Trans Neural Syst Rehabil Eng* 9(3):245–257. doi:[10.1109/7333.948452](https://doi.org/10.1109/7333.948452)
11. Happee R (1994) Inverse dynamic optimization including muscular dynamics, a new simulation method applied to goal directed movements. *J Biomech* 27(7):953–960. doi:[10.1016/0021-9290\(94\)90267-4](https://doi.org/10.1016/0021-9290(94)90267-4)
12. Happee R, van der Helm FC (1995) The control of shoulder muscles during goal directed movements, an inverse dynamic analysis. *J Biomech* 28(10):1179–1191. doi:[10.1016/0021-9290\(94\)00181-3](https://doi.org/10.1016/0021-9290(94)00181-3)
13. Jonic D, Popovic S (1997) Machine learning for prediction of muscle activations for a rule-based controller. In: *Proceedings of the 19th annual international conference of the IEEE*, 4: 1781–1784
14. Karniel GF, Inbar A (2000) Human motor control: learning to control a time-varying, nonlinear, many-to-one system. *IEEE Trans Syst Man Cybern C* 30(1):1–11. doi:[10.1109/5326.827449](https://doi.org/10.1109/5326.827449)
15. Kilgore KL, Peckham PH, Thrope GB et al (1989) Synthesis of hand grasp using functional neuromuscular stimulation. *IEEE Trans Biomed Eng* 36(7):761–770. doi:[10.1109/10.32109](https://doi.org/10.1109/10.32109)
16. Klein Breteler MD, Spoor CW, van der Helm FC (1999) Measuring muscle and joint geometry parameters of a shoulder for modeling purposes. *J Biomech* 32(9):1191–1197. doi:[10.1016/S0021-9290\(99\)00122-0](https://doi.org/10.1016/S0021-9290(99)00122-0)
17. Lan N, Crago PE, Chizeck HJ (1991) Control of end-point forces of a multijoint limb by functional neuromuscular stimulation. *IEEE Trans Biomed Eng* 38(10):953–965. doi:[10.1109/10.88441](https://doi.org/10.1109/10.88441)
18. McLean SG, Su A, van den Bogert AJ (2003) Development and validation of a 3-D model to predict knee joint loading during dynamic movement. *J Biomech Eng* 125(6):864–874. doi:[10.1115/1.1634282](https://doi.org/10.1115/1.1634282)
19. Norgaard M, Ravn O, Poulsen NK, Hansen LK (2000) *Neural networks for modelling and control of dynamic systems*. Springer, London
20. Pedrocchi A, Ferrante S, De Momi E, Ferrigno G (2006) Error mapping controller: a closed loop neuroprosthesis controlled by artificial neural networks. *J Neuroeng Rehabil* 3:25. doi:[10.1186/1743-0003-3-25](https://doi.org/10.1186/1743-0003-3-25)
21. Praagman M, Chadwick EK, van der Helm FC, Veegeer HEJ (2006) The relationship between two different mechanical cost functions and muscle oxygen consumption. *J Biomech* 39(4): 758–765. doi:[10.1016/j.jbiomech.2004.11.034](https://doi.org/10.1016/j.jbiomech.2004.11.034)
22. Riener R, Quintern J, Schmidt G (1996) Biomechanical model of the human knee evaluated by neuromuscular stimulation. *J Biomech* 29(9):1157–1167. doi:[10.1016/0021-9290\(96\)00012-7](https://doi.org/10.1016/0021-9290(96)00012-7)
23. Stroeve S (1996) Learning combined feedback and feedforward control of a musculoskeletal system. *Biol Cybern* 75(1):73–83. doi:[10.1007/BF00238741](https://doi.org/10.1007/BF00238741)
24. van der Helm FC (1994) A finite element musculoskeletal model of the shoulder mechanism. *J Biomech* 27(5):551–569. doi:[10.1016/0021-9290\(94\)90065-5](https://doi.org/10.1016/0021-9290(94)90065-5)
25. Veltink PH, Chizeck HJ, Crago PE, El-Bialy A (1992) Nonlinear joint angle control for artificially stimulated muscle. *IEEE Trans Biomed Eng* 39(4):368–380. doi:[10.1109/10.126609](https://doi.org/10.1109/10.126609)
26. Yoshida N, Tomita T, Honda S, Saitoh E (2002) Functional neuromuscular stimulation for articular angle control with an inverse dynamics model tuned by a neural network. *Ergonomics* 45(9):649–662. doi:[10.1080/00140130210142074](https://doi.org/10.1080/00140130210142074)